

**CERTIFICATE**

This is to certify that, the project entitled “**Study Buddy**” being submitted for the partial fulfillment of the degree of **Master of Computer Application** by him to **Sinhgad Institute of Management and Computer Application affiliated to Savitribai Phule Pune University, Pune** is the result of the original work completed by **GOPALE SHREYAS BALASAHEB** under the guidance of **Prof. Poonam Sawant** To the best of our knowledge and belief, this work has not been previously submitted by the award of any degree or diploma of Savitribai Phule Pune University or any other University.

PLACE: PUNE

DATE:

**Dr. Vijaya Puranik**

**Director SIMCA**

**Dr. Rajesh Gawali**

**Academic Co-ordinator &**

**Project Co-Ordinator**

**Prof. Poonam Sawant**

**Internal Guide**

**External Examiner**



**A**

**PROJECT REPORT**

**ON**

**“Study Buddy”**

**AT**

**“Sinhgad Institute of Management and Computer Application”**

**SUBMITTED**

**TO**

**SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE**

**Partial Fulfilment OF**

**MASTER OF COMPUTER APPLICATION**

**(MCA-II, SEM- III)**

**BY**

**GOPALE SHREYAS BALASAHEB**

**UNDER THE GUIDANCE OF**

**Prof. Poonam Sawant**

**THROUGH**

**THE DIRECTOR**

**SINHGAD INSTITUTE OF MANAGEMENT AND COMPUTER APPLICATION (SIMCA), NARHE, PUNE**

**(AY. 2023-2024)**

**DECLARATION**

I, the undersigned hereby declare that the project titled **“Study Buddy”,** being submitted for the award of degree of **Master of Computer Application** by me to **Sinhgad Institute of Management and Computer Application(SIMCA) affiliated to Savitribai Phule Pune University** is the result of an independent work carried out under the guidance of **Prof. Poonam Sawant,** is my original work .

Further I declare that this project has not been submitted to this or any Institution for the award of any degree.

PLACE: PUNE

DATE:

**Gopale Shreyas Balasaheb**

**ACKNOWLEDGEMENT**

The project developed for the MCA was not possible without the persons and organizations that helped me in completing this. I am deeply grateful to all whose enthusiasm and energy transformed my vision of this study into reality.

I extend my sincere thanks to **Dr. Rajesh Gawali** project co-ordinator for making it easy to work in the organization and providing me needed guidance throughout the project keeping it focused and on track. I am really thankful to him for the extended knowledge imparted to me during the course of project development.

I take this opportunity to thank **Prof. Poonam Sawant** class coordinator, Head of Department **Dr. Rajesh Gawali** and our Director **Dr. Vijaya Puranik**, for encouragement and guidance throughout the progress of this report

**Gopale Shreyas Balasaheb**

**Index**

|  |  |  |
| --- | --- | --- |
| **Ch. No.** | **Title** | **Page No.** |
| **1** | **Introduction**   * 1. Abstract   2. Existing System & Need For System   3. Scope Of The System   4. Operating Environment- Hardware & Software | **6**  6  7  9  11 |
| **2** | **Proposed System**   * 1. Feasibility Study   2. Objective Of Proposed System   3. User Of The System | **12**  12  14  16 |
| **3** | **Analysis & Design**   * 1. System Requirement   2. Data Dictionary   3. ER Diagram   4. Class Diagram   5. Use Case Diagram   6. Activity Diagram   7. Deployment Diagram   8. Output Screens | **18**  18  21  23  24  25  28  35  38 |
| **4** | **Testing**   * 1. Test Strategy   2. Unit Test Plan   3. Integration Test Plan   4. Performance Test Plan | **41**  41  42  43  44 |
| **5** | **Limitation Of Proposed System** | 46 |
| **6** | **Proposed Enhancement** | 47 |
| **7** | **Conclusion** | 48 |
| **8** | **Bibliography** | 49 |

**Introduction: Study Buddy**

Study Buddy is a dynamic and feature-rich Learning Management System designed to revolutionize online education. In an era where digital learning is rapidly evolving, Study Buddy aims to create an engaging, accessible, and efficient platform for both educators and learners. This system empowers instructors using Study Buddy to deliver high-quality educational content, while providing students with a seamless and interactive Study Buddy learning experience. Study Buddy is built with a user-centric approach, offering a diverse array of features to facilitate effective online learning, including robust user management, comprehensive course creation and management, advanced learning and progress tracking, engaging communication and collaboration features, intelligent AI-powered tools, and gamification and monetization options. Study Buddy's technical foundation leverages modern technologies to ensure a scalable and reliable platform. The goal of Study Buddy is to provide a valuable tool for anyone involved in online education, fostering a more effective and engaging learning environment.

**1.1 ABSTRACT**

Study Buddy is a modern Learning Management System designed to bridge the gap between traditional education and the evolving landscape of online learning. It provides a comprehensive platform for instructors to create and manage engaging courses, while offering students a seamless and interactive learning experience. Leveraging a user-centric design, Study Buddy incorporates features such as robust user management, detailed progress tracking, interactive quizzes, automated certificate generation, and integrated communication tools. Beyond basic functionality, Study Buddy integrates AI-powered features like personalized course recommendations and an AI chatbot for instant support, enhancing the learning journey. Furthermore, it incorporates gamification elements and monetization options, fostering a dynamic and sustainable educational environment. Built on a scalable and reliable tech stack, Study Buddy aims to empower educators and learners alike, fostering a more effective and accessible online education experience.

**1.2 EXISTING SYSTEM & NEED FOR SYSTEM**

**Existing Systems**

The online learning landscape is populated with a variety of Learning Management Systems (LMS), each catering to different needs and user bases. These systems range from open-source platforms to proprietary software, and they serve diverse educational sectors, including academic institutions, corporate training, and individual course creators.

Commonly encountered existing LMS systems include:

* Moodle: A widely used open-source LMS known for its flexibility and customization options, often favored by educational institutions.
* Canvas: A popular cloud-based LMS that emphasizes user-friendly interfaces and robust course management tools, prevalent in higher education.
* Blackboard: A comprehensive LMS with a long history, offering a wide range of features for course delivery, assessment, and communication, commonly used in academic and corporate settings.
* Google Classroom: A simplified LMS integrated with Google Workspace, designed for ease of use and seamless collaboration, particularly popular in K-12 education.
* Teachable/Thinkific: Platforms focused on enabling individual course creators to build and sell online courses, emphasizing marketing and monetization tools.
* LMS solutions provided by companies like Adobe Captivate Prime, SAP Litmos, and others These types of LMS are normally geared towards business and corporate training.

These existing systems provide a foundation for online learning, offering features such as course content delivery, assessment tools, and communication forums. However, they often vary significantly in their approach to user engagement, personalization, and integration of emerging technologies like AI.

**Need for System:**

Despite the presence of various existing Learning Management Systems, a significant gap remains in providing a truly engaging, personalized, and intelligent online learning experience. The need for Study Buddy stems from several critical shortcomings observed in current solutions.

Firstly, many existing platforms struggle to foster active student engagement. They often rely on passive content delivery, lacking interactive elements like gamification, personalized quizzes, and robust communication tools. This leads to decreased student motivation and a less effective learning process.

Secondly, personalization is often overlooked. Generic course delivery fails to cater to diverse learning styles and individual student needs. The ability to provide tailored learning paths, AI-powered recommendations, and adaptive assessments is crucial for maximizing learning outcomes.

Thirdly, the integration of emerging technologies, particularly AI, is underutilized. AI-powered features like automated support, plagiarism detection, and personalized learning recommendations can significantly enhance the learning experience, yet they are often absent or poorly implemented in existing systems.

Additionally, user interfaces are often complex and unintuitive, creating barriers for both instructors and students. A user-friendly and streamlined platform is essential for seamless learning.

Furthermore, instructors seeking to monetize their expertise encounter limitations. Many platforms offer restrictive or complex monetization options, hindering their ability to create sustainable online courses.

Finally, the fragmented nature of many LMS ecosystems necessitates the use of multiple external tools, leading to a disjointed user experience. Integrated features for live classes, communication, and payment processing are essential for a unified and efficient learning environment.

Therefore, Study Buddy is needed to address these critical gaps. It aims to provide an engaging, personalized, and intelligent learning platform that empowers educators and learners alike. By leveraging modern technologies and prioritizing user experience, Study Buddy seeks to redefine the online learning landscape.

**1.3 SCOPE OF THE SYSTEM:**

The scope of the Study Buddy encompasses the development and deployment of a comprehensive online learning platform designed to facilitate effective and engaging education. This system aims to provide a robust and user-friendly environment for both instructors and students, addressing the diverse needs of online learning.

Specifically, the scope of Study Buddy includes:

* User Management:
  + Implementation of a secure and scalable user authentication and authorization system, supporting distinct roles for administrators, instructors, and students.
  + Management of user profiles, including registration, login, and profile editing.
* Course Management:
  + Creation and management of online courses, including modules, lessons, and assignments.
  + Support for diverse content formats, such as videos, PDFs, text documents, and interactive quizzes.
  + Enrollment and management of students in courses.
* Learning and Progress Tracking:
  + Tracking of student progress through lessons and assignments.
  + Implementation of interactive quizzes with automated grading.
  + Generation of course completion certificates.
  + Visual progress indicators.
* Communication and Collaboration:
  + Integration of discussion forums and chat systems for student-instructor interaction.
  + Integration with third-party live class platforms (e.g., Zoom, Google Meet).
  + Notification system for announcements, assignments, and events.
* Intelligent Features:
  + Implementation of AI-powered course recommendations based on student learning history.
  + Development of an AI chatbot for answering student queries.
  + Assignment plagiarism detection.
* Gamification and Monetization:
  + Implementation of gamification elements, such as badges and leaderboards.
  + Integration of payment gateways (e.g., Stripe, PayPal) for course subscriptions and purchases.
* Admin Dashboard:
  + Provision of an administrative interface for managing users, courses, and system settings.
  + Generation of reports on student progress and course performance.
* Frontend Development:
  + Development of a user-friendly and responsive web interface.
  + Implementation of all user interfaces that are needed for the functionality of the site.
* Backend Development:
  + Development of all needed API's for the frontend to interact with the database.
  + Development of all backend logic needed for the functionality of the site.
* Database Management:
  + Design and implementation of a robust and scalable database schema.
  + Management of data storage and retrieval.

The scope of Study Buddy is designed to provide a comprehensive and integrated online learning experience, catering to the needs of both educators and learners in a dynamic and evolving digital environment.

**1.4 OPERATING ENVIRONMENT:**

**Software Requirements:**

* + Operating System:
  + Server-Side: Linux-based OS (e.g., Ubuntu, CentOS) or Windows Server (preferred: Linux).
  + Client-Side: Modern Web Browsers (Chrome, Firefox, Safari, Edge) compatible with HTML5, CSS3, and JavaScript (ES6+).

**Database Management:**

* + PostgreSQL: For robust data storage and management.

**Development Tools:**

* + Integrated Development Environment (IDE):
  + Backend: PyCharm, VS Code (with Python extensions).
  + Frontend: VS Code, or IDE specific to the frontend framework chosen (e.g., if React, VS Code with React extensions).
  + Programming Languages:
  + Backend: Python (Django/Django Rest Framework).
  + Frontend: HTML5, CSS3, JavaScript (React.js or Django Templates).

**Web Server:**

* + Nginx or Apache with WSGI (Gunicorn or uWSGI).

**Authentication:**

* + Django's built in authentication, or JWT.
  + Third party API's
  + API's related to Zoom/Google Meet, Stripe/Paypal, and any other external services.

**Hardware Requirements:**

**Server-Side Hardware:**

* + Processor: Intel Core i5 or equivalent.
  + RAM: 8 GB minimum.
  + Storage: 250 GB SSD minimum (for OS and application), 1 TB HDD or equivalent (for data storage and backups).
  + Network: Stable internet connection for cloud services or server hosting.

**Client-Side Hardware:**

* + Processor: Any modern processor capable of running modern web browsers smoothly.
  + RAM: 4 GB minimum.
  + Storage: Sufficient storage for browser cache and user data.
  + Network: Stable internet connection for accessing the LMS.

**Chapter No 2. PROPOSED SYSTEM**

Study Buddy is proposed as a comprehensive online learning platform designed to revolutionize the educational experience. It aims to address the limitations of existing Learning Management Systems by offering a more engaging, personalized, and efficient environment for both educators and learners. This system prioritizes a user-centric design, ensuring intuitive navigation and tailored functionalities for distinct user roles, including administrators, instructors, and students. Study Buddy will provide robust course management capabilities, enabling instructors to easily create and manage courses, modules, and lessons, while supporting diverse content formats. The learning experience will be enhanced through features like progress tracking with visual indicators, interactive quizzes with automated grading, and automated certificate generation. Integrated communication and collaboration tools, such as discussion forums, real-time chat, and seamless live class platform integration, will foster a vibrant learning community. Study Buddy will leverage intelligent features, including AI-powered course recommendations, an AI chatbot for instant support, and assignment plagiarism detection, to further enhance the learning process. Gamification elements and integrated payment gateways will contribute to increased engagement and monetization opportunities. Built on a scalable and reliable architecture, utilizing Django and Django Rest Framework for the backend, React.js or Django Templates for the frontend, and PostgreSQL for robust data management, Study Buddy aims to provide a modern and reliable online learning environment. The overarching objective is to create a holistic learning ecosystem that empowers educators and learners, ultimately redefining the online education landscape.

**2.1 Feasibility Study**

**a) Technical Feasibility:**

* **Assessment:**
  + The technical feasibility of Study Buddy is high. The proposed technology stack (Django/Django Rest Framework, React.js or Django Templates, PostgreSQL) is well-established and widely used for web application development.
  + Cloud-based deployment options (AWS, Azure, Google Cloud) provide scalable and reliable infrastructure.
  + API integrations for third-party services (Zoom, Google Meet, Stripe, PayPal) are readily available and well-documented.
  + AI and machine learning components can be implemented using available libraries and frameworks (e.g., TensorFlow, scikit-learn).
* **Evaluation:**
  + The development team possesses the necessary skills and expertise to implement the proposed technologies.
  + The technology stack is scalable and can handle anticipated user traffic.
  + Security measures can be implemented to protect user data and prevent vulnerabilities.
* **Conclusion:**
  + Study Buddy is technically feasible. The proposed technologies and infrastructure are suitable for building a robust and scalable online learning platform.

**b) Economic Feasibility:**

* **Assessment:**
  + The economic feasibility of Study Buddy depends on the cost of development, deployment, and maintenance, as well as the potential revenue generated from course subscriptions, premium features, and instructor commissions.
  + Market research indicates a growing demand for online learning platforms, suggesting a strong potential for revenue generation.
  + Cost analysis of hardware, software, cloud services, and personnel.
* **Evaluation:**
  + A detailed financial model should be developed to estimate the return on investment (ROI) and breakeven point.
  + Pricing strategies should be developed to attract users while ensuring profitability.
  + Explore potential sources of funding, such as venture capital, grants, or bootstrapping.
* **Conclusion:**
  + Study Buddy is economically feasible, provided that a sound financial model is developed and implemented. The potential for revenue generation is significant, given the growing demand for online learning.

**c) Operational Feasibility:**

* **Assessment:**
  + The operational feasibility of Study Buddy depends on the ability to effectively manage and maintain the platform, as well as the acceptance and adoption of the system by users.
  + Training and support should be provided to instructors and students to ensure smooth onboarding and utilization of the platform.
  + The platform must be easily maintainable, and updates must be able to be rolled out smoothly.
* **Evaluation:**
  + A clear operational plan should be developed, outlining roles and responsibilities, support processes, and maintenance procedures.
  + User feedback should be collected and incorporated into ongoing improvements.
  + The platform should be designed to be user-friendly and intuitive, minimizing the need for extensive training.
* **Conclusion:**
  + Study Buddy is operationally feasible, provided that a well-defined operational plan is implemented, and ongoing support and maintenance are provided. User feedback should be a key part of the maintenance of the system.

**2.2 Objectives of the Proposed System**

The Study Buddy system is designed with several key objectives to create a transformative online learning experience. These objectives are:

1. **Enhance Accessibility and Flexibility:** 
   * To provide a platform that enables learning anytime, anywhere, catering to diverse learner needs and schedules.
   * To break down geographical barriers to education, making quality learning resources available to a wider audience.
2. **Improve Engagement and Motivation:** 
   * To create an interactive and engaging learning environment that fosters active participation and knowledge retention.
   * To utilize gamification, personalized learning paths, and interactive content to increase student motivation.
3. **Facilitate Effective Course Management:** 
   * To empower instructors with intuitive tools for creating, managing, and delivering high-quality online courses.
   * To streamline administrative tasks, allowing instructors to focus on teaching and student engagement.
4. **Promote Personalized Learning:** 
   * To leverage AI and data analytics to provide personalized learning experiences tailored to individual student needs and learning styles.
   * To offer adaptive assessments and personalized course recommendations.
5. **Foster Collaboration and Communication:** 
   * To create a vibrant learning community through integrated communication tools, such as discussion forums and real-time chat.
   * To facilitate seamless interaction between instructors and students.
6. **Ensure Scalability and Reliability:** 
   * To build a robust and scalable platform that can handle increasing user traffic and data volumes.
   * To provide a reliable and secure learning environment.
7. **Enable Monetization Opportunities:** 
   * To provide instructors with tools and features to monetize their expertise through course subscriptions and premium content.
   * To provide a platform that instructors can use to generate revenue.
8. **Leverage Technological Advancements:** 
   * To integrate cutting-edge technologies, such as AI and machine learning, to enhance the learning experience.
   * To stay up-to-date with emerging trends in online education.

**2.3 USER OF THE SYSTEM**

The Study Buddy is designed to serve a diverse range of users, each with specific needs and roles within the platform. These users can be broadly categorized as follows:

1. Administrators:
   * Administrators are responsible for the overall management and maintenance of the Study Buddy.
   * They have full control over user accounts, course management, system settings, and data security.
   * Their primary tasks include user management, system configuration, report generation, and ensuring the smooth operation of the platform.
2. Instructors/Teachers:
   * Instructors are the content creators and facilitators of online courses.
   * They use the platform to create and manage course content, upload materials, assign quizzes and assignments, and track student progress.
   * They interact with students through discussion forums, chat systems, and live classes.
   * They are also able to monetize their courses through the platform.
3. Students/Learners:
   * Students are the primary users of the learning content.
   * They enroll in courses, access learning materials, take quizzes and assignments, participate in discussions, and track their progress.
   * They benefit from personalized learning experiences, AI-powered recommendations, and instant support.
4. Technical Support Staff:
   * These users are responsible for the technical upkeep of the system.
   * They will trouble shoot user problems, maintain the servers, and update the software as needed.
5. Potential Course Creators/Entrepreneurs:
   * Individuals who wish to create and sell online courses.
   * They utilize the monetization features of Study Buddy to generate revenue from their educational content.

**Chapter No 3. ANALYSIS AND DESIGN**

**1. Functional Requirements:**

* **User Management:** 
  + User registration (student, instructor, administrator).
  + User login and logout.
  + Profile management (view, edit).
  + Role-based access control (RBAC).
  + User account management (create, delete, modify).
* **Course Management:** 
  + Course creation (title, description, content).
  + Module and lesson creation.
  + Content upload (videos, PDFs, documents, images).
  + Course enrollment and unenrolment.
  + Course categorization and search.
* **Learning Management:** 
  + Lesson completion tracking.
  + Quiz/assignment creation and management.
  + Automated quiz grading.
  + Assignment submission and grading.
  + Progress tracking (course, module, lesson).
  + Certificate generation upon course completion.
* **Communication and Collaboration**:
  + Discussion forums (course-specific, general).
  + Real-time chat system.
  + Live class integration (Zoom, Google Meet).
  + Notification system (announcements, assignments, messages).
* **AI-Powered Features:** 
  + Course recommendation engine (based on user history).
  + AI chatbot for student support.
  + Assignment plagiarism detection.
* **Monetization:** 
  + Course subscription management.
  + Payment gateway integration (Stripe, PayPal).
  + Gamification features (badges, leaderboards).
* **Admin Dashboard:** 
  + User management (view, edit, delete).
  + Course management (view, edit, delete).
  + System settings management.
  + Reporting and analytics (user activity, course performance).

**2. Non-Functional Requirements:**

Non-functional requirements describe how the system *should be*.

* **Performance:** 
  + Fast page load times.
  + Responsive user interface.
  + Ability to handle a large number of concurrent users.
* **Security:** 
  + Secure user authentication and authorization.
  + Data encryption (in transit and at rest).
  + Protection against common web vulnerabilities (e.g., SQL injection, XSS).
  + Regular security audits.
* **Usability:** 
  + Intuitive and user-friendly interface.
  + Consistent design and navigation.
  + Accessibility for users with disabilities.
* **Reliability:** 
  + High system availability (minimal downtime).
  + Data backups and recovery mechanisms.
  + Error handling and logging.
* **Scalability:** 
  + Ability to handle increasing user traffic and data volumes.
  + Scalable database and server infrastructure.
* **Maintainability:** 
  + Clean and well-documented code.
  + Modular design for easy updates and modifications.
  + Easy to debug.
* **Portability:** 
  + The system should be able to operate on various browsers, and potentially on different cloud platforms.

**3.2 Data Dictionary:**

**Users Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **user\_id** | **INT** | **10** | **Primary Key** | **Unique identifier for each user** |
| **2** | **username** | **VARCHAR** | **255** | **Unique** | **User's chosen username** |
| **3** | **email** | **VARCHAR** | **255** | **Unique** | **User's email address** |
| **4** | **password\_hash** | **VARCHAR** | **255** |  | **Hashed password for security** |
| **5** | **first\_name** | **VARCHAR** | **255** |  | **User's first name** |
| **6** | **last\_name** | **VARCHAR** | **255** |  | **User's last name** |
| **7** | **role** | **VARCHAR** | **50** |  | **User's role(admin, teacher, student)** |
| **8** | **course** | **VARCHAR** | **50** |  | **Select Course** |
| **9** | **registration\_date** | **TIMESTAMP** |  |  | **Date and time of user registration** |
| **10** | **profile\_picture** | **VARCHAR** | **255** | **Optional** | **Path to user's profile picture** |
| **11** | **bio** | **TEXT** |  | **Optional** | **User's biography** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **Faculty id** | **INT** |  | **Unique** | **Unique identifier for each faculty member** |
| **2** | **Name** | **VARCHAR** |  |  | **Faculty member's full name** |
| **3** | **Email** | **VARCHAR** |  | **Unique** | **Faculty member's email address** |
| **4** | **Password** | **VARCHAR** |  |  | **Faculty member's password (likely hashed)** |
| **5** | **Department** | **INT** |  | **Foreign Key** | **Department the faculty member belongs to** |
| **6** | **Role** | **VARCHAR** |  |  | **Faculty member's role (defaults to "Faculty")** |

**Teacher Table**

**Student Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **Student id** | **INT** |  | **Unique** | **Unique identifier for each student** |
| **2** | **Name** | **VARCHAR** |  |  | **Student's full name** |
| **3** | **Email** | **VARCHAR** |  | **Unique** | **Student's email address** |
| **4** | **Password** | **VARCHAR** |  |  | **Student's password (likely hashed)** |
| **5** | **Role** | **VARCHAR** |  |  | **Student's role (defaults to "Student")** |
| **6** | **Course** | **VARCHAR** |  |  | **Courses the student is enrolled in (multiple courses possible)** |
| **7** | **Photo** | **VARCHAR** |  | **Optional** | **Path to student's photo** |
| **8** | **Department** | **VARCHAR** |  |  | **Student's department** |

**Department Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **Department id** | **INT** |  | **Unique** | **Unique identifier for each department** |
| **2** | **Name** | **VARCHAR** |  |  | **Department name** |
| **3** | **Description** | **TEXT** |  |  | **Department description** |

**Courses Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **course\_code** | **INT** | **10** | **Primary Key** | **Unique identifier for each course** |
| **2** | **Name** | **VARCHAR** | **255** |  | **Course name** |
| **3** | **department** | **TEXT** |  |  | **Course department** |
| **4** | **teacher\_name** | **VARCHAR** | **255** |  | **Name of the teacher who created the course** |
| **5** | **Student\_key** | **INT** | **10** | **UNIQUE** | **Access Key For Students** |

**Assignment Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **Course code** | **INT** |  | **Foreign Key** | **Code of the course the assignment belongs to** |
| **2** | **Title** | **VARCHAR** |  |  | **Assignment title** |
| **3** | **Description** | **TEXT** |  |  | **Assignment description** |
| **4** | **Deadline Date** | **DATE** |  |  | **Date the assignment is due** |
| **5** | **Deadline Time** | **TIME** |  |  | **Time the assignment is due** |
| **6** | **File** | **VARCHAR** |  | **Optional** | **Path to the assignment file** |
| **7** | **Marks** | **INT** |  |  | **Maximum marks for the assignment** |

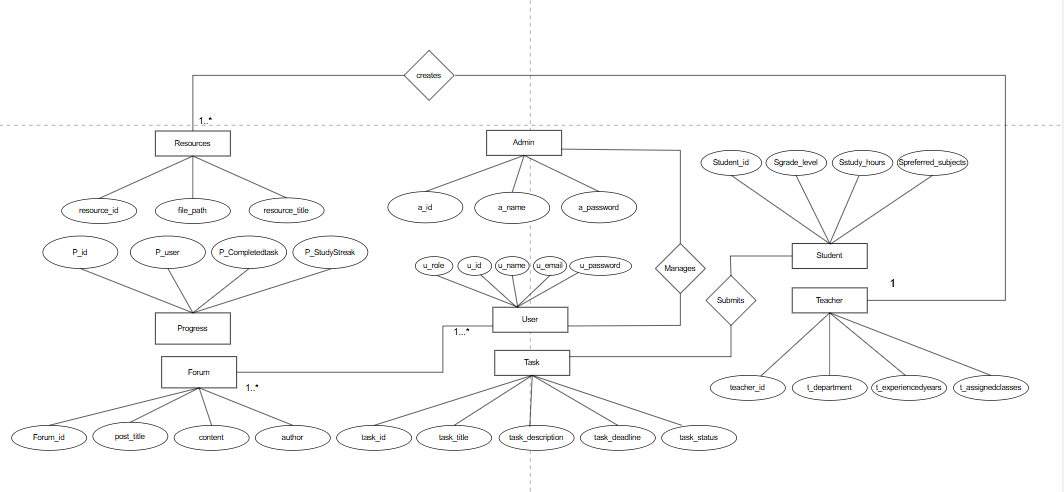
**Attendance Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **Student** | **INT** |  | **Foreign Key** | **ID of the student whose attendance is being recorded** |
| **2** | **Course** | **INT** |  | **Foreign Key** | **ID of the course for which attendance is being recorded** |
| **3** | **Date** | **DATE** |  |  | **Date of the attendance** |
| **4** | **Status** | **VARCHAR** |  |  | **Attendance status (e.g., Present, Absent, Late)** |

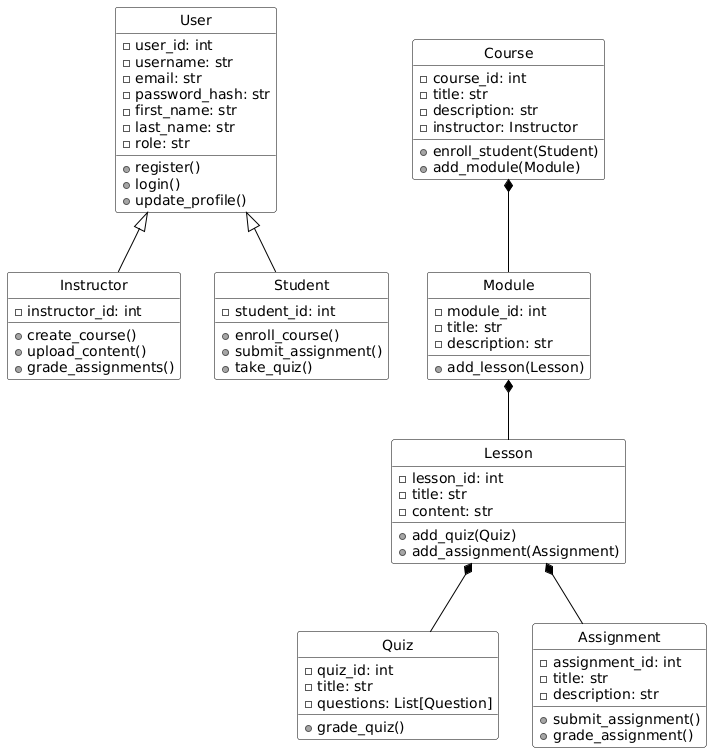
**Quiz Table:-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Fieldname** | **Datatype** | **Fieldsize** | **Constraints** | **Description** |
| **1** | **Title** | **VARCHAR** |  |  | **Quiz title** |
| **2** | **Description** | **TEXT** |  |  | **Quiz description** |
| **3** | **Course** | **INT** |  | **Foreign Key** | **ID of the course the quiz belongs to** |
| **4** | **Start Date** | **DATE** |  |  | **Date the quiz starts** |
| **5** | **Start Time** | **TIME** |  |  | **Time the quiz starts** |
| **6** | **End Date** | **DATE** |  |  | **Date the quiz ends** |
| **7** | **End Time** | **TIME** |  |  | **Time the quiz ends** |
| **8** | **Publish status** | **BOOLEAN** |  |  | **Indicates if the quiz is published (Yes/No or True/False)** |

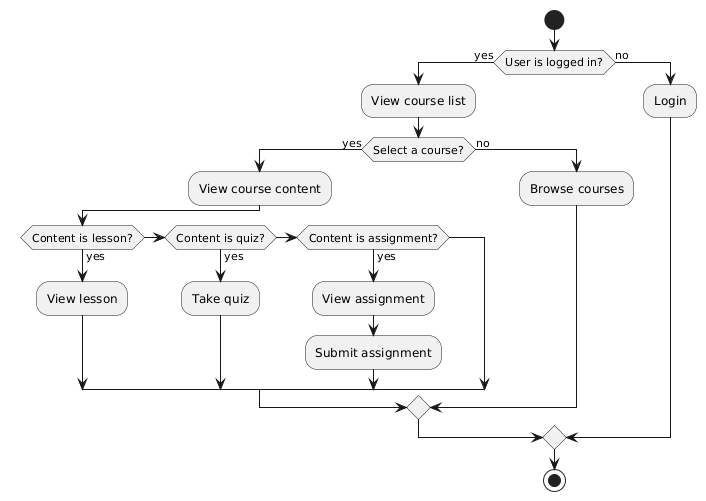
**3.3 ER Diagram:**

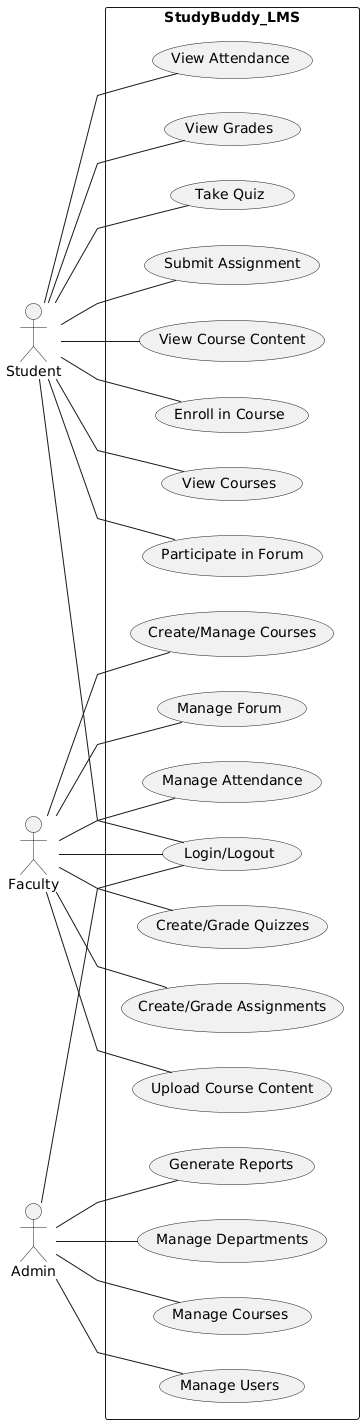


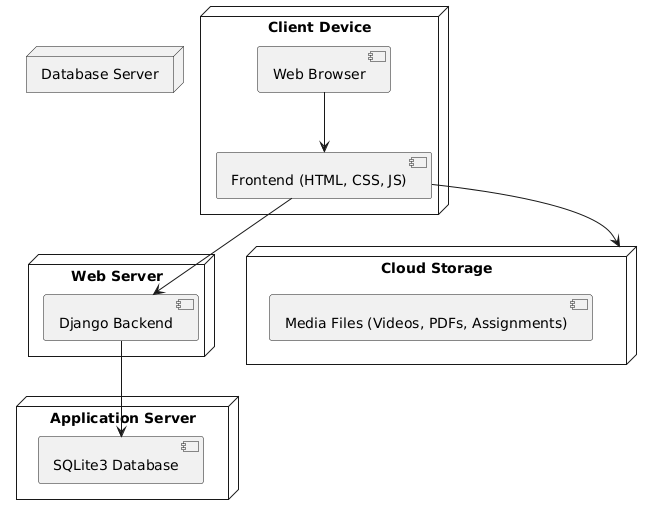
**3.4 Class Diagram:**



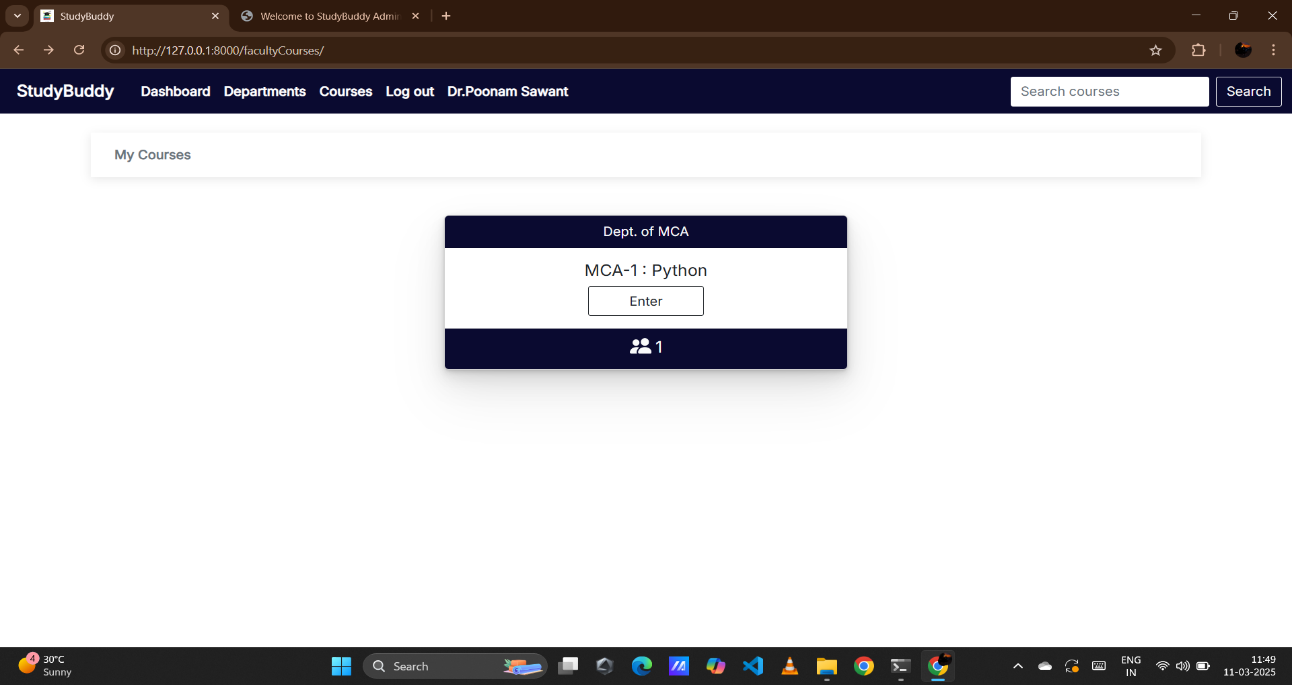
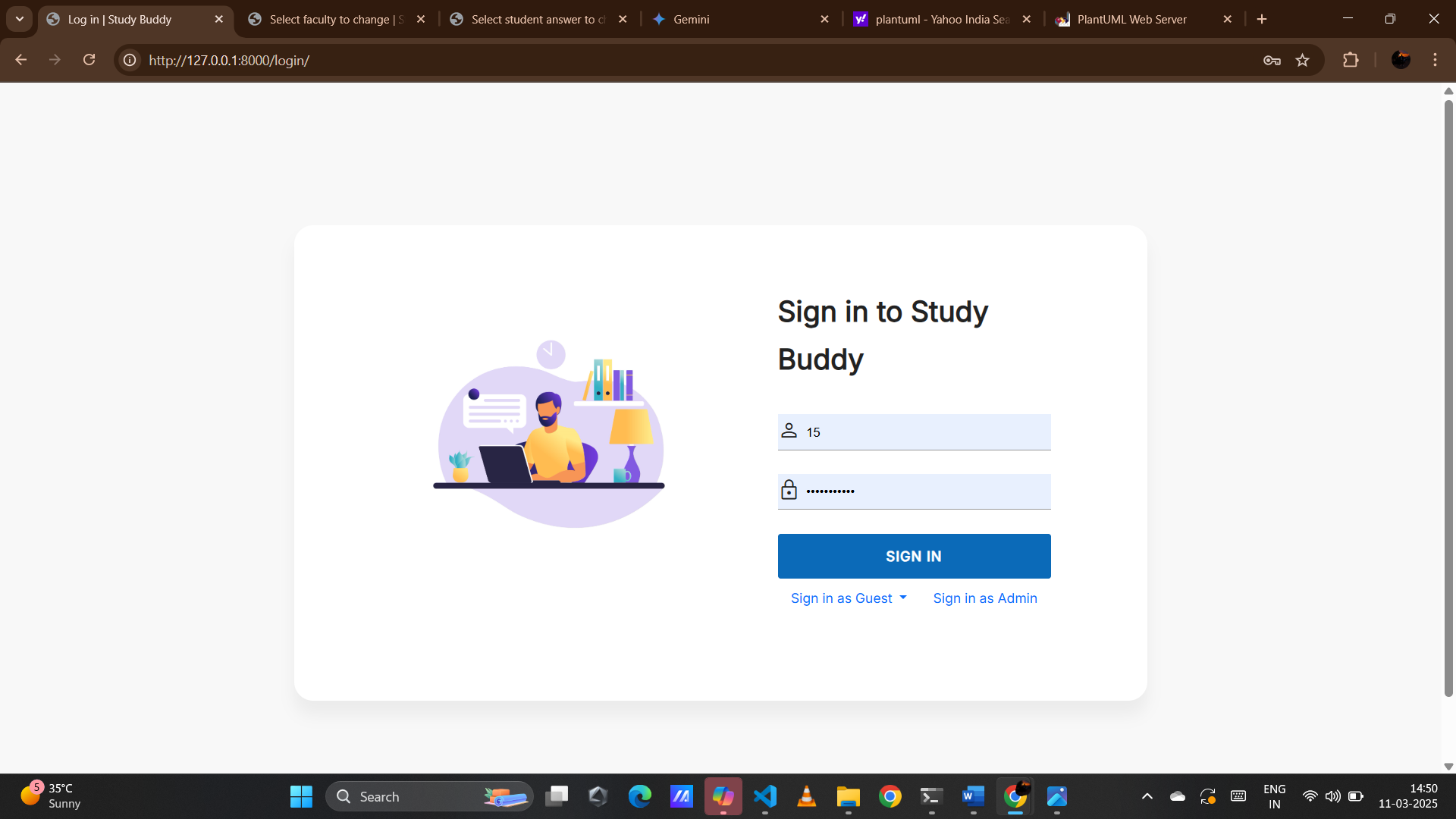
**Activity Diagram:-**

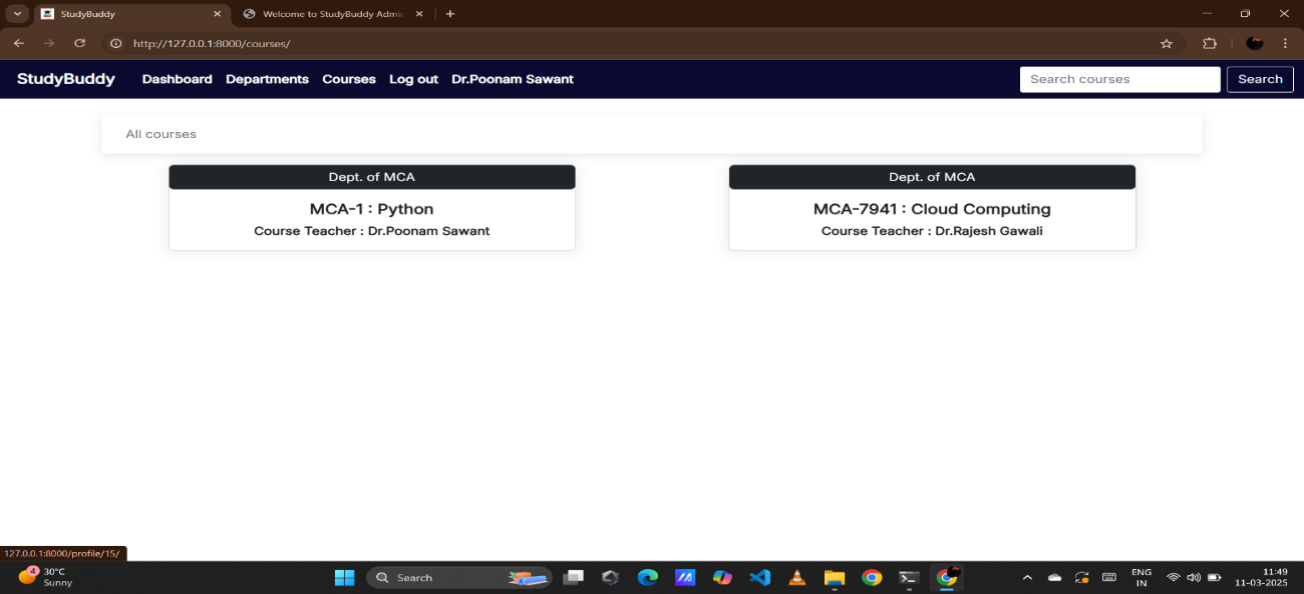


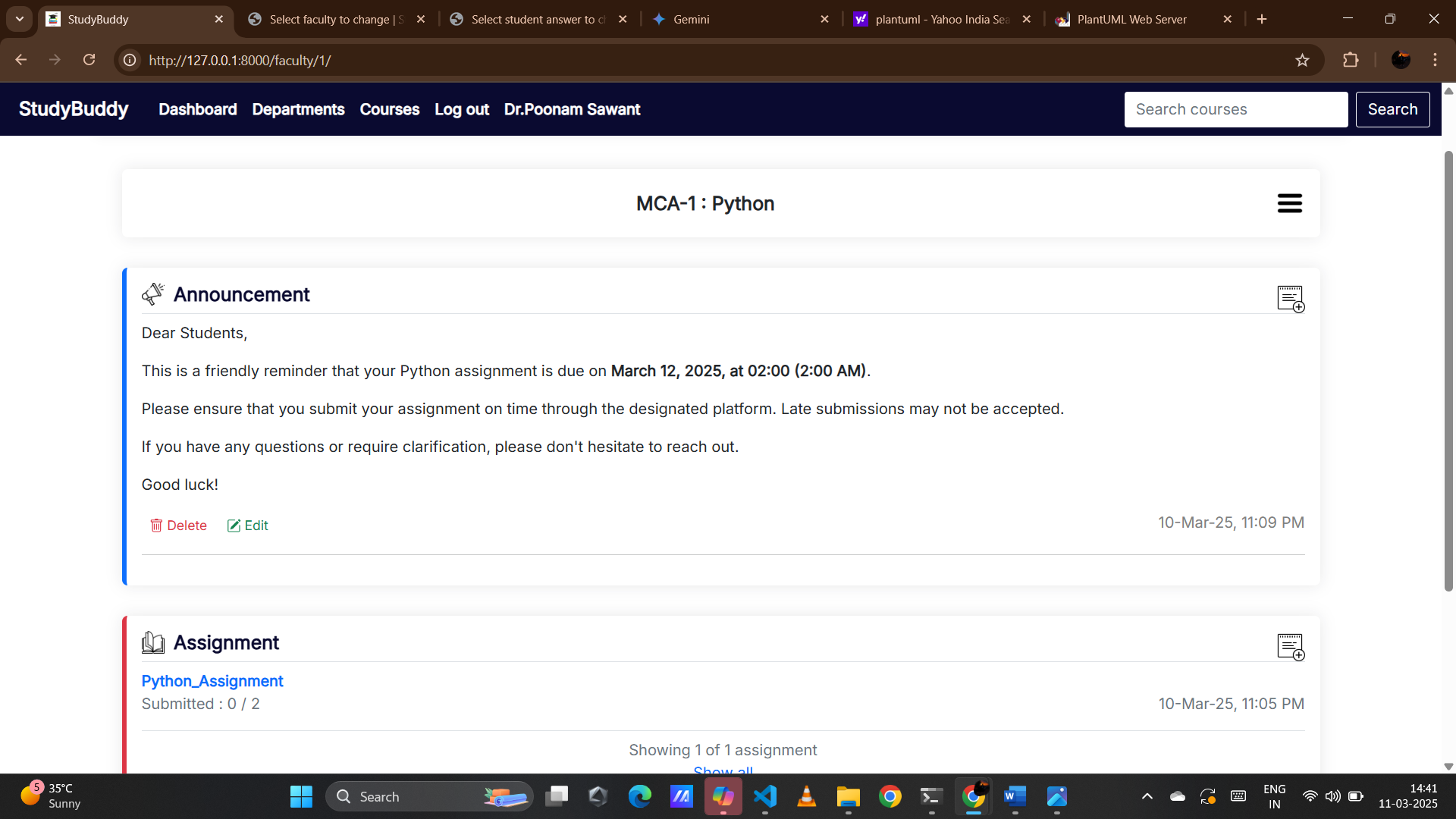
**Use Case Diagram:-  
Deployment Diagram:-**

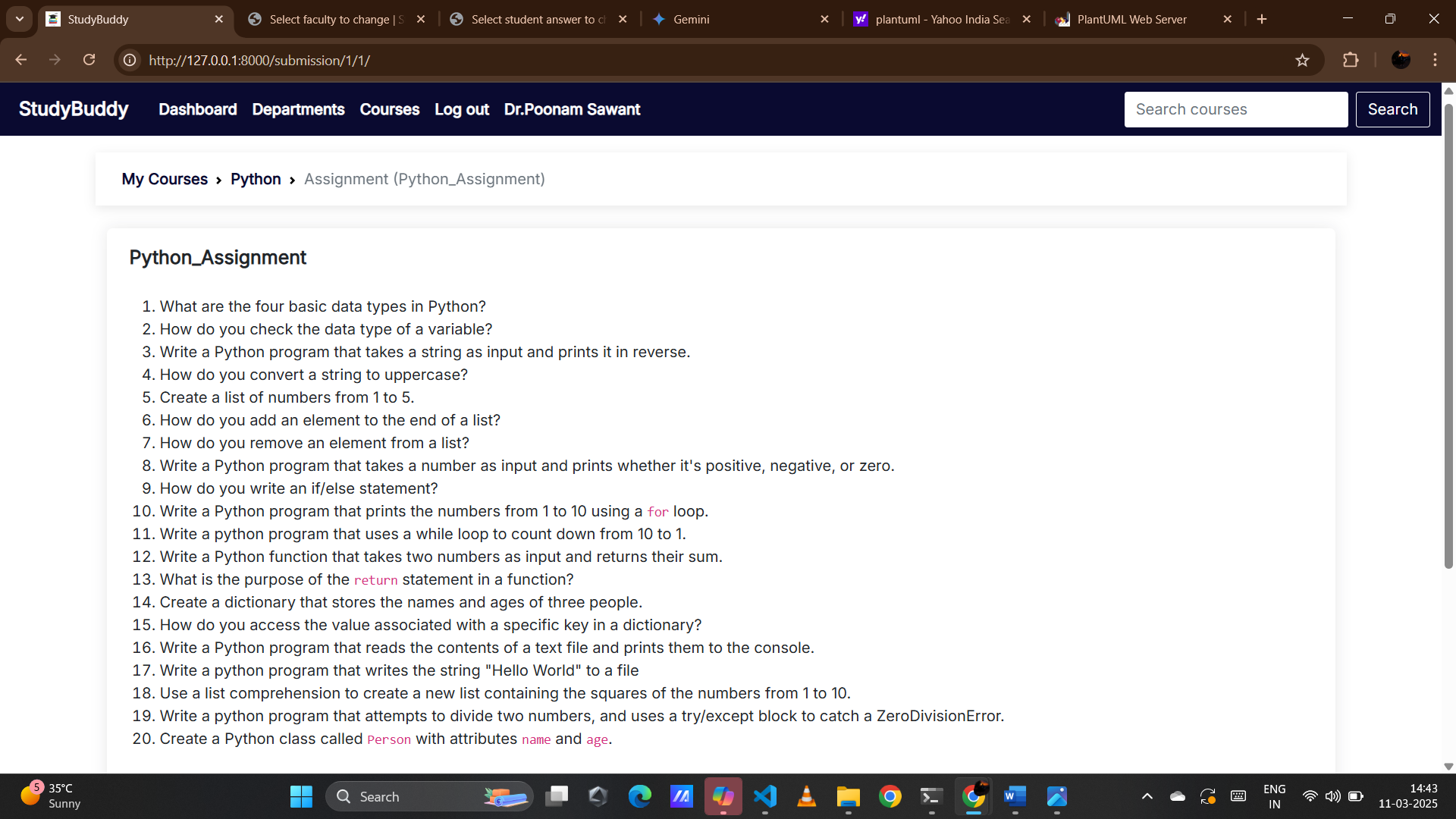


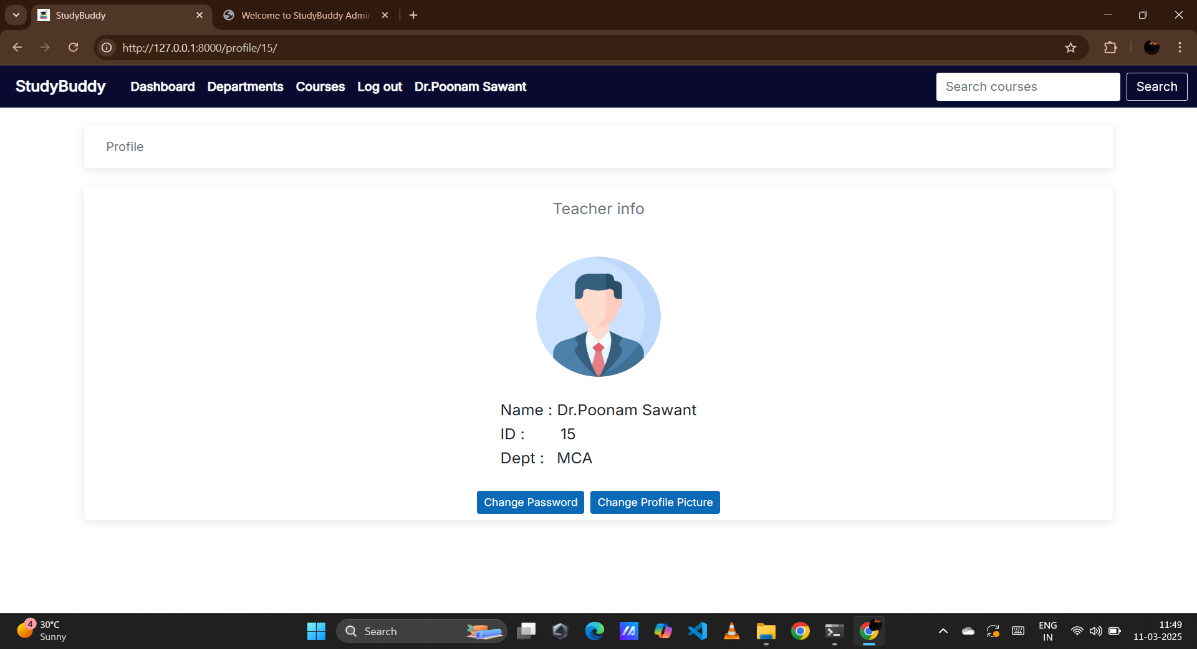
**1.8 Output Screen: Teacher Screens:-**

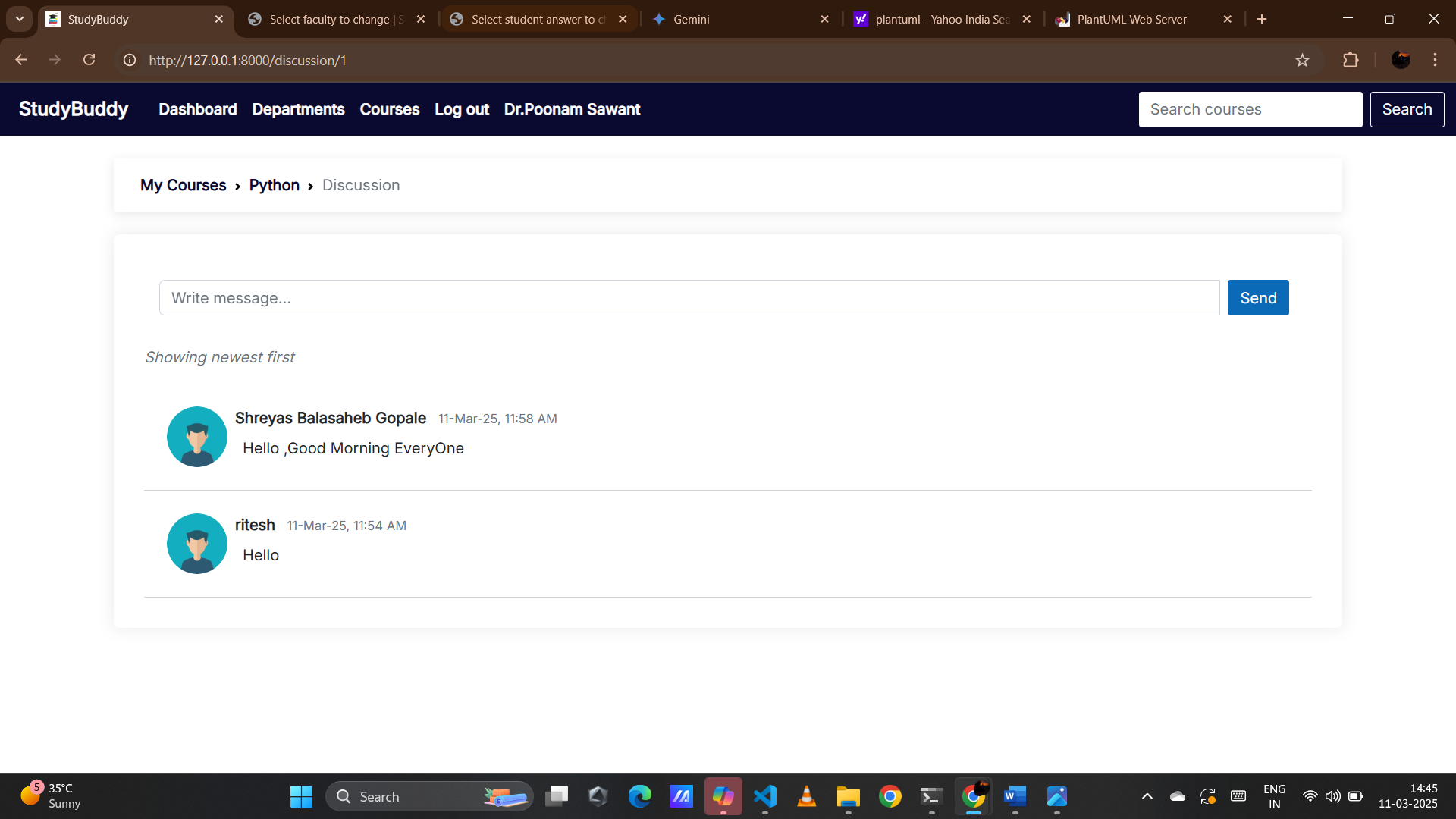
****

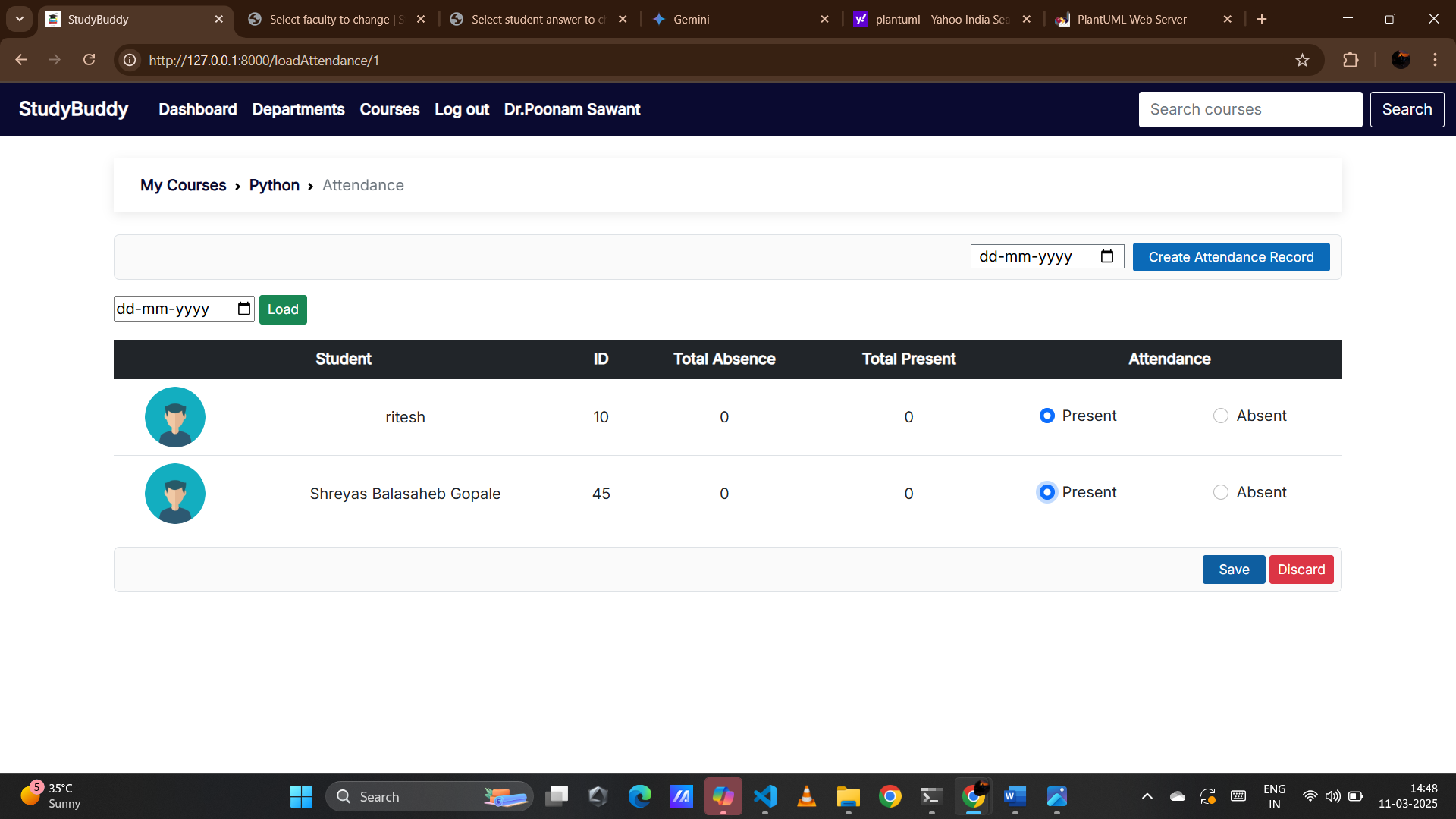
****

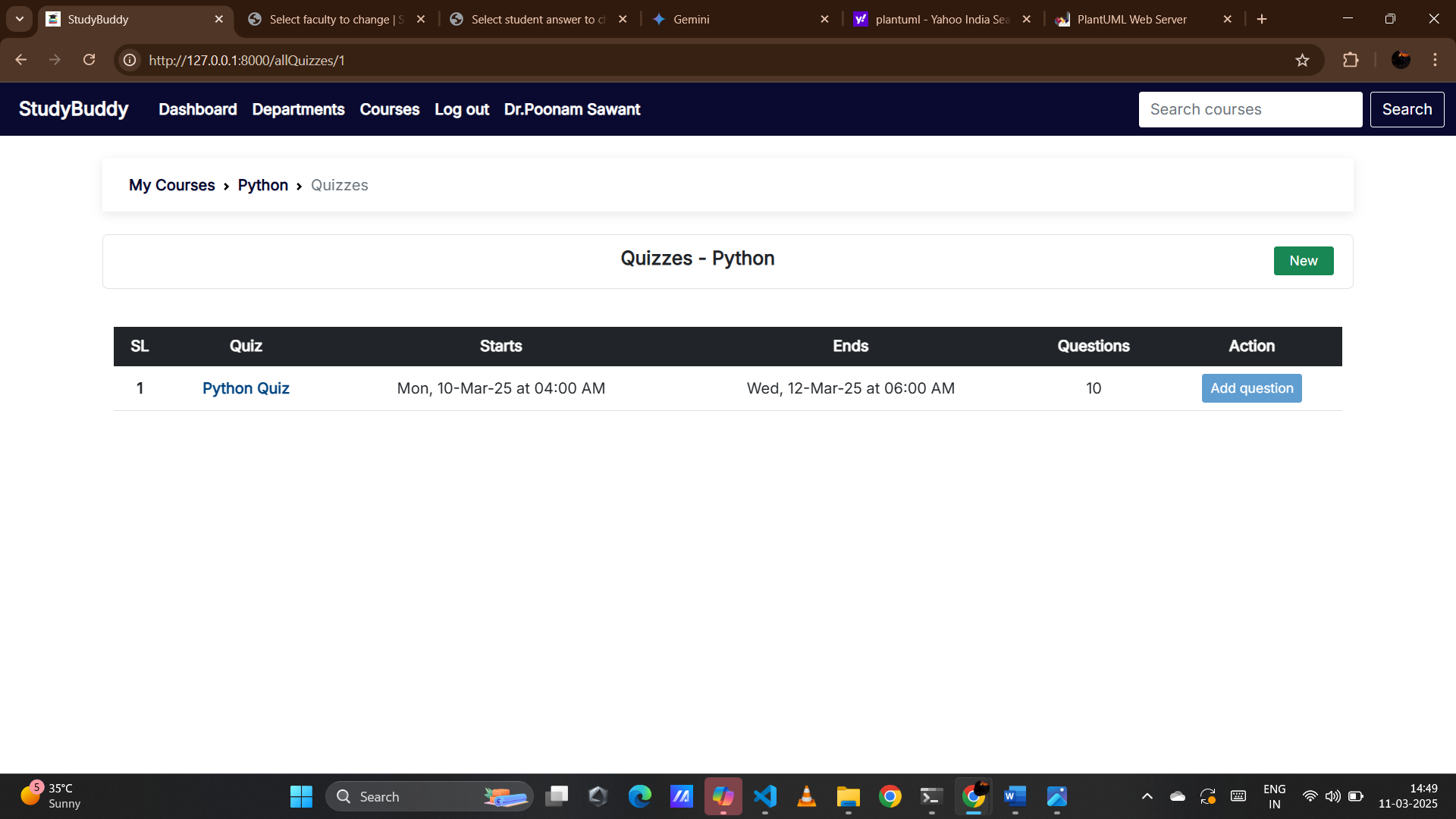
****

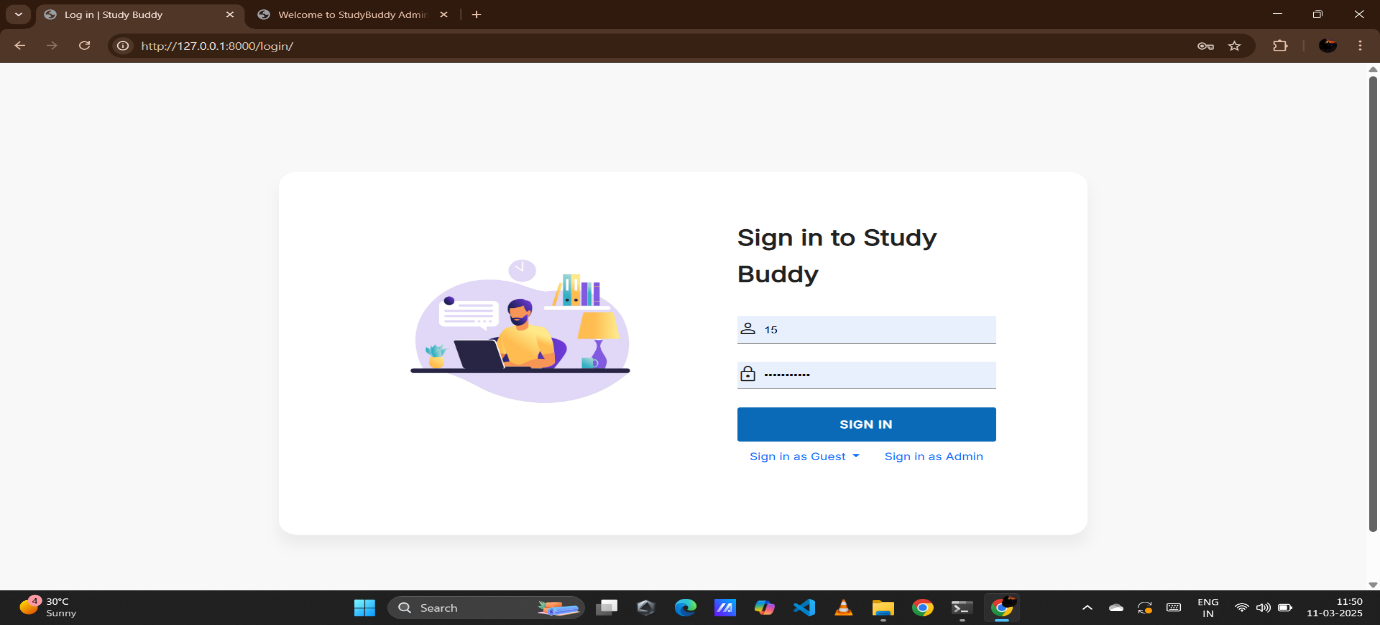
****

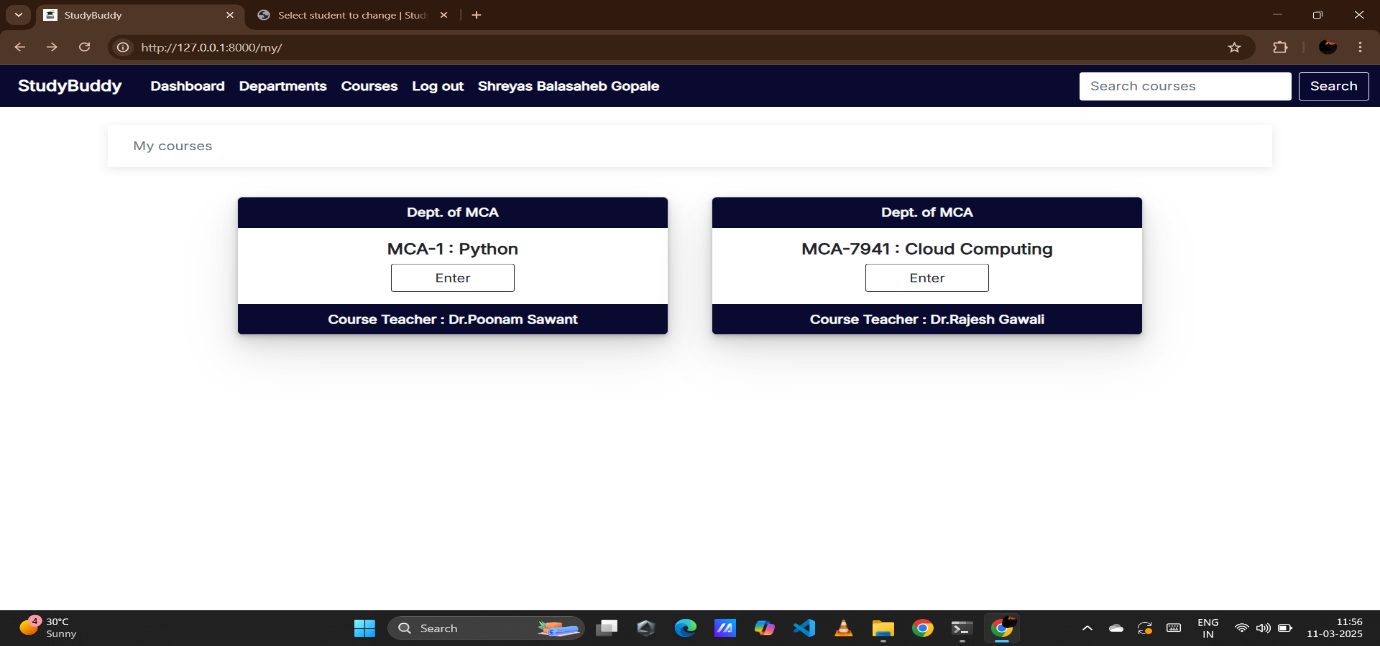
****

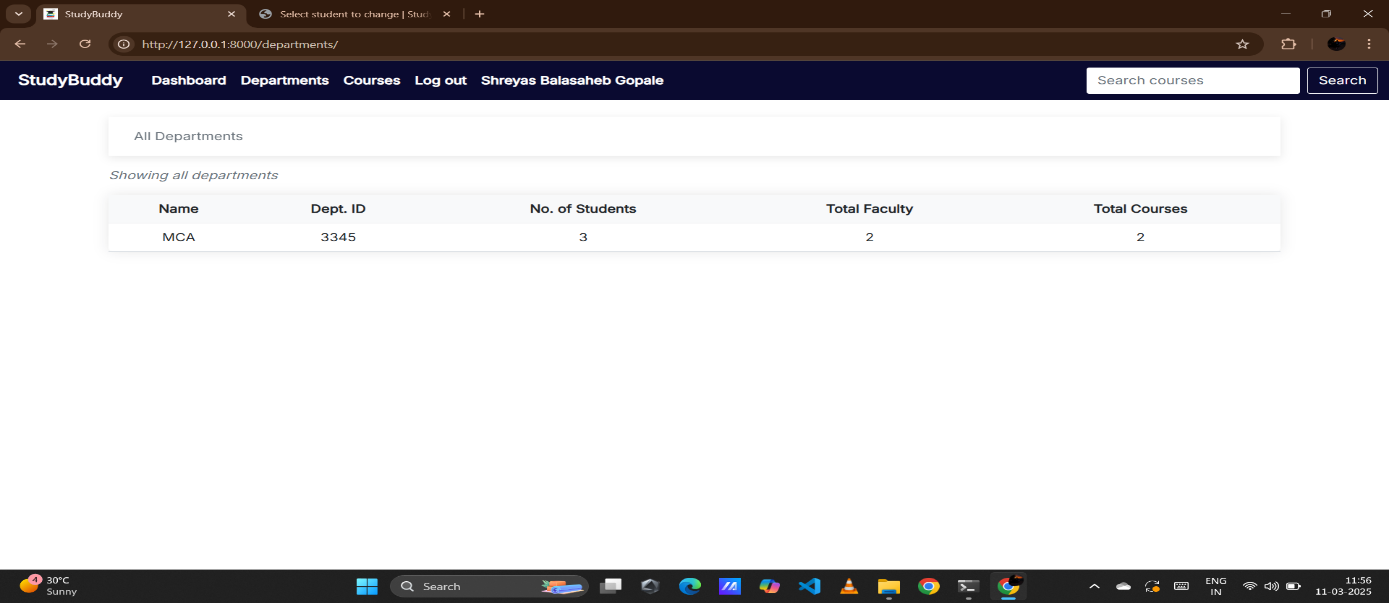
****

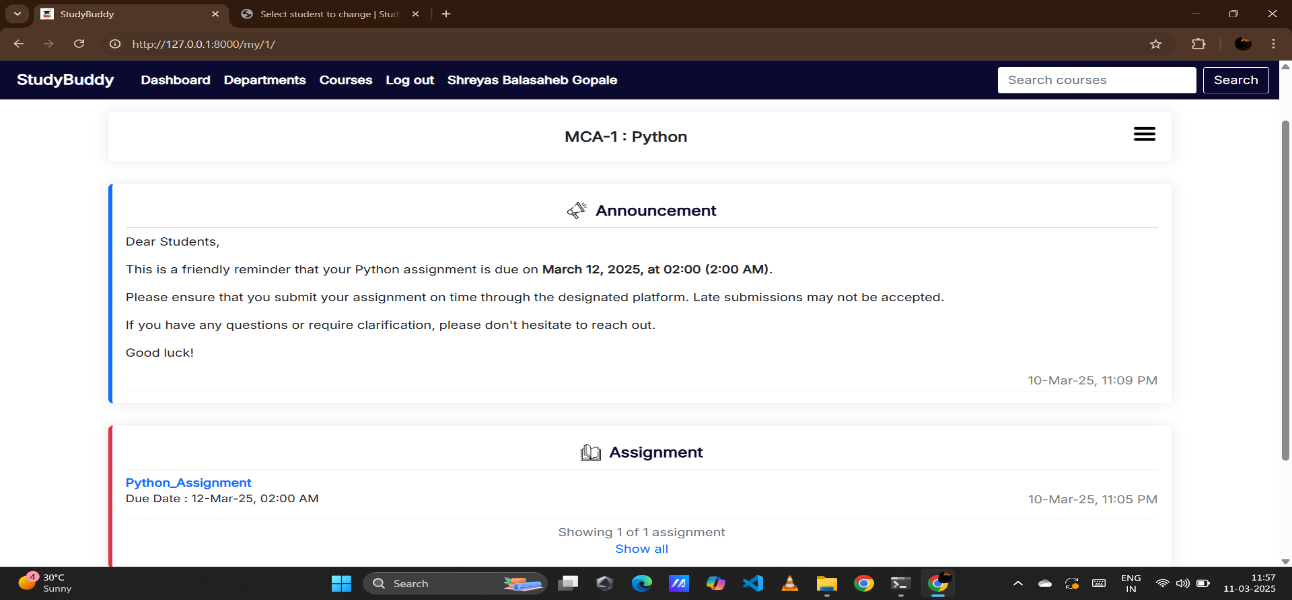
****

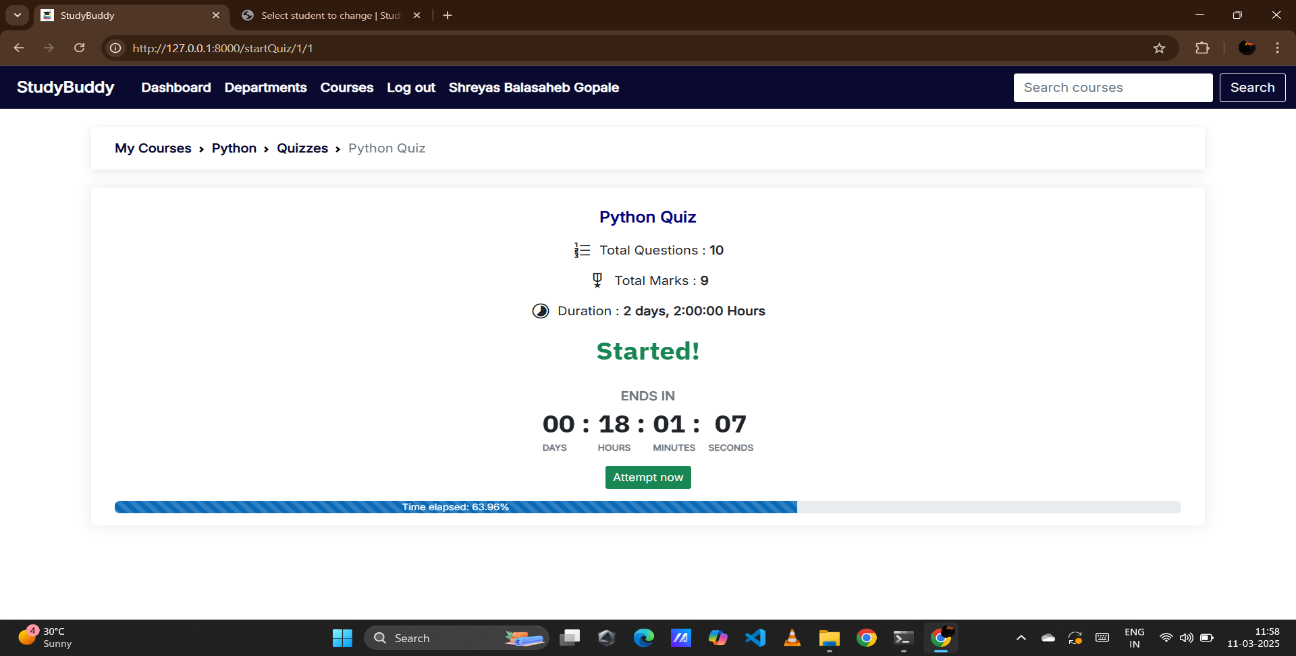
****

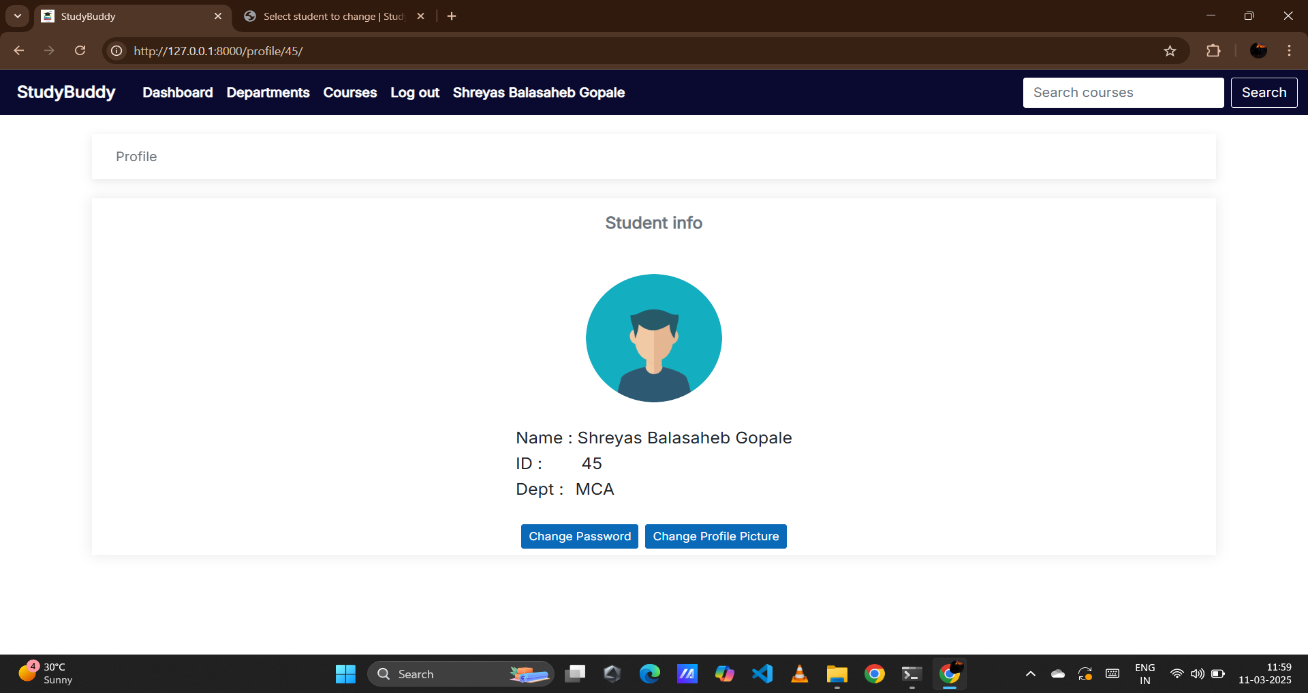
**Student Screens** ****

****

****

****

****

****

**Chapter No. 4: TESTING**

**4.1. TEST STRATEGY**

1. Test Objectives:
   * Ensure the StudyBuddy LMS meets all specified functional and non-functional requirements.
   * Identify and resolve defects to ensure the system's reliability, performance, and security.
2. Test Approach:
   * Implement a comprehensive testing strategy including unit, integration, system, performance, and user acceptance testing.
   * Utilize a combination of manual and automated testing methods to achieve thorough test coverage.
3. Test Environment:
   * Server-side: Linux-based environment (Ubuntu, CentOS), PostgreSQL, Django/DRF setup.
   * Client-side: Modern web browsers (Chrome, Firefox, Safari, Edge) on various operating systems.
4. Test Coverage:
   * Functional: Test all modules, including user management, course management, learning management, communication, and payment processing.
   * Non-functional: Evaluate performance, usability, security, and scalability.
5. Defect Management:
   * Log and categorize defects based on severity (critical, major, minor).
   * Use a defect tracking tool (e.g., Jira, GitHub Issues) to manage and track defect resolution.
6. Test Execution:
   * Execute test cases and scripts, recording pass/fail results.
   * Perform regression testing after bug fixes and updates to ensure system stability.

4.2. UNIT TEST PLAN

1. Objective:
   * Validate that individual components and modules of the StudyBuddy LMS function correctly in isolation.
   * Ensure each module performs as expected before integration with other components.
2. Scope:
   * Covers all critical backend modules (Django views, models, APIs) and frontend components (React components or Django templates).
   * Focuses on testing core functionalities like user authentication, course creation, and data processing.
3. Approach:
   * Use unit testing frameworks (e.g., Python's unittest, pytest; JavaScript's Jest) to automate unit tests.
   * Test inputs, outputs, and error handling for each function and method in isolation.
4. Test Environment:
   * Local development environment with Python and JavaScript testing frameworks.
   * Mocking and stubbing to isolate modules from external dependencies (database, APIs).
5. Test Data:
   * Predefined mock data for users, courses, and other entities.
   * Simulated test cases for both valid and invalid inputs.
6. Test Execution:
   * Execute unit tests as part of the continuous integration (CI) process.
   * Track test results and identify failures during testing.

4.3. INTEGRATION TEST PLAN

1. Objective:
   * Verify that different modules of the StudyBuddy LMS interact correctly with each other.
   * Ensure seamless data flow and communication between integrated components (frontend and backend).
2. Scope:
   * Includes testing the integration of user management, course management, learning management, and payment processing.
   * Covers communication between the frontend (React.js or Django Templates) and backend (Django/DRF).
3. Approach:
   * Perform top-down and bottom-up integration testing to validate module interactions.
   * Use API testing tools (e.g., Postman, curl) to test API endpoints.
4. Test Environment:
   * Development environment with Django/DRF, PostgreSQL, and frontend setup.
   * Test on various web browsers and operating systems.
5. Test Data:
   * Use realistic data for users, courses, lessons, and payments.
   * Validate edge cases where data is incomplete or inconsistent between modules.
6. Test Execution:
   * Run integration tests after unit tests to ensure module connections work properly.
   * Execute test cases covering interactions between all modules.
7. Defect Management:
   * Log defects related to data flow or communication issues between modules.
   * Use an issue tracking tool to prioritize and resolve integration defects.

4.4. PERFORMANCE TEST PLAN

1. Objective:
   * Assess the StudyBuddy LMS's responsiveness, stability, and scalability under various loads.
   * Ensure the system can handle expected and peak user traffic efficiently.
2. Scope:
   * Focuses on testing critical areas like course browsing, user registration, and payment processing.
   * Includes load testing and stress testing to evaluate system limits.
3. Approach:
   * Use performance testing tools (e.g., JMeter, Locust) to simulate user load.
   * Measure response times, resource utilization (CPU, memory), and transaction throughput under different load conditions.
4. Test Environment:
   * Staging or production-like environment with Django/DRF, PostgreSQL, and frontend setup.
   * Use real devices and browsers to capture accurate performance metrics.
5. Test Data:
   * Use large datasets including course listings, user profiles, and payment histories.
   * Simulate high user activity with concurrent login sessions, course browsing, and payment transactions.
6. Test Execution:
   * Conduct baseline tests with normal load, followed by load and stress tests.
   * Record performance metrics like response times, throughput, and error rates.

**Chapter No 5. LIMITATION OF PROPOSED SYSTEM**

**1.Limited User Roles:** The current version of Study Buddy supports three primary user roles: Administrator, Instructor, and Student. This may limit flexibility in the future if additional roles, such as teaching assistants, moderators, or guest users, are required.

**2.Reliance on Third-Party Integrations:** Study Buddy relies on third-party integrations for live classes (Zoom, Google Meet) and payment processing (Stripe, PayPal). Any limitations or disruptions in these third-party services could affect the functionality of the LMS.

**3.AI Capabilities:** The effectiveness of the AI-powered features (course recommendations, chatbot) depends on the quality of data and algorithms. These features may have limitations in handling complex or nuanced queries, and their accuracy may improve over time with more data and refinement.

**4.Content Dependency:** The success of Study Buddy relies heavily on the quality and engagement of the courses created by instructors. The platform itself cannot guarantee effective learning outcomes if the content is inadequate.

**5.Potential Scalability Challenges:** While designed for scalability, handling extremely high user traffic and data volumes might require further infrastructure optimization and investment.

**6.Limited Offline Access:** Study Buddy primarily functions as an online platform. Offline access to learning materials may be limited, potentially affecting users with unreliable internet connections.

**7.Accessibility Features:** While basic accessibility considerations are incorporated, further enhancements may be needed to cater to users with specific disabilities. **8.No Mobile Application:** Currently, Study Buddy is designed as a web application. A dedicated mobile application could enhance user experience and accessibility on mobile devices.

**Chapter No. 6: PROPOSED ENHANCEMENTS LMS**

1. **Expanded User Roles:** Introduce new user roles, such as Teaching Assistant, Moderator, and Guest User, to provide more flexibility and control over user permissions and access within the platform.
2. **Enhanced AI Capabilities:** Continuously improve the AI-powered features by refining algorithms, expanding the knowledge base of the chatbot, and incorporating more sophisticated machine learning models for personalized recommendations.
3. **Offline Access**: Develop offline access capabilities for learning materials, allowing users to download content and access it even without an internet connection. This will improve accessibility for users with limited or unreliable internet access.
4. **Advanced Gamification:** Implement more advanced gamification features, such as points, rewards, challenges, and interactive elements, to further enhance user engagement and motivation.
5. **Interactive Content Creation Tools:** Provide instructors with more interactive content creation tools, such as simulations, interactive exercises, and branching scenarios, to create more engaging and effective learning experiences.
6. **Integration with Learning Analytics Dashboards:** Develop dashboards that provide instructors with detailed insights into student performance, learning patterns, and engagement metrics. This will help them tailor their teaching strategies and improve learning outcomes.
7. **Enhanced Accessibility Features:** Conduct thorough accessibility audits and implement features to cater to users with diverse needs, such as screen reader compatibility, keyboard navigation, and adjustable font sizes.
8. **Mobile Application Development:** Develop dedicated mobile applications for iOS and Android platforms to enhance user experience and accessibility on mobile devices.

**Chapter No 7. CONCLUSION**

The Study Buddy Learning Management System (LMS) is poised to be a comprehensive and robust platform designed to enhance the online learning experience for both instructors and students. The system's proposed features, such as user-friendly course management, interactive learning tools, AI-powered assistance, and gamified elements, address the limitations of existing LMS solutions, creating a more engaging and personalized learning environment. The technical feasibility of the project is supported by the utilization of a well-established technology stack, including Django, React.js (or Django Templates), and PostgreSQL, ensuring scalability and reliability. Economic feasibility is evident in the growing demand for online learning platforms, with Study Buddy offering monetization opportunities for instructors and a potentially sustainable business model. Operational feasibility is ensured through a well-defined operational plan, ongoing support, and continuous improvement based on user feedback. While limitations such as reliance on third-party integrations and the need for high-quality content exist, proactive mitigation strategies can address these challenges effectively. Ultimately, Study Buddy has the potential to revolutionize online education by providing a dynamic, accessible, and effective platform for learners and educators alike.

**Chapter No 8. BIBLIOGRAPHY**

• Teachers Guide

• www.google.com

• www.youtube.com