

CAMPUSMEET

ABSTRACT

CampusMeet is a secure, web-based educational platform designed to streamline online teaching, learning, and meeting management for academic institutions. Developed with the Django framework and modern web technologies, *CampusMeet* empowers faculty and students to participate in real-time classes, track attendance automatically, and interact through dedicated dashboards and messaging features. The platform leverages ZegoCloud's robust video conferencing technology, enabling seamless, high-quality audio and video communication for online meetings and classroom sessions. Its core innovation lies in the intelligent attendance system, which records detailed join and leave statistics and generates comprehensive participation reports in sync with real-time session activity powered by ZegoCloud. Role-based authentication ensures that each user accesses functions suited to their academic role, enabling faculty to create and manage meetings, upload notes, and monitor student engagement, while students can efficiently track sessions, download resources, and interact within their departments. A centralized admin interface offers robust control over user management, departmental settings, and system monitoring, ensuring secure operational oversight. By integrating optimized UI components with scalable backend functionality and advanced real-time video services, *CampusMeet* reduces administrative overhead, encourages active learning, and provides a reliable, scalable solution for the specific needs of educational organizations. Its modular architecture, clear separation of user roles, and persistent resource management make *CampusMeet* a future-ready platform for progressive academic environments.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

CampusMeet is a Django-based virtual classroom platform that enables real-time online teaching and learning for educational institutions. Powered by ZegoCloud's video conferencing technology, it offers live classes, automated attendance tracking, and dedicated dashboards for faculty and students. The platform streamlines meeting management, resource sharing, and student engagement, providing a secure and scalable solution tailored to modern academic needs. With role-based access and interactive features, *CampusMeet* enhances educational accountability and connectivity in a remote learning environment.

1.2 PROBLEM STATEMENT

In modern education, effective management of virtual classrooms and student engagement is crucial yet challenging. Traditional online learning tools often lack seamless integration of real-time communication, attendance tracking, and resource management, leading to disconnected learning experiences and administrative overhead. *CampusMeet* addresses these key challenges:

- **Inefficient Attendance Tracking:** Many platforms rely on manual or inaccurate attendance recording, which can lead to unreliable participation data and accountability issues.
- **Fragmented Communication:** Lack of integrated, real-time video and messaging tools hinders effective interaction between faculty and students.
- **Resource Accessibility:** Difficulties in distributing and managing educational resources such as notes and recordings reduce learning efficiency.
- **Role-Based Management:** Existing tools often do not offer clear segregation of functionalities for faculty, students, and administrators, complicating user experience and security.
- **Scalability and Integration:** Many systems struggle to scale or integrate with modern video conferencing solutions and departmental workflows.

CampusMeet is designed to overcome these issues by providing an integrated, scalable platform powered by Django and ZegoCloud video technology. It automates attendance based on actual session activity, facilitates real-time communication, manages academic resources securely, and offers role-based access tailored to educational needs. This approach improves learning engagement, reduces administrative burden, and supports the growing demands of digital education.

1.3 PROJECT SCOPE AND PROJECT RELEVANCE

Project Scope

CampusMeet is designed as a comprehensive virtual classroom platform tailored to meet the needs of educational institutions. The project scope includes:

1. Virtual Classroom Management:
 - Facilitating the creation and scheduling of online classes and meetings.
 - Enabling real-time video communication through integration with ZegoCloud technology.
 - Providing seamless joining and leaving alerts to track session participation.
2. Automated Attendance System:
 - Tracking student attendance based on actual participation with automatic logging.
 - Generating detailed attendance reports accessible to faculty and administrators.
3. User Authentication and Role-Based Access:
 - Implementing secure login for faculty, students, and administrators.
 - Ensuring access control based on user roles to safeguard sensitive data.
4. Dashboards and Notifications:
 - Offering tailored dashboards for faculty and students displaying schedules, notes, and attendance.
 - Sending real-time alerts and notifications for meeting events and attendance status.
5. Resource and Note Management:
 - Allowing faculty to upload and distribute notes and study materials securely.
 - Enabling students to download resources and track their academic materials.
6. Scalability and Integration:
 - Designing the system to support increasing numbers of users and sessions.
 - Ensuring compatibility with third-party tools and future technology upgrades.

Project Relevance

CampusMeet addresses key challenges in contemporary education, particularly in remote and hybrid learning scenarios:

1. Enhanced Learning Engagement:

- By integrating high-quality real-time video conferencing and interactive dashboards, *CampusMeet* promotes active participation and collaboration.

2. Reliable and Accurate Attendance Tracking:

- Automated attendance ensures accountability while reducing manual overhead for faculty.

3. Streamlined Academic Management:

- Centralized resource sharing and user management reduce administrative burden and enhance operational efficiency.

4. Secure and Role-Based Access:

- Ensures that users interact with data appropriate to their roles, strengthening security and privacy.

5. Future-Ready Scalability:

- The platform is built to evolve with growing institutional needs and technological advancements.

CampusMeet offers a unified, efficient, and secure solution that enhances the experience of digital education for both faculty and students, aligning with the demands of modern academic institutions.

1.4 OBJECTIVES OF THE PROPOSED SYSTEM

The objectives of the *CampusMeet* virtual classroom platform are to:

1. **Enable Seamless Online Learning:**
 - Provide real-time video conferencing and interactive classroom features powered by ZegoCloud technology for effective virtual teaching and learning.
2. **Automate Attendance Tracking:**
 - Track student attendance automatically based on real-time participation during online sessions, producing accurate attendance reports.
3. **Ensure Secure Role-Based Access:**
 - Implement secure login and authorization mechanisms to control access to features based on user roles (faculty, student, admin).
4. **Streamline Academic Resource Management:**
 - Facilitate easy upload, sharing, and downloading of study notes and materials between faculty and students.
5. **Deliver Personalized Dashboards and Notifications:**
 - Provide tailored dashboards and real-time alerts to keep users informed about schedules, attendance, and meeting events.
6. **Support Scalable and Reliable Operations:**
 - Build the platform to handle increasing users and sessions, ensuring smooth performance and integration with third-party tools.

CHAPTER 2

SYSTEM ANALYSIS

2.1 INTRODUCTION

System analysis is a critical phase in the development of *CampusMeet*, a web-based virtual classroom platform designed to enhance online education management for academic institutions. This phase involves a detailed examination of existing online learning systems and educational practices to identify functional requirements and challenges. It also assesses the feasibility of implementing an integrated solution that combines real-time video communication, automated attendance tracking, resource management, and role-based access control. The aim is to ensure that the system effectively meets user needs—both faculty and students—while operating efficiently within technological and operational constraints. Through comprehensive analysis, *CampusMeet* seeks to deliver a scalable, user-friendly platform that improves learning engagement, streamlines administrative tasks, and supports secure, interactive virtual classrooms.

2.2 EXISTING SYSTEM

Current virtual learning platforms and online classroom tools often operate as fragmented solutions, relying on separate video conferencing, attendance tracking, and resource management systems. Many educational institutions still depend on manual attendance recording or disconnected digital records, which lead to inaccuracies and administrative burden. Communication between faculty and students is frequently limited to basic messaging or email without integration into the learning workflow. Additionally, user management and role-based access controls are either simplistic or absent, resulting in inconsistent experiences and security gaps. Resource sharing and note distribution processes are often cumbersome, lacking centralized organization. These limitations reduce engagement, create inefficiencies, and hinder the scalability of digital education, highlighting the need for an integrated and seamless platform like *CampusMeet*.

2.2.1 Limitations of Existing System

Current virtual classroom and online learning platforms face several limitations:

- **Fragmented Tools:** Many platforms rely on separate, disconnected tools for video conferencing, attendance tracking, and resource sharing, creating a disjointed experience for users.
- **Manual Attendance Processes:** Numerous systems require manual effort to mark attendance, leading to inaccuracies and increased administrative workload.
- **Limited Customization and Role Management:** Many platforms offer limited support for distinct user roles, restricting tailored experiences for faculty, students, and administrators.
- **Inadequate Real-time Interaction:** Existing tools often lack seamless, high-quality real-time communication features integrated with other classroom functions.
- **Complex User Interfaces:** Users may face challenges navigating multiple systems or overly complex interfaces, hindering adoption.

- **Scalability and Performance Issues:** Some platforms struggle to handle large numbers of concurrent users effectively, causing lag or disconnections.
- **Security and Privacy Gaps:** Insufficient data protection measures may expose sensitive user information or academic content to unauthorized access.

2.3 PROPOSED SYSTEM

The proposed *CampusMeet* platform aims to overcome the limitations of existing virtual learning solutions by providing an integrated, user-friendly system that combines live video conferencing, automated attendance tracking, and seamless resource management. Leveraging robust technologies like Django and ZegoCloud for real-time communication, *CampusMeet* offers tailored dashboards for faculty and students, secure role-based access, and comprehensive reporting features. This unified approach enhances educational engagement, streamlines administrative tasks, and supports scalability to meet the evolving demands of academic institutions.

2.3.1 Advantages of Proposed System

CampusMeet offers key advantages over existing educational platforms:

- **Integrated Learning Environment:** Combines live video conferencing, attendance tracking, resource sharing, and user management into a unified platform, enhancing operational efficiency.
- **Enhanced Communication:** Leverages ZegoCloud technology for high-quality, seamless real-time audio and video interactions, promoting active engagement.
- **Automated Attendance System:** Provides precise and automatic attendance recording, reducing manual effort and improving accountability.
- **Role-Based Access Control:** Implements secure authentication mechanisms to ensure users access functionalities appropriate to their roles.
- **User-Friendly Interfaces:** Features intuitive dashboards tailored separately for faculty, students, and administrators, facilitating easy navigation and use.
- **Comprehensive Reporting:** Generates detailed participation and session reports, supporting informed decision-making and academic tracking.
- **Scalable Design:** Built to support growing user bases and increased session volumes without compromising performance.

2.4 FEASIBILITY STUDY

The feasibility study for *CampusMeet* evaluates its viability from multiple perspectives to ensure it can be effectively developed and deployed within existing constraints. This

includes assessing technical infrastructure, user readiness, economic costs and benefits, as well as data security and compliance considerations.

2.4.1 Technical Feasibility

Technical feasibility assesses the suitability of the technology stack and infrastructure for *CampusMeet*:

- **Technology Stack:** The Django framework, combined with standard web technologies including HTML, CSS, and JavaScript, along with ZegoCloud for real-time video conferencing, provides a robust and flexible solution meeting the functional requirements for virtual classrooms, attendance tracking, and resource sharing.
- **Infrastructure Needs:** Evaluation of server capabilities, broadband internet access, and compatibility with existing educational institution IT environments.
- **Scalability:** Analysis of the platform's ability to support growing numbers of concurrent users and virtual sessions while maintaining performance and reliability.

2.4.2 Operational Feasibility

Operational feasibility examines how well *CampusMeet* aligns with existing educational workflows:

- **User Adoption:** Analysis of ease of adoption and acceptance among faculty, students, and administrators.
- **Training and Support:** Identification of necessary training sessions and ongoing support resources to ensure effective use of the platform.
- **Operational Impact:** Assessment of how *CampusMeet* improves virtual teaching and learning processes while minimizing disruptions to current academic practices.

2.4.3 Economic Feasibility

Economic feasibility evaluates the financial requirements for developing and maintaining *CampusMeet*:

- **Cost Estimates:** Projection of expenses for software development, deployment on hosting platforms, ongoing maintenance, and personnel involved in support and updates.
- **Cost-Benefit Analysis:** Comparison of these projected costs against anticipated benefits, such as improved operational efficiency, reduced administrative workload, and enhanced learning outcomes.
- **Return on Investment:** Analysis of the platform's financial viability, focusing on potential savings from process automation and improved student engagement, justifying the investment.

2.4.4 Legal Feasibility

Legal feasibility ensures *CampusMeet* complies with relevant laws and regulations:

- **Data Privacy and Protection:** Adheres to data protection laws such as GDPR and local privacy regulations to safeguard personal information of users, including students and faculty.
- **Intellectual Property:** Ensures proper licensing and protection of educational content and software components to avoid copyright infringements.
- **Accessibility Standards:** Complies with web accessibility guidelines (e.g., WCAG) to provide inclusive access for users with disabilities, supporting equitable learning opportunities for all.

2.5 SOFTWARE ENGINEERING PARADIGM APPLIED

The Agile Software Development paradigm is applied to the *CampusMeet* project to facilitate flexible, iterative development and continuous improvement:

- **Iterative Development:** The platform is developed in sprints, allowing frequent feedback from stakeholders and making adjustments to meet evolving academic needs.
- **User Stories:** Requirements are gathered through user stories representing the needs of faculty, students, and administrators, enabling prioritized and focused development.
- **Continuous Testing:** Ongoing testing and integration are conducted throughout development to ensure quality, reliability, and prompt resolution of issues.
- **Collaboration:** Close collaboration among developers, educational stakeholders, and users ensures that *CampusMeet's* features align well with operational workflows and user expectations.

CHAPTER 3

SYSTEM ENVIRONMENT

3.1 INTRODUCTION

CampusMeet is a web-based virtual classroom platform designed to provide educational institutions with an efficient, secure, and user-friendly environment for online teaching and learning. The system is built using the Django framework, with frontend technologies based on HTML, CSS, and JavaScript, and integrated with ZegoCloud for real-time video conferencing. This technology stack offers flexibility, scalability, and reliability to support a seamless virtual classroom experience. The system environment requires specific software, hardware, and network configurations to ensure smooth operation, optimal performance, and ease of maintenance for institutions adopting the platform.

3.2 SOFTWARE REQUIREMENTS

The software requirements for *CampusMeet* encompass all necessary development frameworks, runtime environments, and libraries to support backend and frontend functionalities.

- **Operating System:** Compatible with Windows, macOS, or Linux for development and deployment environments.
- **Backend:**
 - Python (programming language)
 - Django (backend web framework)
 - Database management system (e.g., PostgreSQL or SQLite for data persistence)
- **Frontend:**
 - HTML5, CSS3, and JavaScript (for building user interfaces)
 - Optional CSS framework (e.g., Bootstrap) for responsive and consistent design
- **Real-Time Communication:**
 - Integration with ZegoCloud SDK for live video conferencing and audio streaming
- **Additional Libraries and Tools:**
 - Django Authentication and Authorization modules for user management
 - JavaScript libraries or APIs for handling asynchronous requests (AJAX, Fetch API)
 - Image handling and file upload libraries as needed
- **Development Tools:**
 - Visual Studio Code or other suitable IDEs
 - Version Control System (Git) for code collaboration and repository management

3.3 HARDWARE REQUIREMENTS

The hardware requirements for *CampusMeet* address both server-side and client-side needs to ensure smooth operation and optimal user experience.

Server Requirements:

- **Processor:** Quad-core CPU with a minimum clock speed of 2.5 GHz.
- **RAM:** At least 8 GB, with 16 GB recommended for handling larger user loads and concurrent sessions.
- **Storage:** Solid State Drive (SSD) with at least 100 GB of free space for application files, database storage, and media content.
- **Network:** High-speed, reliable internet connectivity with sufficient bandwidth to support multiple simultaneous video conferencing sessions.

Client Requirements:

- **Processor:** Dual-core CPU at 2.0 GHz or higher for smooth video streaming and interaction.
- **RAM:** Minimum 4 GB to run browsers and video conferencing tools effectively.
- **Browser:** Up-to-date version of major browsers such as Google Chrome, Mozilla Firefox, or Microsoft Edge.
- **Internet Connection:** Stable and fast internet connection with adequate bandwidth for uninterrupted video and voice communication.

3.4 TOOLS AND PLATFORMS

The tools and platforms utilized for *CampusMeet* encompass both development and deployment phases, ensuring efficient project execution and reliable service delivery.

Development Tools:

- **Visual Studio Code:** The primary integrated development environment (IDE) for source code editing and management.
- **Postman:** API testing tool used to verify backend endpoints and ensure data accuracy.
- **Git and GitHub:** Version control system and collaboration platform for source code management.

Deployment Platform:

- **Render:** Platform-as-a-Service (PaaS) used for deploying the Django backend and hosting database services. Render offers seamless GitHub integration, automatic builds,

scaling capabilities, and SSL certificate provisioning, enabling a secure and scalable deployment environment.

Real-Time Communication:

- **ZegoCloud SDK:** Provides real-time video and audio communication capabilities integrated into *CampusMeet*. The platform initializes ZegoCloud services to enable live streaming, screen sharing, and interactive video conferencing essential for virtual classrooms.

Cloud Storage Solutions:

- **Render Database or External Cloud Storage:** For storing media files such as notes, images, and recordings securely and reliably.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

The system design phase for CampusMeet focuses on translating the requirements into a structured, scalable technical solution that supports seamless virtual classroom experiences. This phase encompasses the definition of the architectural design, core components, user interfaces, and data flow mechanisms required to fulfill both functional and non-functional specifications. Moving from the "what" in the analysis phase to the "how" of implementation, the design addresses backend and frontend interactions, real-time communication integration, attendance tracking, and resource management.

In CampusMeet, a well-planned system design enables critical features such as secure user authentication, interactive video conferencing using ZegoCloud, and intuitive dashboards for faculty and students. It ensures that user interactions, data transactions, and security protocols are carefully orchestrated for performance and reliability. The design process ultimately determines the system's success by providing a cohesive user experience, efficient data management, and scalability to accommodate growing institutional needs. This foundation sets the stage for the detailed implementation of a modern, high-performance virtual classroom platform tailored to educational contexts.

4.2 DATAFLOW DIAGRAMS

The data flow diagram is also known as "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design, so it is the starting point of specification down to the lowest level of detail. A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformation, and the lines represent the data flow in the system.

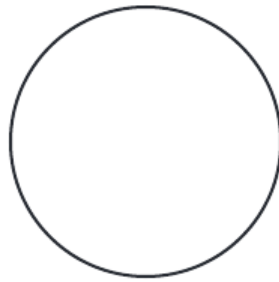
DFD Symbols:

- A system defined a source or destination of data.
- An arrow identifies data flow, data in motion.
- A circle represents the process that transforms incoming data flow to outgoing data flow.
- An open rectangular is data store-data at rest or a temporary repository of data.

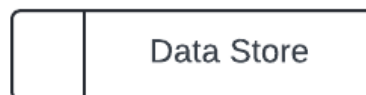
Arrow identifies Data Flow – Data in motion. It is a pipeline through which information flows. Data flow is a route, which enables packets travels from one point to another. Data may flow from a source to a processor and from data store or process. An arrow line depicts the flow, with arrowhead pointing in the direction of the flow.



A process represents a transformation where incoming data flows are changed into outgoing data flows.



A data store is a repository of data that is to be stored for use by one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names. If a process merely uses the content of store and does not alter it, the overhead goes from the store to the process.



A square defines a Source or Destination of system data. A source or sink is a person or part of an organization which enters or receives information from the system but is considered to be the context of data flow model.

