**BAMIDELE OLUMILUA UNIVERSITY OF EDUCATION SCIENCE AND**

**TECHNOLOGY**

**NAME: SEGUN - GOLD PETER ABIOLA**

**MATRIC NUMBER: 2862**

**DEPARTMENT: COMPUTING AND INFORMATION SCIENCE**

**FINAL PROJECT TOPIC:**

**DEVELOPMENT OF A VIRTUAL CLASSROOM PLATFORM USING DJANGO FRAMEWORK**

**CHAPTER ONE**

**1.0 INTRODUCTION**

**1.1 Background of Study**

Learning is the process of acquiring new knowledge or skills through study, practice, or instruction. Education serves as the backbone of any nation, and a minimum level of it is essential for meaningful participation in society. Traditionally, formal education has been delivered in a face-to-face classroom setting, from primary education through to higher institutions. However, before the COVID-19 pandemic, only a small portion of higher education systems incorporated online classes (Rani et al., 2020)

The emergence of the COVID-19 pandemic, a highly contagious viral outbreak forced a drastic transformation in the global education sector. As the virus spread rapidly across the world, causing widespread illness and death, the World Health Organization (WHO) declared it a global public health emergency. Consequently, various sectors including education faced severe disruptions. Government-enforced measures such as lockdowns, isolation, and social distancing made the traditional classroom setup impractical. This led many educational institutions worldwide to adopt digital alternatives such as Learning Management Systems (LMS) (Paramita et al., 2022)

In the modern educational context, LMS platforms have become vital tools for continuity in learning. These systems enable students to access educational content at any time and from any location, whether at home or in the library. The online classroom model is now implemented across all levels of education from elementary schools to universities. To enhance user experience, institutions continue to develop more user-friendly e-learning platforms. (Elmasry et al., 2021)

As a result of this technological shift, educational service providers are increasingly investing in digital tools and platforms. Devices such as computers, smartphones, and internet-based applications are being optimized for accessibility, and software solutions are continually evolving to support interactive and inclusive learning experiences. (Yu Ding et al., 2020)

A virtual classroom is a web-based online learning environment that allows teachers and students to interact in real-time or asynchronously. These platforms are often accessed through browsers or applications that may require installation. They enable remote learning while simulating the dynamics of a traditional classroom. (Ferriman, 2022)

E-learning has gained widespread acceptance as a flexible and effective way to share and acquire knowledge. An E-learning system is a comprehensive platform that develops, delivers, manages, and evaluates digital learning resources. It allows learners to enroll in courses, submit assignments, take quizzes, and engage in discussion with peers and instructors. With tools such as smartphones and tablets, students can learn anytime and anywhere. (Mahesh et al., 2023)

Modern LMS platforms facilitate the learning process by managing student data, tracking progress, and enabling communication. Tools like interactive whiteboards, digital highlighters, and document scanners enhance the learning experience by making educational materials easily accessible. Through internet connectivity, students can collaborate with instructors and classmates globally, making education more inclusive and flexible. (Gore et al., 2021)

Numerous studies have proposed frameworks that turn any environment into a smart learning space using online tools. The development of this project presents an innovative solution named “Kn9ght”, a Django-based (Python) virtual classroom platform that aims to improve digital learning environments through efficient content delivery, communication, and user interaction.

Advancements in technology have revolutionized distance education, making it a viable option, particularly for adult learners. Virtual classrooms represent one of the most impactful outcomes of this digital evolution. As defined by eLearnPortal, a virtual classroom is an online space that houses all course materials and facilitates real-time interaction between instructors and learners (eLearnPortal, 2020).

The increasing accessibility and usage of the Internet have brought profound changes to educational delivery. E-learning has gradually replaced traditional classroom models. The widespread availability of personal computers and internet connectivity has made distance learning more feasible and effective. Communication tools such as email, discussion boards, and forums allow for meaningful interaction between students and teachers, even in the absence of physical presence. (Fernández et al., 2022)

E-learning can thus be defined as a technology-based approach that enhances the learning process through computers and the internet. It ranges from instructors posting lecture materials online for asynchronous access, to fully interactive platforms where students participate in live classes. Learners are expected to take responsibility for their progress by keeping up with assigned readings and completing coursework independently. (Shafiq et al., 2021)

As E-learning continues to grow, terms like “virtual classroom” have become central to educational discourse. In a virtual classroom, students and teachers engage as if they were in a physical classroom, but through an online medium. The objective is to replicate the real-world classroom experience using technology, where all participants can see and interact with one another virtually in real-time.

Top of Form

Bottom of Form

**1.2 Statement of the Problem**

The current mode of education in many institutions remains heavily dependent on traditional face-to-face classroom interaction, which presents several limitations in today’s digital world. Students often rely solely on the information provided during physical lectures, restricting their access to broader educational resources and diminishing their ability to learn independently. Important academic materials such as notes and assignments are typically shared in hard copy, which students must manually organize and store, increasing the risk of loss and inefficiency.

Furthermore, physical presence in the classroom is required to benefit from lessons and interact with lecturers, which may force students to sacrifice other responsibilities such as work, family commitments, or health-related issues. Access to lecturers is also limited to their availability during office hours, which restricts timely support and clarification of course content. New lecturers taking over courses often lack access to previously used teaching materials, creating inconsistency in course delivery.

Another major challenge is the lack of a centralized digital system for collaborative learning. Students have no efficient platform to share resources, participate in supervised group discussions, or submit assignments digitally. These limitations collectively hinder the effectiveness of learning, reduce student engagement, and highlight the urgent need for a virtual classroom platform that supports resource sharing, interaction, and flexibility in teaching and learning.

**1.3 Aim and Objectives**

The specific aim of this project is to develop a functional, user-friendly, and scalable virtual classroom platform using the Django web framework. The platform is intended to support remote learning by enabling effective communication, resource sharing, and interaction between lecturers and students in a virtual learning environment.

**Objectives**

The objectives of the study are to:

1. To design a responsive and user-friendly interface using Django, HTML, and CSS, including dark/light mode and proper data management.
2. Develop a virtual classroom platform with user login, content sharing, assignment submission, and class interaction features.
   1. **Significance of the Study**
3. This project is important because it helps solve the problems students and teachers face when learning and teaching online. The main goal of this project is to create a platform that allows learning to happen without needing to be in a physical classroom. With this virtual classroom, students and teachers can connect from anywhere in the world, as long as they have internet access.
4. The platform will make learning easier by providing a place where teachers can upload notes, videos, assignments, and class schedules. Students can download learning materials, join video classes, submit assignments, and even ask questions through chat. This makes it more organized and less stressful for both students and teachers.
5. The virtual classroom also helps students who live far from school or cannot attend physical classes due to sickness, travel, or other reasons. It allows them to continue learning from home without missing important lessons. Teachers can also manage their classes better, save time, and teach more students at once.
6. This study is also helpful to schools and educational institutions because it shows a way to improve education through technology. It can reduce the cost of running physical classrooms, and it helps promote learning for all, including students in rural areas.
7. Lastly, this project can serve as a guide for other developers or schools who want to build similar platforms. It encourages the use of digital tools in education and supports the idea that learning should be available to everyone, anytime, and anywhere.

This project uses **Django** **framework** for the backend (server-side logic), and **HTML and CSS** for the frontend (user interface). The platform will be designed to work well on laptops and desktop computers. It will focus on easy navigation, user-friendly design, and basic security to protect user information.

### ****1.5 Limitations of the Study****

Although this project aims to create a useful and user-friendly virtual classroom platform, there are some limitations that must be noted:

1. **Limited to Web-Based Platform**: The virtual classroom is designed only as a website and not as a mobile app. This means it may not work perfectly on smartphones or tablets.
2. **No Advanced Security Features:** The platform includes basic login and logout systems but does not have advanced security features such as encryption, two-factor authentication, or full data protection.
3. **No Offline Access:** The platform requires an internet connection at all times. Students cannot access materials or lessons offline.
4. **Not Fully Accessible to All Devices**: The design may not work smoothly on all browsers or devices.

### ****1.6 Operational Definition of Terms****

To make this project easy to understand, here are simple meanings of some key terms used in the study:

1. **Virtual Classroom:** An online space where students and teachers can meet, learn, and share information without being in the same physical room.
2. **Platform**: A website or system that brings together different tools and features in one place for users to interact with.
3. **Login System:** A part of the website that allows users (like students and teachers) to enter their details (like username and password) to access their personal dashboard.
4. **Dashboard:** A personal homepage where users can see and manage their classes, assignments, notes, and other features.
5. **Assignments/Quizzes:** Tasks or questions given to students through the platform to test their understanding of what they’ve learned.
6. **File Upload/Download:** The ability for teachers to upload notes or videos, and for students to download and use them for learning.
7. **Chat Feature**: A messaging tool on the platform that allows students and teachers to talk or ask questions easily.
8. **Video Class:** A feature where students can watch recorded lessons or join live classes through video.
9. **Dark Mode / Light Mode:** A button that lets users change the screen colors to either dark or light, when being toggled depending on what feels more comfortable to their eyes.
10. **Django:** A web framework used by developers to build the platform, especially the backend or server part.
11. **HTML & CSS:** The languages used to design and style the platform's look and layout (how it appears to users).

**CHAPTER TWO**

**LITERATURE REVIEW**

**Overview of Literature Review**

The COVID-19 pandemic revolutionized the education landscape by necessitating the shift to virtual learning environments. Institutions worldwide had to rapidly transition from traditional face-to-face learning models to online platforms to ensure continuity in education. This shift spotlighted the growing need for robust, scalable, and user-friendly virtual classroom systems. Among the available technologies, Django, a Python-based web framework, emerged as a powerful tool for developing such platforms. This literature review explores scholarly articles, research papers, and case studies published between 2020 and 2025, focusing on the architecture, functionalities, challenges, security, and usability of virtual classroom platforms built using Django.

**2.2 Concept of Virtual Classroom**

Just as the term virtual means a simulation of the real thing, virtual classroom is a simulated classroom via internet, which provides a convenient communication environment for distance learners just like traditional face to face classroom. A virtual classroom allows learners to attend a class for anywhere in the world and aims to provide a learning experience that is similar to a real classroom. When we go to college we have a schedule of lectures, which we must attend. Student must arrive on time, and when he enters the classroom, he finds a teacher, fellow learners, a blackboard or whiteboard, LCD projector, optionally a television screen with videos. Likewise, a virtual classroom is scheduled, online teacher-led training session where teachers and learners interact together using computers linked to a network such as the internet.

A virtual classroom enables to bring learners from around the world together online in highly interactive virtual classes while greatly reducing the travel, time, and expenses of site teaching/training programs. It can be used as a solution for live delivery and interaction that addresses the entire process of creating and managing our teaching-learning process. It facilitates instructor and student in teaching-learning events, such As a seminar, online discussion or a live training for employees in company. As in tradition classroom, there are professor and fellow learners present with the student; we have many participants present in virtual classroom.

Himanshu Gore et al (2021)created a web application framework which is open source and written in the Python language. It uses MVT design structure (MVT stands for Model, View, and Template), due to its rapid development feature. Django is very demanding in the current Market. It takes less time to build any kind of application. Why we say this Model, View, Template because this framework will work based upon the model as a database and view as a controlling functionality and template will work as a user side for communication interaction.

The Django model will work as database management, we use two main commands like; python manage.py make migrations. Django will deduct the changes in models.py file and ready to send data into the sqlite3 (choose any database). Then we make python manage.py migrate, and then the Django system will save all changes in his database system. Then we make one more command Python manage.py run server at the end this will start our project and gives us the local host address for the project running locally. And views.py file will handle the request for the project to the API's call to template management in requests. We can write the views in the form of python functions.

Afroj Satwilkar et al, (2021) conducted a study that focuses on development of web application using Django. Django is a modern Python web framework that redefined web development in the Python world. A full stack approach, pragmatic design, and superb documentation are some of the reasons for its success.

Django, an open-source python web framework that saves time and make web development fun. Django follows the Model-View Controller (MVC) architectural pattern. Its goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and “plug ability” of components, rapid development, and the principle of “DRY” (Don't Repeat Yourself). Python is used throughout, even for settings, files, and data models. Technological Implementations in the field of job search has helped skilled people as well as people who want skilled workers in very important ways. The availability of all job searching sites helps the skilled people a lot in their day-to-day life. The paper illustrates a website model with the help of which skilled people can be able to update their skills and people who want skilled workers can find them from the same digital platform as well. The project is developed on Django Framework; the backend development is in Python, and SQLite. The frontend consists of HTML, CSS and JavaScript. The project developed is highly efficient, user friendly and simple.

Ikuomola (2018) created and deployed a reliable cloud mobile learning management scheme. The client model, mobile network, and cloud model are the three core parts of the system. Users like teachers and students can reach cloud services through a mobile network using the client model, which permits them to use a mobile application on their smart phones. The authentication server assists in the verification of each user’s identification when they attempt to connect the system. The system was tested on an Android phone, and the results reveal that it provides students with instructional content and materials where and when they need them. It also allows the user to interact with the information more easily because of the many modes of audio, graphics, and video, as well as feedback functions.

Khaleel et al, (2018) shared his idea for a prospective learning management model that combines IoT apps and tools to improve learning and teaching. It also changes education into a highly dynamic and adaptable process, allowing students to direct and govern their personal learning encounters. There are eight applications in the LMS that will gain from IoT integration possibilities. Although some of these applications are currently included in a number of current LMSs, they do not make use of IoT functionality. Virtual reality and classroom surveillance, for example, are innovative applications that will offer significant value to forthcoming LMSs. A general overview and illustration of the application contents are given for each application. For every suggested application, a preliminary action plan is presented, which is dependent on the existing IoT technologies. For college students, professors, and administrators, the planned IoT-enhanced LMS will deliver numerous extra services and opportunities.

Yu Ding et al, (2020) wrote an article that proposes and constructs a virtual environment for college physical education that incorporates the IoT, a cloud infrastructure, and a mobile user. This technology gathers essential data from the Internet of Things and reacts with the virtual reality environment, rendering the scene in the cloud and allowing users to explore virtual reality via a mobile node. The system’s core functionalities and data warehouse, as well as software design and system validation, were completed following the standard analysis and system architecture design. It is demonstrated that the built virtual reality technology for college physical education has a great applicability and promotional impact through the study of particular trial cases and user experiences information from a college. It also serves as a scholarly resource for furthering college physical education reforms, as suggested.

Paramita et al, (2022) conducted a research that examined the devastating consequences of a deadly virus pandemic, the ensuing lockdown, and the necessity to transform the Offline School to an online class. Due to the closure of schools and universities as a result of the epidemic, the education industry relies primarily on “online learning.” The rapidly growing cloud-centric aims among enterprises and institutions have emphasized the crucial need to progressively upgrade security within cloud architecture. The goal of this study is to investigate and characterize the various online educational portals, learning resources, strategies, and technologies available. It also outlines the infrastructure and techniques that can be used to conduct tests in a safe and non-cheating setting. It also considers if the concept of cloud based learning administration is generally appropriate in the particular situation.

Elmasry et al. (2021) conducted a study aimed at recommending suitable cloud infrastructure for higher education institutions in developing countries, including the optimal cloud deployment and operational models. The research explored the current adoption of cloud computing in higher education, its architectural design, and its potential applications in distance learning. Furthermore, the study proposed a framework for integrating cloud computing with learning management systems (LMS) to establish a cloud-based academic environment.

Shafiq et al, (2021) conducted a study that focuses on learning methods and other technologies that have been used to help the educational process. In this paper, a methodological approach of a cloud-based learning management scheme for educational institutions is described. In addition, the study will show how the suggested model produces the predicted results. The goal is to create a friendly environment for instructors, teaching staff, and managers. It also takes care of more important e-learning duties including seeking learning management system (LMS) navigations, textual and multimedia learning materials, and video learning materials.

Digvijay Pandey et al. (2021) studied how to socio-demographic and related factors on perceptions regarding the delivery of online classes during India’s COVID-19 lockdown. The study employed a community-based survey to collect primary data, focusing exclusively on responses obtained directly from participants at a specific point in time.

**CHAPTER THREE**

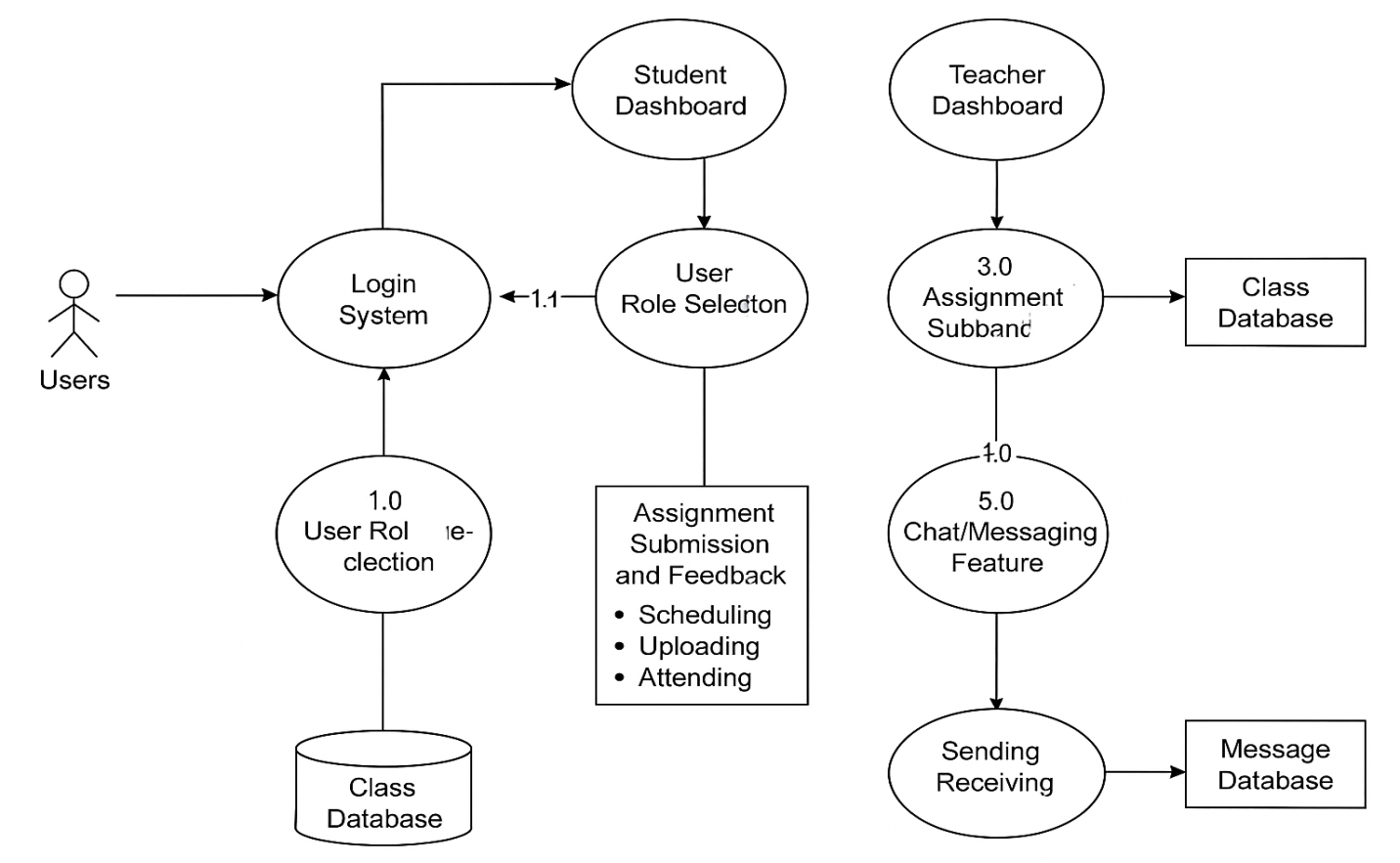
**METHODOLOGY**

**Introduction**

The introduction of several educational websites that offer academic materials makes life easier and more pleasant in this age of digital learning. Kn9ght is a new digital or e-learning application that is being proposed and developed for resource sharing purposes. The Django Python Web Platform is used to create this virtual environment or service, which is utilized to deliver solutions via the Internet. It promotes speedy development and clear, realistic design created by skilled programmers. It eliminates a lot of the challenges associated with Web development, allowing focusing on building the application rather than reinventing the wheel.

**3.2 Architecture of the Proposed System**

**Data Flow Diagram: Virtual Classroom System**



**Main Features Represented:**

1. Login System
2. User Role Selection
3. Student & Teacher Dashboards
4. Class Management (scheduling, uploading, attending)
5. Assignment Submission and Feedback
6. Chat/Messaging Feature

**3.2 SYSTEM REQUIREMENTS**

**3.2.1 Hardware Requirements**

The selection of hardware is very important in the existence and proper working of any software. When selecting the hardware, the size and requirements are also important to run the software. The minimum hardware requirements are as follows:

1. Processor: Intel CORE i3
2. RAM: 4 GB

3. Disk space: minimum 256 GB

**3.2.2 Software Requirements**

1. Operating System, (Windows, MacOS).

2. Python, HTML, CSS, JavaScript.

3. Django Framework and SQLite database, (which comes by default with Django).

4. An Internet Browser, (Google Chrome, Microsoft Edge, Firefox, etc.).

5. Code Editor, (Visual Studio code, etc.).

6. The package manager PIP (pip is a python package-management system written in Python used to install and manage software packages).

The construction technique applied the Iterative Pattern of Software Development, which makes it easy to find and correct errors simultaneously. The concept is to add functionality prior to actually designing, testing, and implementing it. To develop the proposed system, several facts and features are evaluated, as listed below, in order to get the desired result.

1. To provide course materials to support various educational categories and its subcategories.

2. Each User has the ability to add Class Notes and Assignments to the app.

3. Present instructional resources via an open live board.

4. To make video conferencing between students and lecturers more convenient.

5. To keep track of and evaluate the user’s progress.

A feasibility analysis is performed to assess the practicality of a concept, such as verifying that a proposal is legally, technologically, and commercially feasible. The feasibility analysis for this work is carried out as follows:

**Project Requirements:** A Project Control List is a list that contains a succession of activities that must be included in a specific system. When we come across a new demand during the development process, it can be added to the list.

1. User Registration and User Login

2. Separate Account Page for each User.

3. A whiteboard for inter-user conversation.

4. A Search Bar

**Project Objectives:** The following objectives for the efficiency of the project to be developed should also be considered.

1. Planned Approach: The application’s operation is meticulously planned and managed. The data will be correctly saved in data stores, which will aid in both retrieval and storage of information.
2. Information retrieval in real time: The suggested system’s major goal is to enable speedy and efficient information retrieval.
3. Easy to Use: The system should be user-friendly and designed in such a way that it may be produced in a short amount of time while staying within the organization’s financial constraints.

**3.3 SOFTWARE USED**

**3.1.1 Front End Technologies**

HTML is the standard markup language used to create web pages. It defines the structure of a webpage using a series of elements, also called tags.

**3.1.1 Key Concept of HTML:**

* **Elements & Tags:** HTML documents consist of elements enclosed in angle brackets (<tag>). Example: <p>this is a paragraph. </p>.
* **Document Structure:** HTML documents have a structure including <html>, <head>, and <body> sections.
* **Attributes:** Provide additional information about elements (e.g., <img src="image.jpg" alt="Description">).
* **Forms and Inputs:** Used for collecting user input through text fields, buttons, checkboxes, and dropdowns.
* **Hyperlinks & Navigation:** Links (<a href="URL">Click here</a>) allow users to navigate between pages.
* **Tables & Lists:** HTML supports tables (<table>), ordered (<ol>), and unordered (<ul>) lists.

**3.1.2 Understanding CSS (Cascading Style Sheets)**

CSS is a styling language used to control the presentation and layout of web pages. Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the Visual style of web pages and user interfaces written in HTML. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications. CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

**3.1.3 Key Concept of CSS:**

* **Selectors & Properties:** Select elements and apply styles (e.g., p {color: blue ;}).
* **Box Model:** Defines how elements are spaced with properties like margin, border, padding, and width/height.
* **Positioning & Layouts:** Uses static, relative, absolute, and fixed positioning for layout control.
* **Flex box & Grid:** Modern layout techniques that simplify responsive web design.
* **Media Queries:** Allow styles to change based on screen size for responsiveness.
* **Animations & Transitions:** Add dynamic effects using properties like transform and animation.

**3.1.4 Relationship between HTML and CSS**

* HTML provides the structure, while CSS enhances the appearance and layout.
* CSS can be applied **inline**, **internally**, or **externally** using style sheets.
* A well-structured HTML document with CSS improves usability, accessibility, and design consistency.

**3.1.5 COMMOM USES OF HTML AND CSS**

HTML (HyperText Markup Language) and CSS (Cascading Style Sheets) are used to create and design web pages. Here are their common uses:

**HTML Uses**

1. **Building Web Pages** – HTML is the foundation of all websites, helping structure content like text, images, and links.
2. **Creating Forms** – Used to make contact forms, login pages, and sign-up forms.
3. **Embedding Media** – Helps add videos, audio, and images to web pages.
4. **Navigation Menus** – Helps create menus and links for easy website browsing.
5. **Tables & Lists** – Organizes information neatly with tables and bullet points.

**CSS Uses**

1. **Styling Web Pages** – Adds colors, fonts, and spacing to make websites look nice.
2. **Making Websites Responsive** – Helps adjust web pages for phones, tablets, and desktops.
3. **Animations & Effects** – Creates hover effects, transitions, and animations.
4. **Layout Design** – Helps arrange content using grids and flex box.
5. **Theming & Customization** – Allows easy changes to website appearance without editing HTML.

**3.1.6 Principal Duties:**

**HTML (HyperText Markup Language)**

1. Defines the structure of web pages.

2. Uses elements like headings, paragraphs, lists, and links.

3. Embeds multimedia (images, audio, and video).

4. Creates forms for user input.

**CSS (Cascading Style Sheets)**

1. Styles and enhances the visual presentation of web pages.
2. Controls layout, colors, fonts, and spacing.
3. Enables responsive design for different screen sizes.
4. Uses animations and transitions for interactive effects.
5. Provides consistency across multiple web pages with external style sheets.

The front end consists of the user device and the application that allows them to use the cloud computing system. With the use of HTML, CSS, and JavaScript, the program’s front end has been made more user-friendly. By installing the web application under HTTPS, the web application may safely transfer data with the webserver. JavaScript is a component of web applications that allows for interactive web sites. As the form provided by the user may contain incorrect information, it is critical to validate it. As a result, validation is required to verify the user’s identity.

**3.2 Back End Technologies**

**3.2.1 Python Language**

Python language is a high-level, dynamically typed one that is among the most popular general-purpose programming languages. Python is an Interpreted, object-oriented, and high-level programming language. It is called an interpreted language as its source code is compiled to byte-code which is then interpreted. Python’s features, among other things, are what make it popular. For instance, it supports dynamic typing and dynamic binding.

**3.2.2 Features of Python**

1. Python is open source. You can download it for free and use it in your application. You can also read and modify the source code.

2. The Python framework also has modules and packages, which facilitates code reusability.

3. It provides rich data types and easier to read syntax than any other programming Languages.

4. Compared to other programming languages, it allows more run-time flexibility.

5. A module in Python may have one or more classes and free functions.

6. Libraries in Pythons are cross-platform compatible with Linux, Macintosh, and Windows.

7. Python supports functional and structured programming as well as OOP.

8. It supports interactive mode that allows interacting Testing and debugging of snippets of code.

Django Framework Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.

Django is a high-end Python Web platform that promotes fast expansion and a simple, practical design. The framework prioritizes component extensibility and “adaptly,” as well as minimal code, less coupling, and quick development. SQLite is the best utilized database engine in the industry, and Django default database is SQLite3.

**3.2.3 Characteristics of Django Framework**

1. Django was designed to help developers take applications from concept to completion as quickly as possible.

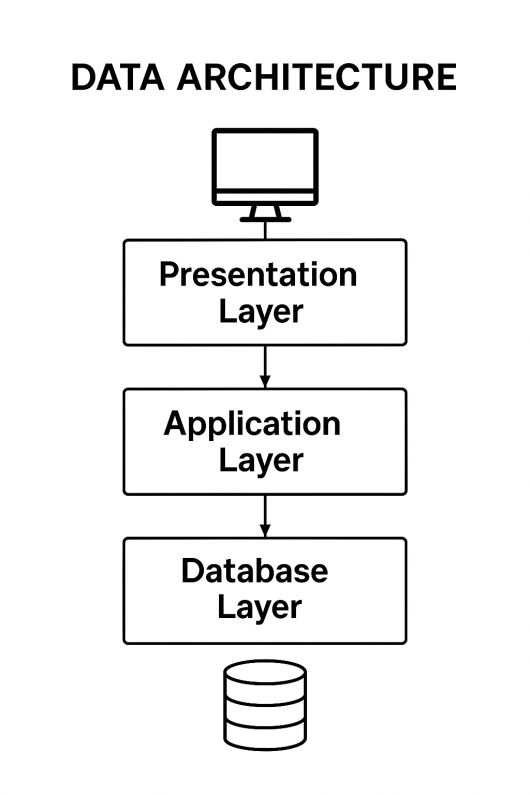
2. It includes dozens of extras you can use to handle common web development tasks. Django takes care of user authentication, content administration, site maps, RSS feeds, and many more tasks right out of the box.

3. It takes security seriously and helps developers avoid many common security mistakes, such as SQL injection, cross-site scripting, cross-site request forgery and click jacking. Its user authentication system provides a secure way to manage user accounts and passwords.

4. Some of the busiest sites on the planet use Django ability to quickly and flexibly scale to meet the heaviest traffic demands.

5. Companies, organizations and governments have used Django to build all sorts of things from content management systems to social networks to scientific computing platforms.

**Data Architecture**

****

**1. Presentation Layer (Frontend)**

* **Technologies**: HTML, CSS, JavaScript (optional)
* **Purpose**: Interface through which users interact with the system
* **Users**: Teachers, Students, Admins
* **Data Flow**:
  + Login forms
  + Dashboard views
  + Assignment upload/submission forms
  + Messaging/chat UI
  + Class attendance interface

**2. Application Layer (Backend with Django Framework)**

* **Technologies**: Django (Python), Django Views & Templates
* **Purpose**: Logic layer to process user requests and manage data interaction
* **Modules**:
  + Authentication (Login, Register)
  + Role Management (Student/Teacher)
  + Dashboard Routing
  + Class Scheduling & Material Upload
  + Assignment Handling
  + Chat/Messaging Management
* **Key Django Apps**:
  + accounts: Login and role management
  + classroom: Classes, schedules, uploads
  + assignments: Submission and feedback
  + chat: Messaging and notifications

**3. Database Layer (Data Storage)**

1. **Technology**: SQLite (for dev), PostgreSQL/MySQL (for production)
2. **Purpose**: Stores persistent data used across the platform
3. **Main Tables/Models**:

* User (username, email, password, role)
* Student Profile / Teacher Profile
* Class (title, description, schedule, teacher\_id)
* Material (file, associated class)
* Assignment (file, student, teacher, feedback)
* Message (sender, receiver, content, timestamp)

**CHAPTER FOUR**

**SYSTEM REQUIREMENTS**

**3.2.1 Hardware Requirements**

The selection of hardware is very important in the existence and proper working of any software. When selecting the hardware, the size and requirements are also important to run the software. The minimum hardware requirements are as follows:

1. Processor: Intel CORE i3
2. RAM: 4 GB

3. Disk space: minimum 256 GB

**3.2.2 Software Requirements**

1. Operating System, (Windows, MacOS).

2. Python, HTML, CSS, JavaScript.

3. Django Framework and SQLite database, (which comes by default with Django).

4. An Internet Browser, (Google Chrome, Microsoft Edge, Firefox, etc.).

5. Code Editor, (Visual Studio code, etc.).

6. The package manager PIP (pip is a python package-management system written in Python used to install and manage software packages).

The construction technique applied the Iterative Pattern of Software Development, which makes it easy to find and correct errors simultaneously. The concept is to add functionality prior to actually designing, testing, and implementing it. To develop the proposed system, several facts and features are evaluated, as listed below, in order to get the desired result.

1. To provide course materials to support various educational categories and its subcategories.

2. Each User has the ability to add Class Notes and Assignments to the app.

3. Present instructional resources via an open live board.

4. To make video conferencing between students and lecturers more convenient.

5. To keep track of and evaluate the user’s progress.

A feasibility analysis is performed to assess the practicality of a concept, such as verifying that a proposal is legally, technologically, and commercially feasible. The feasibility analysis for this work is carried out as follows:

**Project Requirements:** A Project Control List is a list that contains a succession of activities that must be included in a specific system. When we come across a new demand during the development process, it can be added to the list.

1. User Registration and User Login

2. Separate Account Page for each User.

3. A whiteboard for inter-user conversation.

4. A Search Bar

**Project Objectives:** The following objectives for the efficiency of the project to be developed should also be considered.

1. Planned Approach: The application’s operation is meticulously planned and managed. The data will be correctly saved in data stores, which will aid in both retrieval and storage of information.
2. Information retrieval in real time: The suggested system’s major goal is to enable speedy and efficient information retrieval.
3. Easy to Use: The system should be user-friendly and designed in such a way that it may be produced in a short amount of time while staying within the organization’s financial constraints.

**REFERENCES**

 Afroj Satwilkar, Amruta Sawant, Ashwini Pawar, & Snehal Shelke. (2021).  
  Development of Web Based Learning Management System for Online Education.  
  International Journal of Research Publication and Reviews, 2(7), 234–237.

 Django Documentation. (2024).  
  Security Overview. Retrieved from <https://docs.djangoproject.com/>

 Digvijay Pandey, Chetna Chauhan, Rishika Kumar, & Kritika. (2021).  
  Impact of Socio-Demographic Factors on Views of Online Learning During India’s

COVID-19 Lockdown.  
  Education and Information Technologies, 26(4), 4567–4583.

 Elmasry, W., Hassan, M., & Salem, A. (2021).  
  Cloud Infrastructure Recommendation for Educational Institutions During the Pandemic.  
  International Journal of Cloud Applications and Computing (IJCAC), 11(3), 45–57.

 Fernández, P., et al. (2022).  
  Immersive Learning Frameworks: A Systematic Literature Review. arXiv.

 Kuriščák, P., et al. (2022).  
  Design and Implementation of a Framework for Remote Experiments in Education. arXiv.

 Mahesh, K, Govinda Sai K\*, & Suraj Hussain D. (2023).  
  Muster: Virtual Classroom using Django Framework. JScholar Online.

 Paramita, D., Anggraeni, N., & Yuliani, R. (2022).  
  The Devastating Consequences of Deadly Pandemics on Education Systems: A Study of

COVID-19.  
  Jurnal Pendidikan Indonesia, 3(2), 89–97.

 Rakesh Kumar Singh, Ashutosh Singh, Arnav Pratap Singh, Mohammad Shabaz, Bhupesh

Kumar Singh, & Vishal Jagota. (2021).  
  Untitled Paper.  
  Annals of R.S.C.B., 25(6), 4576–4585.

 Schreens, J. (2022).  
  Effective Learning Factors in Online Classrooms.  
  Journal of Educational Psychology and Technology, 8(2), 150–164.

 Shafiq, M., Wang, M., & Shaheen, S. (2021).  
  A Study Focusing on Learning Effectiveness of Virtual Classrooms in Pakistan.  
  International Journal of Distance Education and E-Learning, 6(1), 50–60.

TRUSTID Project. (2024).  
  Ensuring Academic Integrity with Biometric Verification. MDPI.

 Tushar Sawant, Vaibhav Shirke, & Santosh V. Jadhav. (2021).  
  Untitled Research Note.  
  Volume 4, Issue 11, pp. 119–120.

 UNESCO. (2022).  
  Education in a Post-COVID World: Nine Ideas for Public Action.  
  Paris: United Nations Educational, Scientific and Cultural Organization.

 Yan, T. (2023).  
  Construction of University Cloud Classroom Platform Based on Python.  
  Atlantis Press.

 Yu Ding, H., Chen, Y., & Liu, W. (2020).  
  Construction of Virtual Environment for Online Classroom Interaction.  
  IEEE International Conference on Information Technology.