



MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

COMPUTER SCIENCE

IT LAB PROJECT REPORT:

DISTRIBUTED SYSTEMS SUBJECT WEBSITE IMPLEMENTATION

(Project Title)

Name:

Anurag Nayak, 210905016, anurag.nayak@learner.manipal.edu
Adruti Onam, 210905190, adruti.onam@learner.manipal.edu
Kushala Sarada, 21905189, kushala.v@learner.manipal.edu

Manipal Institute of Technology, Manipal.

March 2024

Table of Contents

| | |
|--|-------------|
| 1. Introduction | [3] |
| 2. Methodology | [3] |
| 3. Implementation | [5] |
| 4. Results and Analysis/Snapshots | [9] |
| 5. Discussion | [17] |
| 6. Conclusion | [19] |
| 7. Future scope | [20] |
| 8. References | [21] |

1. Introduction

Overview

The advent of digital technologies has profoundly transformed the educational landscape, offering new avenues for teaching and learning. This project aims to harness these technological advancements through the development of a Distributed Systems Information platform, specifically designed to cater to the needs of the educational sector.

Utilizing Django, a high-level Python Web framework known for its simplicity and efficiency, the project seeks to create a robust, user-friendly platform that facilitates interaction between teachers and students in a virtual environment. The core objective is to streamline the educational process, making it more accessible, flexible, and engaging for all parties involved. By providing separate access points for teachers and students, the platform ensures a tailored experience that addresses the distinct needs of each user group, thereby enhancing the overall efficiency of the learning process.

Scope

The project encompasses a wide range of functionalities designed to optimize the educational experience. For teachers, the platform offers an array of administrative features, including the ability to mark student performance, upload educational materials (such as slides and notes), and make announcements.

These features are aimed at simplifying the administrative burden on educators, allowing them to focus more on teaching and less on logistical tasks. Additionally, the platform provides the capability for targeted communication, enabling teachers to broadcast announcements to the entire class or send messages to individual students, thereby facilitating personalized instruction.

For students, the platform offers a centralized location to access all course-related materials and communications. This includes viewing grades, downloading lecture slides and notes, and receiving announcements from their instructors. By consolidating these resources

in one easy-to-navigate platform, students can manage their academic responsibilities more efficiently, leading to a more organized and effective learning process.

Relevance

The implementation of this platform can lead to significant improvements in information dissemination, making education more inclusive and equitable.

2. Methodology

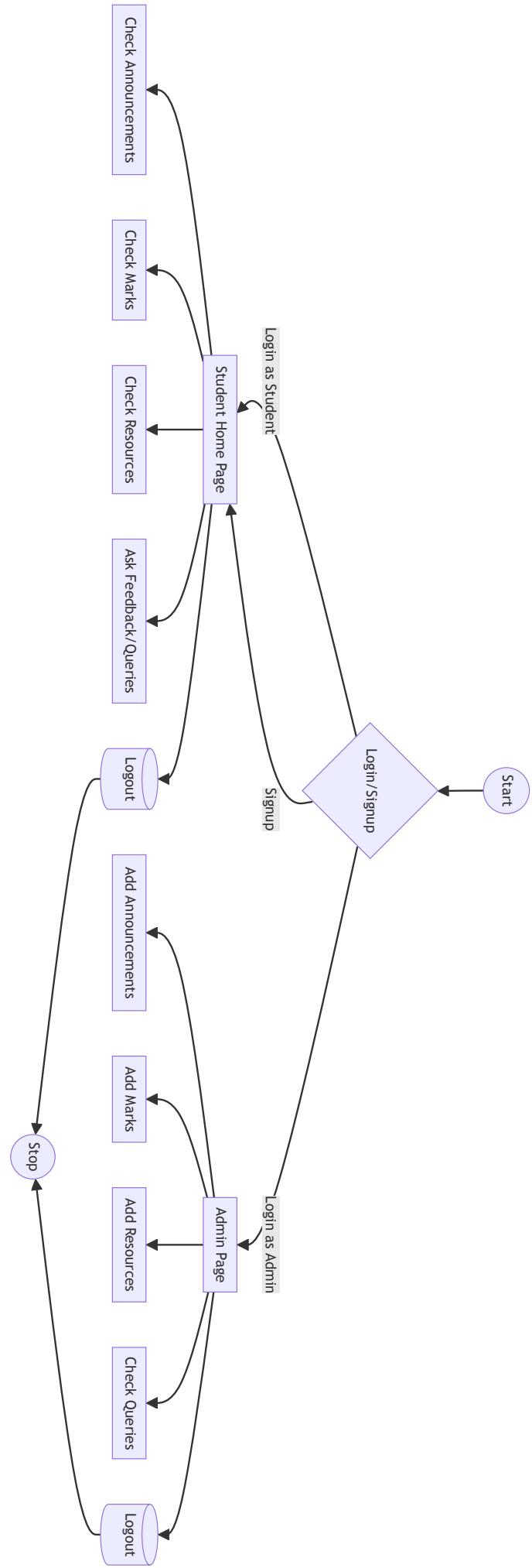
2.1 Authentication Implementation

Authentication Mechanism

The project leverages Django's built-in authentication system to manage user logins and session management. This robust framework provides out-of-the-box functionalities for user authentication, including password hashing, session management, and form handling, ensuring a secure and efficient login process. To accommodate the distinct roles of teachers and students within the platform, the system utilizes Django's Group and Permission models. Upon registration, users are assigned to either the Admin or "Student" group, each with its unique set of permissions that control access to various functionalities of the platform.

Managing Permissions and User Roles

User roles and permissions are central to the platform's security and functionality. Django's auth system allows for a granular definition of permissions at the model level, which are then assigned to groups. Teachers, being part of the Admin group, are granted administrative permissions that include creating and uploading content, entering grades, and making announcements. Conversely, the "Student" group is primarily assigned read-only permissions, enabling them to view content, download materials, and receive announcements. This delineation of roles and



permissions is critical for maintaining the integrity of the educational process, ensuring that users can only access features relevant to their roles.

2.2 System Testing

Testing Strategies

The reliability and performance of the system are ensured through a comprehensive testing strategy, encompassing both unit tests and integration tests. Unit tests are employed to verify the correctness of individual components or functions, such as the authentication process, user role assignments, and data handling operations. Integration tests, on the other hand, assess the system's performance as a whole, focusing on the interaction between different components, such as the integration of the database with the application logic and the user interface.

Testing Frameworks and Tools

For implementing these tests, the project utilizes Django's built-in testing framework, which is based on Python's unittest library. This framework provides a range of tools and functionalities to simulate requests, insert test data, and inspect the application's output, making it an invaluable resource for ensuring the platform's functionality and stability. Additionally, tools such as Coverage.py are used to measure the extent of code executed by the tests, identifying areas that may require additional testing.

2.3 Documentation and User Instructions

Documentation Approach

Comprehensive documentation is critical for ensuring that both teachers and students can effectively navigate and utilize the platform. The project adopts a multi-faceted approach to documentation, encompassing both technical documentation for future developers and user instructions for the platform's end-users.

Technical Documentation: For developers and system administrators, the project documentation includes detailed explanations of the system architecture, codebase, and deployment processes. This documentation is

created using Sphinx, a tool that generates documentation from reStructuredText sources into various formats, providing a cohesive and navigable set of documents.

User Instructions: For teachers and students, user manuals and instructional videos are created to guide them through the platform's features and functionalities. These resources are designed to be intuitive and accessible, incorporating screenshots, step-by-step instructions, and FAQs to address common issues and queries. The user documentation is made available directly within the platform for easy access and reference.

By employing these methodologies for authentication implementation, system testing, and documentation, the project ensures a secure, reliable, and user-friendly platform that meets the needs of both teachers and students in the digital learning environment.

3. Implementation

The development of the Distributed Systems Information platform for educational purposes was a multifaceted process that required careful planning, design, and implementation. This section outlines the development process, with a focus on the Django framework and the application of distributed systems principles, as well as how testing and documentation were integrated to ensure the platform's quality and usability.

Development Process and Django Framework

The Django framework was chosen for this project due to its "batteries-included" philosophy, providing a comprehensive suite of tools and functionalities that facilitate rapid development and clean, pragmatic design. The development process was iterative, adhering to agile methodologies to accommodate changes and feedback efficiently.

Model-View-Template (MVT) Architecture: Django's MVT architecture was pivotal in structuring the application. This architecture separates the application into three interconnected parts:

- **Models** define the data structure, facilitating interaction with the database to perform CRUD (Create, Read, Update, Delete) operations.
- **Views** handle the business logic, processing requests and returning responses.
- **Templates** manage the presentation layer, defining how the data is presented to the user.

This separation of concerns not only streamlined development but also enhanced maintainability and scalability, essential attributes for distributed systems.

3.1 Testing and Documentation

Testing Integration: Testing was integrated from the onset of the development process, adhering to a test-driven development (TDD) approach. This approach emphasizes the creation of tests for new features or bug fixes before the actual development, ensuring that all code added to the project is reliable and meets the specified requirements. Django's testing framework facilitated the creation of unit and integration tests, allowing for automated testing of the application's components and workflows. Continuous Integration (CI) tools were employed to automate the testing process, running tests automatically with each code commit to detect and resolve issues early in the development cycle.

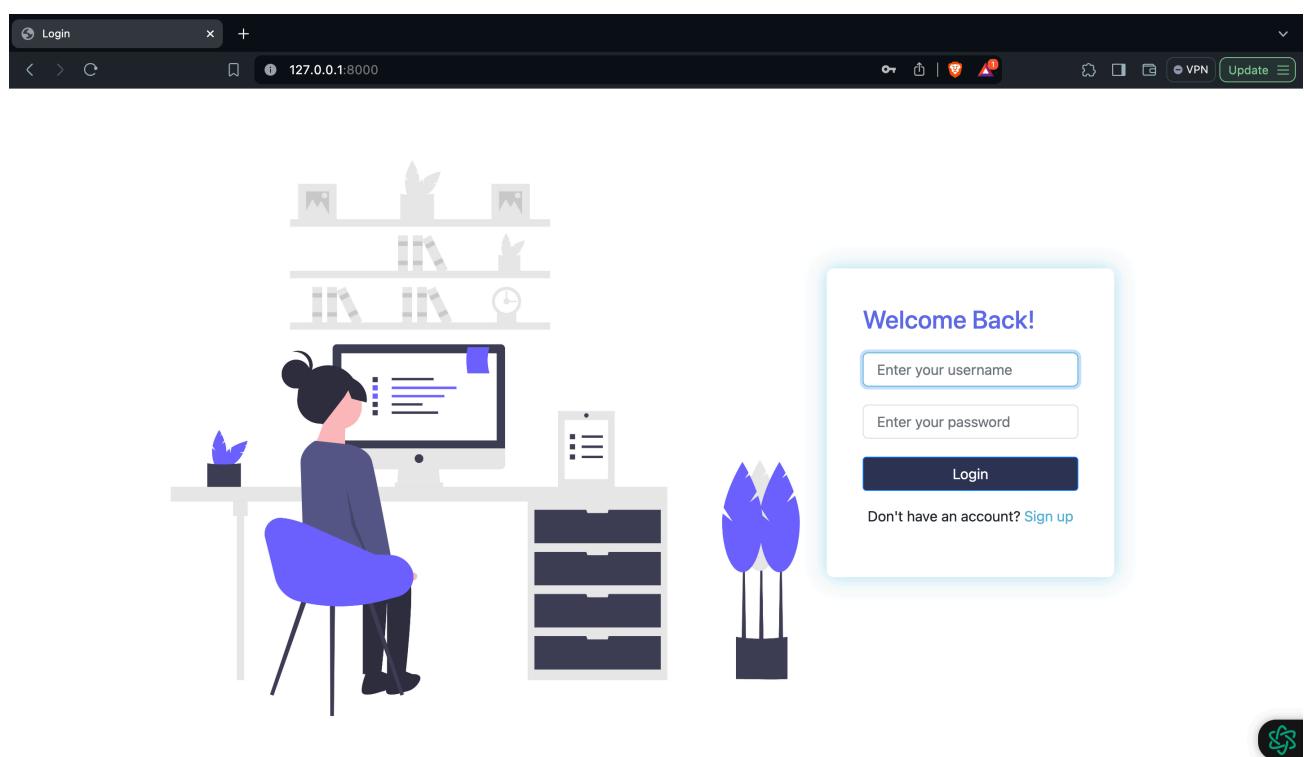
Documentation: Documentation was treated as an integral part of the development process. Technical documentation was generated using tools like Sphinx, which allowed for the creation of detailed and structured documentation for the system's architecture, API endpoints, and development guidelines. This documentation serves as a valuable resource for current and future developers, ensuring the system's longevity and ease of maintenance.

For user documentation, a user-centric approach was adopted. Guides, FAQs, and instructional videos were developed in tandem with the platform's features, ensuring that documentation kept pace with the development. This material was designed to be intuitive and accessible, providing users with the knowledge needed to navigate and utilize the platform effectively. By integrating testing and documentation into the

development process, the project team ensured that the final product was not only functionally robust but also user-friendly and well-documented, reflecting the high-quality standards set for the project.

4. Results and Analysis/Snapshots

4.1 Snapshots



4.1.1 Login Page

The screenshot shows a student dashboard titled "Distributed Systems Dashboard". The top navigation bar includes a "Logout" button and a "VPN" status indicator. The main content area features three cards:

- Notes/Resources**: A sidebar with a "Links" heading and a list of resources:
 - DS Module 3
 - Course Plan
 - DS Module 4
 - DS Module 1
 - TextBook 1
 - TextBook 4
 - DS Playlist(Additional Recourse)
- Marks**: A table showing assignment scores:

| Assignment Title | Score |
|------------------|-------|
| MISAC 1 | 4 |
| MISAC 2 | 2 |
| FISAC 1 | 8 |
- Announcements**: A table of messages with timestamps:

| Time | Message |
|------------------------|--|
| Thu, 14 Mar 2024 13:49 | Your attendance has been low lately. Let's chat in my cabin during my free hours to discuss how we can improve this. Let me know when you're available. |
| Thu, 14 Mar 2024 13:45 | Exciting update: Don't miss out on our upcoming webinar discussing real-world applications of distributed systems! Details will be available on the dashboard soon |
| Thu, 14 Mar 2024 13:44 | Reminder: Tomorrow's group project evaluation has been rescheduled. Check the dashboard for the new time slot |

A "Submit Feedback/Query" button is located at the bottom right of the dashboard.

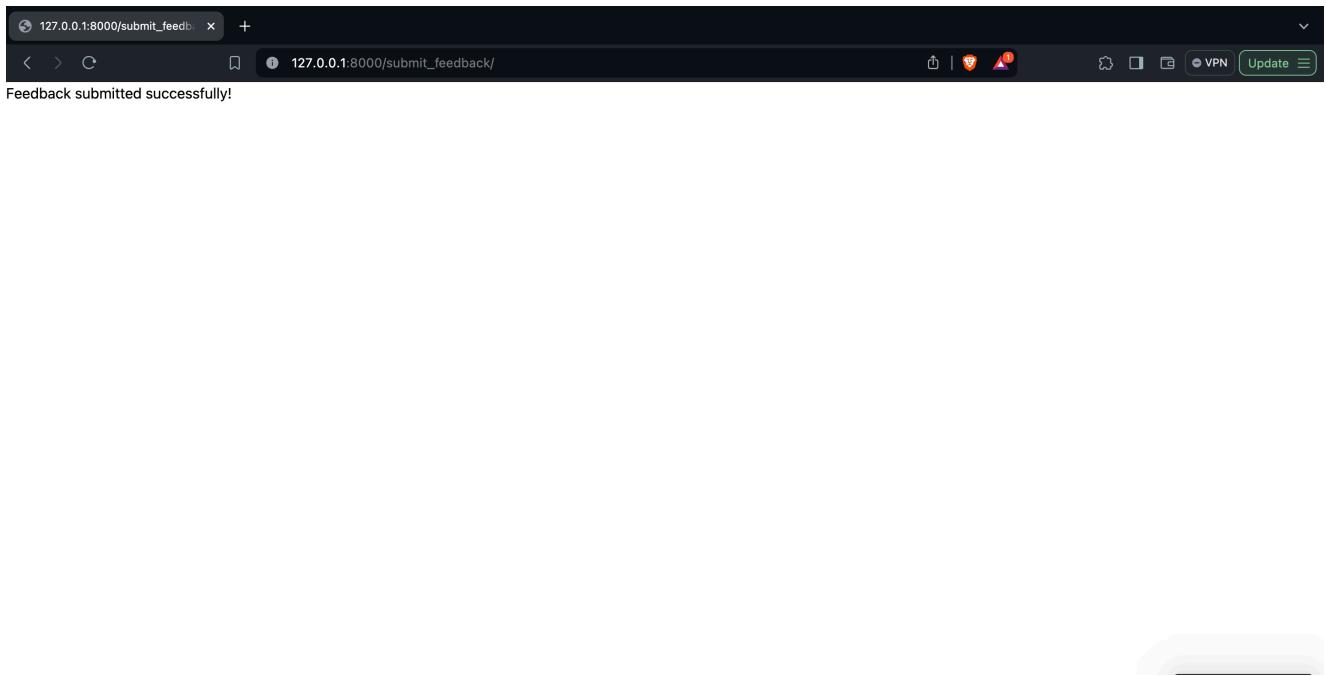
4.1.2 Student Dashboard

The screenshot shows a student dashboard titled "Student Dashboard". The top navigation bar includes a "Logout" button and a "VPN" status indicator. The main content area features a list of announcements and a feedback form:

- Announcements**: A table of messages with timestamps:

| | |
|------------------------|---|
| Thu, 14 Mar 2024 13:44 | Reminder: Tomorrow's group project evaluation has been rescheduled. Check the dashboard for the new time slot |
| Thu, 14 Mar 2024 13:42 | Reminder: Project proposal deadline approaching. Submit by 20th March 2024. |
| Thu, 14 Mar 2024 13:41 | New resource added: Distributed Systems Module 3 slides. |
| Thu, 14 Mar 2024 13:41 | Guest lecture on distributed system security tomorrow from 2:30 PM to 4:30 PM. Attendance mandatory. |
| Thu, 14 Mar 2024 13:40 | Class cancelled for 12th March 2024. Please check the Distributed System Dashboard for updates. Thank you. |
| Thu, 14 Mar 2024 13:38 | Welcome to our Distribute System Dashboard! Stay updated on course news, assignments, and resources here. |
- Feedback Form**: A modal window with a text input field containing "I have a doubt in Q3 of FISAC 1." and a "Submit" button.

4.1.3 Student Query and Feedback



4.1.4 Successful Implementation of query/feedback

A screenshot of the Django admin dashboard. The top navigation bar shows "Site administration | Django site" and the URL "127.0.0.1:8000/admin/". The dashboard has two main sections: "AUTHENTICATION AND AUTHORIZATION" (Groups, Users) and "MYAPP" (Announcements, Marks, Queries, Resources). On the right side, there is a sidebar titled "Recent actions" and "My actions", listing various administrative tasks such as "DS LAB Resource", "TextBook 1 Resource", and "FISAC 1 - Max01 (8) Mark". A small green logo is visible in the bottom right corner of the sidebar.

4.1.5 Admin Dashboard

The screenshot shows the Django administration interface for the 'Announcements' model. The left sidebar lists 'MYAPP' models: Groups, Users, Announcements, Marks, Queries, and Resources. The main content area is titled 'Select announcement to change' and displays a list of 8 announcements. The first announcement is selected, showing its details: 'To Max01: Your attendance has been low lately. Let's chat in...'. Other announcements include broadcast messages about group project evaluations, proposal deadlines, new resources, guest lectures, class cancellations, and a welcome message for the distributed system dashboard.

4.1.6 Adding announcements by Admin

The screenshot shows the Django administration interface for the 'Marks' model. The left sidebar lists 'MYAPP' models: Groups, Users, Announcements, Marks, Queries, and Resources. The main content area is titled 'Select mark to change' and displays a list of 5 marks. The first mark is selected, showing its details: 'FISAC 1 - Max01 (8)'. Other marks include MISAC 2 - Max01 (2), MISAC 1 - Max01 (4), INSEM -1 - anurrag (100), and INSEM -1 - nate (35).

4.1.7 Adding assignment and marks

The screenshot shows the Django administration interface for the 'Queries' model. The left sidebar has 'MYAPP' selected, showing 'Announcements', 'Marks', 'Querys', and 'Resources'. The main area title is 'Select query to change'. It lists five queries, each with a checkbox and a timestamp. Action buttons include '-----', 'Go', and 'ADD QUERY'. A status bar at the bottom shows the URL: 127.0.0.1:8000/admin/myapp/query/.

4.1.8 Checking Queries from Student

The screenshot shows the Django administration interface for the 'Resources' model. The left sidebar has 'MYAPP' selected, showing 'Announcements', 'Marks', 'Querys', and 'Resources'. The main area title is 'Select resource to change'. It lists seven resources, each with a checkbox and a name. Action buttons include '-----', 'Go', and 'ADD RESOURCE'. A status bar at the bottom shows the URL: 127.0.0.1:8000/admin/myapp/resource/.

4.1.9 Editing Resources

Django administration

Home > Myapp > Resources > Add resource

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups + Add
- Users + Add

MYAPP

- Announcements + Add
- Marks + Add
- Querys + Add
- Resources + Add

Add resource

Title:

Link:

Description:

Buttons: SAVE, Save and add another, Save and continue editing

4.1.10 Adding Resources

Select user to change | Django site admin

127.0.0.1:8000/admin/auth/user/

Django administration

Home > Authentication and Authorization > Users

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups + Add
- Users + Add

MYAPP

- Announcements + Add
- Marks + Add
- Querys + Add
- Resources + Add

Select user to change

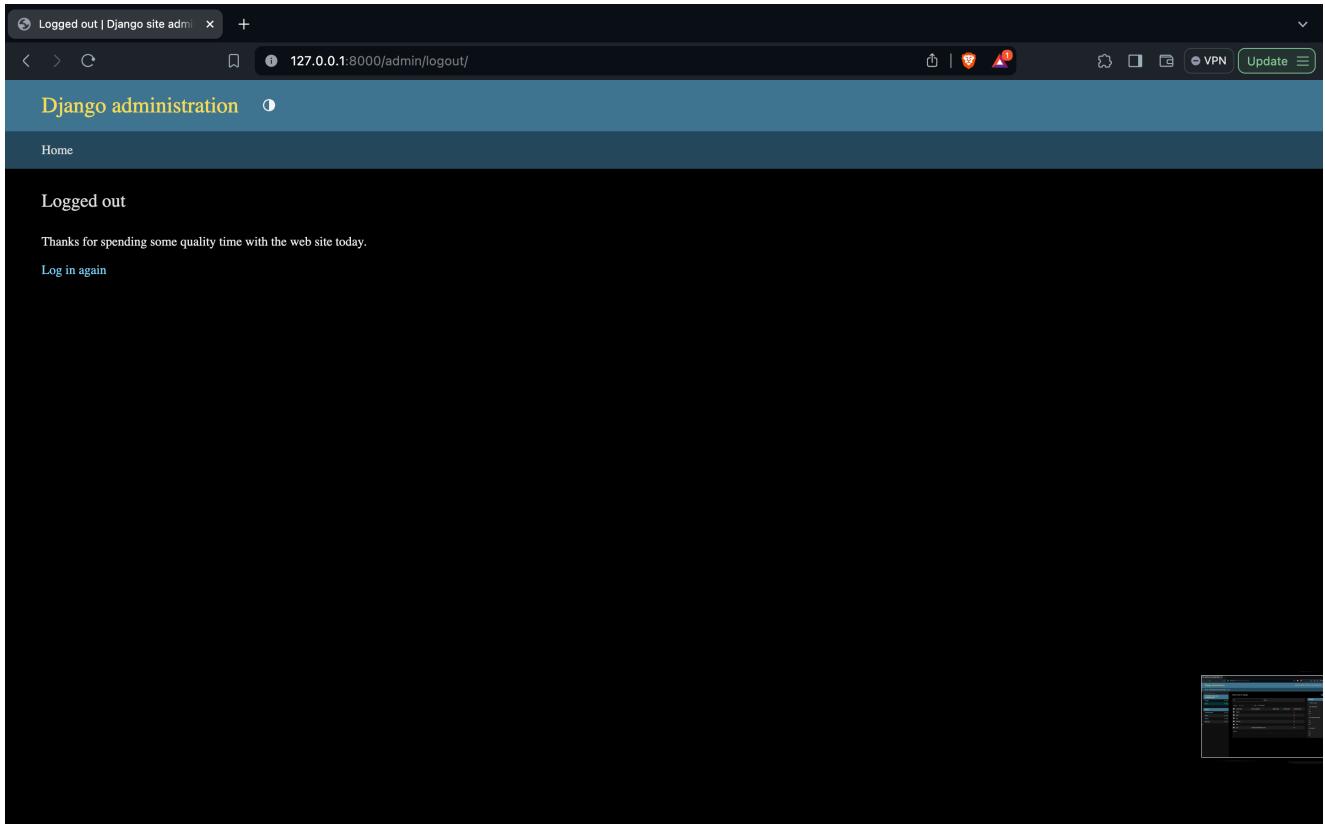
| Action: | USERNAME | EMAIL ADDRESS | FIRST NAME | LAST NAME | STAFF STATUS |
|--------------------------|----------|---------------------------|------------|-----------|-------------------------------------|
| <input type="checkbox"/> | Max01 | | | | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | adri | | | | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | anu | | | | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | anurrag | | | | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | nate | | | | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | root | nayakanurag2003@gmail.com | | | <input checked="" type="checkbox"/> |

6 users

FILTER

- Show counts
- By staff status
 - All
 - Yes
 - No
- By superuser status
 - All
 - Yes
 - No
- By active
 - All
 - Yes
 - No

4.1.11 Managing and Editing students list



4.1.12 Logout

4.2 Challenges

Throughout the development of the Distributed Systems Information platform, several significant challenges were encountered. These challenges ranged from technical hurdles to design considerations, each requiring a thoughtful approach to resolution.

Scalability and Performance: One of the primary challenges was ensuring that the platform could scale effectively to accommodate a large number of users without degradation in performance. As user base and data volume grew, it became crucial to implement strategies that would allow the system to handle increased loads efficiently.

- **Solution:** To address scalability and performance issues, the project team optimized database queries and implemented caching mechanisms to reduce load times. Additionally, the application architecture was designed to be stateless, allowing for easier scaling horizontally by adding more servers to distribute the load.

User Authentication and Security: Ensuring robust security measures for user authentication and data protection was another significant challenge. With the platform handling sensitive educational data, it was imperative to implement a secure authentication system and protect user data from potential breaches.

- **Solution:** The team utilized Django's built-in authentication system, which provided strong security features out of the box, including hashed passwords and CSRF (Cross-Site Request Forgery) protection. SSL/TLS encryption was enforced for all data in transit, and rigorous access control measures were put in place to ensure that users could only access data relevant to their roles.

User Interface (UI) Usability: Creating an intuitive and user-friendly interface was essential for ensuring that both teachers and students could easily navigate the platform and access its features. Designing a UI that was both functional and aesthetically pleasing presented a challenge, particularly in accommodating a wide range of user preferences and abilities.

- **Solution:** The team adopted a user-centered design approach, conducting usability testing with real users to gather feedback on the UI design. This feedback informed iterative improvements to the interface, focusing on simplifying navigation and ensuring that key functionalities were easily accessible. Responsive design principles were also applied to ensure that the platform was accessible across various devices and screen sizes.

4.3 Bugs

During the project lifecycle, several bugs were identified and addressed. These ranged from minor user interface issues to more complex functional bugs affecting the platform's operation. Below are examples of bugs encountered, their impact, and the resolution process:

Session Timeout Issue: Users were unexpectedly logged out of the platform after a short period of inactivity, leading to data loss and frustration.

- **Impact:** This bug significantly affected user experience, particularly for students engaged in long reading sessions or teachers entering grades.

- **Resolution:** The session timeout settings were adjusted, and users were given the option to extend their sessions if they were still active. This solution balanced the need for security with usability.

File Upload Size Limit: Teachers experienced issues when trying to upload large files, such as high-resolution videos or extensive lecture materials, due to a file size limit that was too restrictive.

- **Impact:** This limitation hindered the ability of teachers to share comprehensive resources with students, affecting the quality of available educational content.
- **Resolution:** The file size limit was increased, and additional compression and optimization techniques were implemented for uploaded files to manage storage more effectively without sacrificing quality.

Incorrect Grade Calculations: A bug in the grading module led to incorrect calculations of students' final grades under certain conditions.

- **Impact:** This bug had a direct impact on the accuracy of academic records, potentially affecting students' academic progress and perceptions of fairness.
- **Resolution:** The grading module was thoroughly reviewed and debugged, ensuring that all grade calculations were performed accurately. Comprehensive tests were added to cover a wide range of grading scenarios, preventing similar issues in the future.

By addressing these challenges and bugs, the project team was able to enhance the platform's functionality, security, and user experience, ensuring that it effectively met the needs of its users.

5. Discussion

5.1 Interpretation of Results and Implications

The development and implementation of the Distributed Systems Information platform have yielded significant insights into the application of distributed systems principles in an educational context. The platform

successfully facilitated a dynamic and interactive learning environment for teachers and students, demonstrating the potential of digital solutions to enhance educational processes. The implementation of distinct user roles with tailored functionalities addressed the specific needs of each user group, thereby improving the efficiency and effectiveness of teaching and learning activities.

Implications: The positive outcomes of this project underscore the transformative potential of technology in education. By leveraging distributed systems, educational institutions can achieve greater scalability, flexibility, and accessibility in their offerings. This project exemplifies how digital platforms can support the shift towards more personalized and engaging learning experiences. Furthermore, the insights gained from this project highlight the importance of considering scalability, security, and user experience in the design of educational technologies. The challenges and solutions encountered provide valuable lessons for future projects in this domain, emphasizing the need for ongoing innovation and adaptation to meet the evolving needs of educators and learners.

5.2 Addressing Challenges

The process of addressing the challenges faced during the project, such as scalability, security, and usability, has been instrumental in refining the platform and ensuring its success. These challenges highlighted the complexities involved in developing digital educational tools and underscored the importance of a flexible and iterative development approach.

Learning from Challenges: Overcoming these challenges required a combination of technical expertise, creative problem-solving, and user-centric design thinking. The experience gained from navigating these issues has not only enhanced the project team's capabilities but also contributed to a deeper understanding of the key factors that influence the effectiveness of digital learning platforms. These learnings can inform future projects, guiding the development of more resilient, user-friendly, and impactful educational technologies.

5.3 Resolution of Bugs

The identification and resolution of bugs throughout the development process played a crucial role in ensuring the platform's reliability and usability. This iterative process of testing, identifying issues, and refining the platform was essential for maintaining high standards of quality and functionality.

Continuous Improvement: The approach taken towards bug resolution reflects the project team's commitment to continuous improvement and excellence. Each bug encountered was treated as an opportunity to not only fix a specific issue but also to enhance the platform's overall robustness and user experience. This proactive stance on bug resolution underscores the importance of thorough testing and user feedback in the development of complex systems. It also highlights the team's dedication to delivering a platform that meets the highest standards of quality and reliability, ensuring that it effectively supports the educational goals of its users.

6. Conclusion

The Distributed Systems Information platform, developed using Django, marks a significant advancement in the realm of educational technology. By integrating distributed systems principles into a digital platform for education, this project successfully addresses the complex needs of teachers and students in a modern learning environment. The achievements of the project can be highlighted as follows:

- **Enhanced Educational Process:** Through the implementation of separate access for teachers and students, along with a suite of administrative and educational tools, the platform has significantly streamlined the teaching and learning process. It has facilitated a more efficient distribution of materials, more effective communication, and a more personalized learning experience for students.
- **Scalability and Flexibility:** By leveraging the principles of distributed systems, the platform ensures scalability and flexibility,

accommodating a growing number of users and evolving educational requirements without compromising performance.

- **Increased Accessibility:** The platform makes educational resources more accessible, breaking down geographical and temporal barriers to learning and thus contributing to a more inclusive educational environment.

This project demonstrates the potential of combining distributed systems with educational technology to create powerful tools that enhance the educational experience for both teachers and students. It underscores the importance of adopting a user-centric approach in the design and development of such technologies, ensuring they meet the specific needs of their intended users.

7. Future Scope

Looking ahead, there are several avenues for enhancing the platform and extending its capabilities:

- **Integration with Other Educational Tools:** Integrating the platform with other educational technologies and tools, such as virtual labs, simulation software, and e-libraries, could provide a more comprehensive learning environment for students.
- **Artificial Intelligence (AI) and Machine Learning (ML) Applications:** Incorporating AI and ML could personalize the learning experience further, offering adaptive learning paths based on individual student performance and preferences.
- **Enhanced Data Analytics:** Advanced analytics features could provide teachers with deeper insights into student performance and engagement, enabling more informed teaching strategies and interventions.
- **Blockchain for Certification and Record Keeping:** Implementing blockchain technology could offer secure and immutable record-keeping for grades and certifications, enhancing the credibility and verification process of academic achievements.

- **Increased Interactivity:** Further development could focus on increasing interactivity within the platform, such as implementing real-time feedback mechanisms, live polls, and interactive discussion forums.

These potential enhancements not only aim to increase the platform's effectiveness and utility but also anticipate the future needs and challenges of digital education, ensuring that the platform remains at the forefront of educational technology.

8. References

The development of the Distributed Systems Information platform was supported by a wealth of resources that provided both foundational knowledge and specific technical guidance. While an exhaustive list of all materials consulted would be extensive, key references include:

- **Django Documentation:** The primary guide for the Django framework, offering comprehensive information on its features, capabilities, and best practices.
- **Python Documentation:** For Python-specific programming concepts and techniques used in conjunction with Django.
- **MDN Web Docs (Mozilla Developer Network):** For web technologies and API references that were integral to the front-end development of the platform.
- **W3Schools:** As a supplementary resource for HTML, CSS, and JavaScript tutorials.

