A

Project Report

on

Development of Next-Gen Meeting Portal

Submitted to

Sant Gadge Baba Amravati University, Amravati

Submitted in partial fulfilment of

the requirements for the Degree of

Bachelor of Engineering in

Computer Science and Engineering

Submitted by

Ankush Deepak Bhonde
(PRN: 213120359)

Bhuvnesh Hariom Kale
(PRN:213120362)

Gaurav Rajendra Dhale
(PRN: 213120210)

Gaurav Vijay Kaple
(PRN:213120327)

Under the Guidance of

Dr. R. A. Zamare

Assistant Professor , CSE Dept.

Department of Computer Science and Engineering

Shri Sant Gajanan Maharaj College of Engineering,

Shegaon – 444 203 (M.S.)

Session 2024-2025

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,
SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Ankush Deepak Bhonde, Mr. Bhuvnesh Hariom Kale, Mr. Gaurav Rajendra Dhale and Mr. Gaurav Vijay Kaple** students of final year Bachelor of Engineering in the academic year 2024-25 of Computer Science and Engineering Department of this institute have completed the project work entitled "**Development of Next-Gen Meeting Portal**" and submitted a satisfactory work in this report. Hence recommended for the partial fulfilment of degree of Bachelor of Engineering in Computer Science and Engineering.

Dr. R. A. Zamare
Project Guide

Dr. J. M. Patil
Head of Department

Dr. S. B. Somani
Principal
SSGMCE, Shegaon

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,
SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Ankush Deepak Bhonde, Mr. Bhuvnesh Hariom Kale, Mr. Gaurav Rajendra Dhale and Mr. Gaurav Vijay Kaple** students of final year Bachelor of Engineering in the academic year 2024-25 of Computer Science and Engineering Department of this institute have completed the project work entitled "**Development of Next-Gen Meeting Portal**" and submitted a satisfactory work in this report. Hence recommended for the partial fulfillment of degree of Bachelor of Engineering in Computer Science and Engineering.

Internal Examiner

Dr. R. A. Zamare
Name and Signature
Date: 9/5/25

External Examiner

A. D. Shaikh
Name and Signature
Date: 9/5/25

Acknowledgement

It is our utmost duty and desire to express gratitude to various people who have rendered valuable guidance during our project work. We would have never succeeded in completing our task without the cooperation, encouragement and help provided to us by then. There are a number of people who deserve recognition for their unwavering support and guidance throughout this report.

We are highly indebted to our guide **Dr. R. A. Zamare** for her guidance and constant supervision as well as for providing necessary information from time to time. We would like to take this opportunity to express our sincere thanks, for his esteemed guidance and encouragement. His suggestions broaden our vision and guided us to succeed in this work.

We are sincerely thankful to **Dr. J. M. Patil** HOD, CSE Department, SSGMCE, Shegaon, and to **Dr. S. B. Somani** Principal, SSGMCE, Shegaon who always has been kind to extend their support and help whenever needed.

We would like to thank all teaching and non-teaching staff of the department for their cooperation and help. Our deepest thank to our parents and friends who have consistently assisted us towards successful completion of our work.

Ankush Deepak Bhonde (22BD310736)
Bhuvnesh Hariom Kale (22BD310751)
Gaurav Rajendra Dhale (22BD310749)
Gaurav Vijay Kaple (22BD310740)

Table of Contents Page Number

Abstract

The Meeting Portal is a web-based platform designed to streamline the gathering, organization, and retrieval of documents. Unlike traditional meeting management systems that focus primarily on scheduling and discussions, Meeting Portal emphasizes document security, role-based access control, and automated agenda creation. It features a hierarchical structure where Master-admins can oversee multiple departments (Funds), each managed by an administrator responsible for assigning user roles. The platform enhances document management and ensures compliance through calendar-based scheduling, automated notifications, and submission tracking. Built with Django and React.JS, this scalable and user-friendly platform improves document traceability, accessibility, and collaboration. Key advancements include enhanced organizational collaboration, secure document handling, and a fund-based hierarchy.

Keywords: *Agenda, Document, Meeting, Fund, Role-Based Access.*

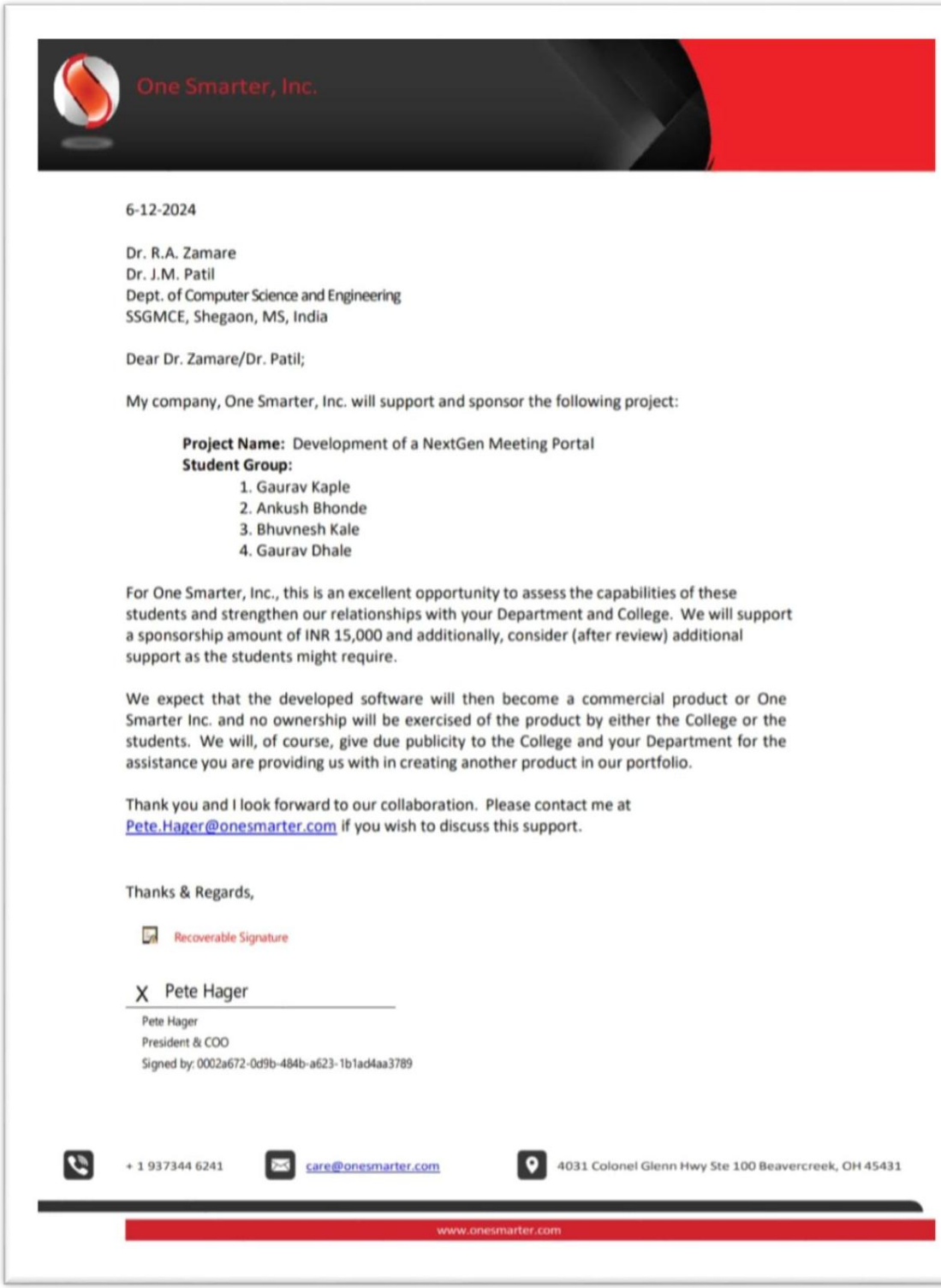
List of Abbreviations

Abbreviation	Description
API	Application Programming Interface
CORS	Cross-Origin Resource Sharing
DRF	Django REST Framework
JWT	JSON Web Token
ORM	Object-Relational Mapping
RBAC	Role-Based Access Control
UAT	User Acceptance Testing
UI	User Interface
UX	User Experience

List of Figures

List of Screenshots

Sponsorship letter



One Smarter, Inc.

6-12-2024

Dr. R.A. Zamare
Dr. J.M. Patil
Dept. of Computer Science and Engineering
SSGMCE, Shegaon, MS, India

Dear Dr. Zamare/Dr. Patil;

My company, One Smarter, Inc. will support and sponsor the following project:

Project Name: Development of a NextGen Meeting Portal
Student Group:

1. Gaurav Kaple
2. Ankush Bhonde
3. Bhuvnesh Kale
4. Gaurav Dhale

For One Smarter, Inc., this is an excellent opportunity to assess the capabilities of these students and strengthen our relationships with your Department and College. We will support a sponsorship amount of INR 15,000 and additionally, consider (after review) additional support as the students might require.

We expect that the developed software will then become a commercial product or One Smarter Inc. and no ownership will be exercised of the product by either the College or the students. We will, of course, give due publicity to the College and your Department for the assistance you are providing us with in creating another product in our portfolio.

Thank you and I look forward to our collaboration. Please contact me at Pete.Hager@onesmarter.com if you wish to discuss this support.

Thanks & Regards,

Recoverable Signature

X Pete Hager

Pete Hager
President & COO
Signed by: 0002a672-0d9b-484b-a623-1b1ad4aa3789

+ 1 937344 6241 care@onesmarter.com 4031 Colonel Glenn Hwy Ste 100 Beavercreek, OH 45431

www.onesmarter.com

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1 PREFACE

The Meeting Portal is a cutting-edge web-based solution for meeting document organization and management in contemporary businesses. In the fast-paced business world of today, plain scheduling applications or standalone digital programs prove to be insufficient in meeting the complex organizational needs. Legacy systems are shallow in ensuring sensitive data, traceability, and collaborative efficiency. In response, the Meeting Portal is powered by a solid Django backend and an extremely interactive React.JS frontend. This architecture not only ensures operational integrity and scalability but also ensures an easy-to-use interface for securely and efficiently managing lots of meeting-related documentation.

At its core, the Meeting Portal is not just a scheduling tool—it is a centralized command center for meeting document life cycle management. It introduces advanced features like role-based access control (RBAC) and automated agenda generation. With RBAC, each user—Masteradmin, Admin, or User—is granted tailored access rights, and thus confidential data is safeguarded and actions can be traced to specific roles. The automated agenda generation takes submitted documents and creates concise meeting agendas, reducing administrative workload and ensuring each meeting is well-prepared. This new system, therefore, not only outperforms the shortcomings of conventional meeting solutions but also sets the bar higher on secure, collaborative, and efficient document management.

1.2 MOTIVATION

The fast speed of digital changes in organizational management has put the shortcomings of simplistic or manual computerized tools in the limelight. The majority of organizations had been using paper-based approaches or basic applications that were time-consuming, error-prone, and inefficient. The Meeting Portal was created as a solution to these issues. Contemporary companies require systems that enable effortless communication, offer real-time online access to essential documents, and safeguard sensitive information without sacrificing efficiency.

The information explosion in the digital age, along with the need to make informed decisions swiftly, necessitates an integrated platform that is able to bring together document management. It is in this regard that the Meeting Portal fulfills the requirements by simplifying the document submission, review, and consolidation process. It replaces disparate systems—everything from multiple email threads to network folders shared across groups—into one secure location reducing delays and miscommunications. Furthermore, as businesses continue to be held accountable for data management practices, the need for a compliant, user-friendly, and secure platform has never been stronger. The Meeting Portal is therefore a timely solution, closing the gap between the past practices and current digital imperatives.

1.3 PROBLEM STATEMENT

In organizations, meetings like Financial Review Meetings and Board Meetings need significant documents like Financial Statements and User Audit Reports from various participants. But handling and accessing these documents in an efficient manner can be difficult. Our project is intended to consolidate all these documents into a single place so that they can be accessed easily during meetings.

1.4 AIM & OBJECTIVES

The primary goal of the Meeting Portal project is to offer a secure, efficient, and complete web-based application with emphasis on meeting-related documents management. To achieve this, the project has a couple of special goals in place:

a) To choose proper programming languages, frameworks, and set up the technology stack.
b) To train staff members and create the database.
c) To build core foundation modules: Funds, Materials, User Types.
d) to create user interface layouts with Figma.
e) To create main system modules: Users, Meetings, and Notes.
f) To integrate and combine all the uploaded meeting documents and publish the final document.
g) To create final system elements: Interactive Dashboard, Settings, and Notifications, and designate secondary and backup users for task management.

h) To perform thorough testing of the application and release it in a production environment.

Through the achievement of these objectives, the Meeting Portal is designed to avoid duplications, foster cooperation, and maintain high security and traceability levels at all phases of meeting document management.

1.5 SCOPE OF PROJECT

The scope of the Meeting Portal has been specifically broad to encompass the entire meeting document life cycle of an organization, ranging from initial meeting and consideration of meeting papers to final consolidation and post-meeting follow-up. The portal has been structured to deliberately highlight key functionalities without the inclusion of non-critical features such as embedded video conferencing or advanced AI-based document analysis that may distract it from its core purpose.

Major modules in the system are:

a) Dashboard and Overview: Real-time activity dashboard giving an overview of meeting status, impending deadlines in the near future, and documents submitted recently.

b) Scheduling and Calendar Management: Integrated functionality for managing meeting calendars, monitoring events, and synchronizing availability across disparate groups.

c) Notification and Reminder Systems: Reminders are ensured by automated reminders, making all stakeholders timely aware of deadlines for submissions and meeting schedules.

d) Document Collaboration Tools: Live notetaking, revisions, and secure document sharing features to facilitate collaboration.

e) Administrative Controls: Special modules for material definition, user role definition, fund management, and automatic agenda generation.

f) Audit Trail and Tracking: Complete record of all user activity and document modification to support compliance activities and accountability.

The project is implemented in a well-set organizational structure where user roles reflect the hierarchy and functional needs of the business. The clearly defined difference

between Masteradmin, Admin, and User roles is essential not only for security but also for optimizing the workflow efficiency by linking responsibilities to privileges.

CHAPTER 2

LITERATURE

REVIEW

LITERATURE REVIEW

A detailed plan and execution of a Meetings Document Management and Retrieval System are presented by Obagbuwa et al. (2012), which act as a reference point for the MeetingPortal project. The importance of proper handling of documents in meeting scenarios is emphasized, relating to the aim of creating a centralized platform for gathering and maintaining documents. The authors refer to document retrieval and organization problems within organizations, including the inability to easily access relevant documents and the possibility of miscommunication among team members. This is in line with MeetingPortal's systematized way of handling documents through its Materials and Assignments modules for easy retrieval and submission of documents. With role-based access added, MeetingPortal simplifies document management in such a way that users may efficiently retrieve and send necessary documents based on the role assigned to them. Management of documents as well as document retrieval is necessary to make any system for managing meetings successful, as they affect the productivity as well as user collaboration, and ultimately lead to decision-making process effectiveness in companies [1].

In "Meetings through the cloud," Saranga et al. (2023) explore privacy-preserving scheduling on mobile devices. The research has direct application in MeetingPortal's Calendar and Notifications modules that will enable users to schedule meetings with privacy protection for the user as well as security for data. The authors provide a cloud-based solution to address the problems of scheduling in a mobile environment, such as real-time updates and secure access to meeting information. This can be customized to enhance the MeetingPortal's functionality by safeguarding sensitive information about meetings and filing of documents. With the era of digital technology in which data breaches are prevalent, and customers are privacy-savvy, incorporating privacy-protection techniques can enhance user trust and compliance with data protection legislation. This enhances MeetingPortal as a preferable option for organizations concerned with security in their business [2].

A review of web-based meeting scheduling software is given by Thalawattha and Vidanagama (2021), listing a number of methods and technologies used in this field. Those results can be applied to guide the design of MeetingPortal, particularly in the Meetings module, where planning for document collection meetings is paramount. The survey also identifies the most significant characteristics that will improve user experience, including easy interfaces, automatic reminders, and synchronization with current calendar systems. These features can be integrated into the MeetingPortal so that the scheduling process is easy, and the users can schedule meetings and manage their work easily. According to this survey, MeetingPortal can increase its efficiency and usability, such that users will be able to handle the scheduling process easily and remain informed of their work. User experience emphasis in scheduling programs is crucial because it directly affects user interaction and satisfaction, which are vital to the success of the MeetingPortal [3].

The development of a Document Management System for a private cloud infrastructure is described by Kao and Liu (2013), which is very similar to the MeetingPortal's architecture. Their emphasis on cloud-based document management systems indicates the scalability and accessibility potential that is so critical to the MeetingPortal's role-based access model. The authors explain the benefits of using a private cloud as a document management system, such as enhanced security, handling sensitive information, and handling multiple users simultaneously. Using similar principles, the MeetingPortal can provide the assurance that documents are safely stored and readily accessible to authorized users, thereby facilitating different user roles to collaborate effortlessly. This security and accessibility focus is crucial in organizations handling sensitive information, as it minimizes risk from data breaches and misuse, and it ensures the integrity of the documents is maintained [4].

Emphasizing the automation of business processes through automatic integration with document management systems, Djedović et al. (2016) provide valuable insights for MeetingPortal's productivity. Their research emphasizes the importance of automating document workflows to minimize manual intervention and maximize productivity. This concept directly relates to the MeetingPortal's Assignments and Notifications modules, where automated reminders and document submission can significantly influence user participation and compliance. By implementing

automated processes, the MeetingPortal can automate the workflow process involved in document collection and meeting preparation, thus leading to more efficient organizational processes. Document management automation is a new trend that can lead to tremendous time savings and improved precision in document handling, essential for organizations that seek higher efficiency in their operations [5].

Examining hierarchically-structured document fusion in workflow systems, Badouel and Tchoupé Tchendji (2008) offer knowledge that can be used with the MeetingPortal approach to document collection and production of the final report. Document merging strategies that can be gleaned from their studies can be applied to the MeetingPortal's feature, particularly in the construction of the final document with the agenda and materials submitted. By employing strong merging strategies, MeetingPortal can provide an end output that is well-coordinated and well-organized, thereby enhancing meeting documentation usability and understanding for all concerned stakeholders. Effective document merging is highly necessary for organizations whose role is aggregating reports and meeting minutes since it conserves time and reduces opportunities of making mistakes on the final reports. This feature is particularly applicable in the instance of the MeetingPortal, where accurate documentation is key to maintaining a clear record of meetings and decisions made [6].

Organizational meeting management and the benefits of organizational meetings are discussed by Bagire et al. (2015) as they provide a broader perspective to the MeetingPortal's goals. Their analysis of meeting management also highlights the role of disciplined processes in achieving meeting goals, which aligns with MeetingPortal's design on the basis of role-based access and document repository. Through the respect for the dividends in properly organized meetings, MeetingPortal can better serve its customers by offering structured document submissions and open communication between the participants. This focus on structured meeting management is key to making meetings productive and keeping all participants focused on goals and outcomes. The conclusions of this study can be used to design features that optimize user interaction and workflow optimization, resulting in more efficient meetings [7].

The constant endeavors of Djedović et al. (2016) reaffirm the importance of process optimization by business processes through document management systems, casting even stronger shadows on the relevance of automation within the MeetingPortal. Their conclusions on the integration of document management and business processes can impact the development of features that optimize user interaction and streamline workflows. With a focus on automation and integration, the MeetingPortal can optimize overall efficiency such that users can navigate the system and perform their functions effectively. The emphasis on business process simplification is necessary for organizations that aim to maximize operational effectiveness and cost benefits in handling documents manually, and hence the MeetingPortal proves to be a valuable addition to modern-day organizations [5].

Joining the discussion on blockchain-driven document management systems and IPFS, Saranga et al. (2023) present an alternative method of delivering data integrity and security. This concept can be particularly beneficial to the MeetingPortal, which is interested in providing confidentiality and authenticity of documents presented by users. Through the study of blockchain technology implementation, the MeetingPortal can enhance its security features, ensuring that each document is tamper-proof and verifiable. It is even more essential in a time when the authenticity of the data is most important, and organizations need to verify that their documents are not subject to unauthorized manipulation. Integrating such newer technologies can place the MeetingPortal in the lead position of secure document management software [2].

Suggesting a secure document distribution scheme, Yang et al. (2021) emphasize the importance of secure document distribution. This research is relevant to MeetingPortal's role-based access and document management emphasis because it emphasizes the need for secure sharing in the platform. By embracing such avenues of sharing the documents, the MeetingPortal can ensure that sensitive information is only accessed by authorized users, thereby guaranteeing the integrity of the documents and privacy of the affected users. Secure document sharing is necessary for organizations handling confidential data, as it keeps the threat of data breaches and misuse at bay. This feature is essential in building user confidence and adherence to data protection legislation [9].

Negotiating skills in terms of planning meetings are addressed by Hossain (2012), which can be used for the MeetingPortal's user interface and document submission strategy. The results of this research can be applied to the Assignments and Notifications modules, where effective communication and negotiation are crucial for meeting document submission deadlines. By integrating negotiation tactics into the MeetingPortal design, the system can improve the collaboration among users, leading eventually to improved meeting results. Successful negotiation is critical within a collaborative environment, as it guarantees that all parties are in alignment in terms of goals and tasks, thereby increasing the general efficiency of the meeting process [10].

Hossain and Shakshuki (2013) propose a negotiation protocol for meeting scheduling agents, which can be integrated into the MeetingPortal's functions as a way of providing a structured method of interaction with the users. Their research centers on negotiation and effective communication as being needed in setting up meetings, which is also the MeetingPortal's aim to streamline document collection and collaboration. By integrating negotiation protocols, the MeetingPortal has the ability to increase user engagement and coordination among parties towards meeting purposes and roles. This formal approach to negotiation can lead to more efficient scheduling processes and better results for all concerned users, ultimately propelling the success of the MeetingPortal [11].

Swarm intelligence meeting scheduling is explored by Salem and Ben Hassine (2015), introducing new knowledge about scheduling process optimization. This can be a stimulus for the MeetingPortal's Calendar module, where intelligent scheduling algorithms can be applied to optimize the effectiveness of meeting bookings. Through the incorporation of swarm intelligence principles, MeetingPortal can strengthen its scheduling abilities so that the meetings are planned in a way that maximizes attendance and avoids conflicts between the users. Use of intelligent algorithms in scheduling is a new phenomenon that can create more effective and efficient management of meetings, hence making MeetingPortal a useful tool for organizations with the objective to automate their meeting process [12].

CHAPTER 3

METHODOLOGY

METHODOLOGY

3.1 RESEARCH DESIGN:

The research design of this project follows an applied research methodology for real-world problem-solving regarding the inefficiencies of conventional and digital meeting systems in educational or organizational environments. The study uses a blend of descriptive and experimental design methods to measure prevailing issues, determine system requirements, and validate the efficiency of developed web portal.

The descriptive aspect is concerned with identifying the difficulties encountered by users (students, administrators, and faculty) in handling meetings with traditional tools such as Zoom, Google Meet, and paper documentation. This includes literature review, interviews with stakeholders, and comparative analysis with current systems.

The experimental component consists of the creation and validation of a prototype solution based on current web technologies (Django, DRF, React.js, Bootstrap) and measuring its performance in terms of functionality, usability, and efficiency. This is done by following iterative design, development, testing, and deployment processes.

Prime Aims of the Design:

• To suggest a system of meeting management that is structured and role-based.
• To consolidate sharing of documents, notifications, and notes under a single portal.
• To maintain security, scalability, and user-friendly interfaces.
• To assess user experience by pilot testing.

The use of both qualitative and quantitative methods provides a complete overview of user needs and system efficiency.

3.2 DATA COLLECTION METHODS:

The methods of data collection employed in this study include both primary and secondary sources:

3.2.1 Primary Data Collection:

• User Interviews: Carried out with students, department heads, and academic staff to take firsthand information of prevalent meeting management challenges and aspirations from a digital platform.

• Online Questionnaires: A group of organized questions was distributed using Google Forms among users to obtain information regarding wished-for features, user difficulties, and desired things from a system based on portals.

• Observation: The staff watched real-time processes employed during meeting documentation as well as in communication to search for gaps as well as inefficiencies.

• Prototype Testing: In-person feedback was gathered during pilot operation of the application by terminal users.

3.2.2 Secondary Data Collection:

• Literature Review: Research articles, journals, and online articles were read to comprehend current technological solutions, system architecture best practices, and security measures.

• Comparative Analysis: Examined characteristics of tools such as Google Meet, Zoom, MS Teams, and internal academic platforms to determine disadvantages and benefits.

All information was recorded and analysed through tabulation and thematic coding to arrive at major design decisions for the system architecture and modules.

3.3 SYSTEM DEVELOPMENT APPROACH:

The system was developed following the Agile Software Development Life Cycle (SDLC) approach. Agile was selected as it is an iterative and incremental approach that supports frequent feedback, improved requirement management, and rapid adaptation to change.

3.3.1 Development Phases:

a) Requirement Gathering:
   a. Performed detailed stakeholder sessions.
   b. Identified roles: Master Admin, Admin (HoD), User.
   c. Determined key modules: Meetings, Assignments, Notes, Notifications.

b) System Design:
   a. UI/UX designs were prepared using Figma/Canva.
   b. Database schema and API flow diagrams were created.
   c. Diagrams utilized: ER diagram, Flowcharts, Sequence diagrams.

c) Development:

   a. Frontend created using React.js with modular components.
   b. Backend driven by Django and Django REST Framework.
   c. API communication managed using Axios and token-based authentication.

d) Testing:

   a. Unit testing using Django test cases.
   b. Integration testing using Postman.
   c. User acceptance testing through pilot deployment.

e) Deployment:

   a. Frontend deployed on Netlify; backend hosted on Railway.
   b. CORS, HTTPS, and production settings set.

f) Review & Iteration:

   a. Daily standups with users and mentors to shape features.
   b. Real-time changes made based on user input.

3.3.2 Tools & Technologies Used:

Category	Tool/Framework
Frontend	React.js, Bootstrap
Backend	Django, DRF, CORS
Database	SQLite3
Authentication	JWT-RestFramework
Dev Tools	Git, Github, Figma, VScode

Agile was successful in delivering working modules in regular intervals for feedback from users and ensured the system was up to real-time expectations.

3.4 IMPLEMENTATION AND EVALUATION:

3.4.1 Implementation Strategy:

The implementation was done in modular stages for easy testing and debugging. Each module (User Auth, Meetings, Assignments, Notes, Notifications) was built, tested, and deployed separately before integration.

3.4.2 Deployment Environment:

Development: Localhost with SQLite, Django runserver

- Production: Backend hosted on Railway with PostgreSQL, frontend hosted on Netlify using production builds from Vite
- CORS and middleware configured for secure cross-origin communication

Authentication Strategy:

- JWT Authentication using SimpleJWT for API calls
- Session Authentication for admin panel login
- Securely stored tokens using Universal-Cookie on client side

3.4.3 Role-Based Access:

a) Admin and Master Admin view extra controls for user management.
b) Normal users have restricted access based on fund and meeting association.
c) Conditional rendering applied on frontend, Django permissions on backend.

3.4.4 Evaluation:

To assess the performance and acceptance of the implemented portal, the following metrics and methods were utilized:

a) Functionality Testing:
    a. Verified all CRUD operations on Meetings, Notes, Assignments.
    b. Checked file uploads, downloads, and session validation.

b) Usability Testing:
    a. Carried out feedback sessions with test users to determine ease of use and task fulfilment.
    b. Iterative UI enhancements based on feedback.

c) Performance Testing:

   a. Checked response times of API calls using browser developer tools and Postman.
   b. Checked login, dashboard load times under sample data.

d) Security Testing:

   a. Confirmed JWT expiration and refresh.
   b. Blocked unauthorized access to the API via permission checks.

e) Feedback Collection:

   a. Got positive feedback for usability, speed, and tidy interface.
   b. Some suggestions were to incorporate calendar integration and document history (not yet done)

CHAPTER 4

IMPLEMENTATION

IMPLEMENTATION

4.1 SYSTEM OVERVIEW

The "NextGen Meeting Portal" is a web-based, end-to-end solution crafted to automate meeting scheduling, documentation, and reminders in a multi-level organizational structure. The application is crafted with three primary user roles — Master Admin, Admin, and Users — that have different levels of access. With a blend of Django (Python) as the backend and React.js as the frontend, the portal offers a responsive and secured interface to all the stakeholders.

The architecture is client-server-based, with the client (React) interacting with the server (Django REST Framework) via secure, token-based APIs. Meeting management, document upload, notification, and task assignment features are designed modularity for scalability.

4.2 TECHNOLOGY STACK

The implementation of this system utilizes the following technologies:

Layer	Technology Used
Frontend	React.js, Bootstrap, Axios
Backend	Django, Django REST Framework (DRF), Django CORS
Authentication	JWT-RestFramework
Database	SQLite3
Tools	Vite, Postman, Git, Github ,Canva/Figma, Draw.io

Each layer is important in providing a secure, efficient, and user-friendly system. The technologies employed were selected based on their scalability, community support, developer experience, and performance in constructing strong web applications.

4.2.1 Frontend:

The frontend of the system handles the user interface and interaction. It is what users interact with directly, and therefore usability, responsiveness, and clarity are of prime concern.

a) React.js

React.js is an open-source, powerful JavaScript library created by Facebook to develop user interfaces. It enables developers to build complex web applications that can render and update efficiently according to changing data. React works on a component-based structure, meaning every component of the interface (such as a button, navbar, or sidebar) is considered an individual component. This enhances reusability, modularity, and code maintainability.

React uses a virtual DOM, which reduces direct interaction with the actual DOM and improves performance. The UI rendering becomes faster. In this project, the whole user interface, from login forms to dynamic dashboards and modal components, was built using React. The state and route management without class components was enhanced using React hooks such as useState, useEffect, and useNavigate.

b) Bootstrap

Bootstrap is a front-end framework developed by Twitter that provides ready-to-use components and utility classes for building responsive, mobile-first websites. It simplifies UI development by offering pre-styled buttons, forms, tables, and layout grids, allowing developers to focus more on logic than styling.

Bootstrap was employed in this project together with React for responsive and consistent layout. The grid system of Bootstrap played a critical role in designing layouts that transform elegantly across devices—laptops, tablets, and phones. It also sped up development of elements such as modals, alerts, tabs, and cards using few custom CSS.

c) Axios

Axios is a promise-based JavaScript HTTP client that is used for communicating with APIs. It makes asynchronous HTTP request sending to REST endpoints straightforward

and easy, and it easily handles requests such as GET, POST, PUT, and DELETE. Axios is capable of automatically transforming JSON data, and it is easier to read than the native fetch API.

In this system, Axios will be used in React components to make communication with Django REST APIs. Axios handles all the procedures that involve fetching data - logging in the user, submitting meeting forms, getting list assignments, and deleting notes. Axios interceptors were also used to attach JWT tokens to headers for secured access.

4.2.2 Backend:

Backend is responsible for business logic, database operations, and API returns. Backend makes sure data is consistent, validates user roles, and processes client requests securely.

a) Django

Django is a high-level web framework for Python that promotes rapid development and pragmatic, clean design. Django has the "batteries-included" philosophy, where it includes built-in modules for authentication, admin interface, database models, and routing.

Django's Model-View-Template (MVT) structure enabled us to declare data models in a clean manner using Python classes, process business logic within views, and deliver HTML pages where necessary. Its Object-Relational Mapping (ORM) facilitated database operations naturally without the need to write raw SQL queries. For example, retrieving meetings authored by a specific user could be performed easily using a simple query within the application.

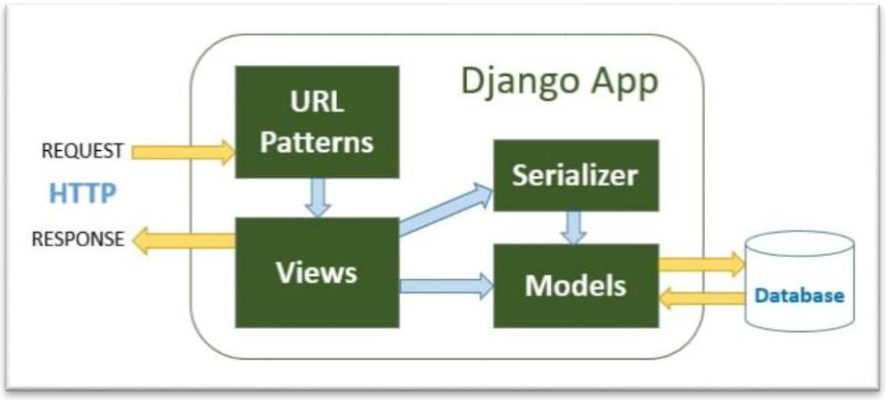


Figure 4.2.2.1 — Linkage within backend

b) Django REST Framework (DRF)

Django REST Framework is a strong toolkit for developing Web APIs in Django. DRF provides features such as serialization, authentication, filtering, pagination, and permission, making it easy for developers to expose models in the form of RESTful APIs.

In this framework, DRF was utilized for serializing models like Meeting, Assignment, Notification, and UserType. Class-based views (APIView, ListCreateAPIView) were heavily utilized for neater logic. DRF's TokenAuthentication was overridden by JWT for more secure access. Permissions such as IsAuthenticated and IsAdminUser assisted in applying role-based access control at the API level.

c) Django CORS Headers

Cross-Origin Resource Sharing (CORS) is a browser security feature that blocks cross-domain requests. Because our React frontend is running on a different domain/port than our Django backend, CORS needs to be enabled.

Django CORS Headers is a middleware that enables cross-origin requests. In this project, it was set to receive requests from Netlify (frontend host) to Railway (backend host). Without it, the frontend could not talk to the backend APIs.

4.2.3 Authentication Layer:

Security is the very essence of any web application, particularly those that handle sensitive information such as meetings and company roles.

a) JWT Authentication (SimpleJWT + DRF)

JWT (JSON Web Token) is a stateless, compact, token-based authentication scheme. It creates tokens upon login, which are saved on the client (typically in cookies or local storage) and prefixed to all requests to authenticate user identity.

We employed the SimpleJWT package with Django REST Framework to support JWT authentication. Upon successful login, an access and refresh token are delivered. The access token is used for normal requests, whereas the refresh token is used to receive a fresh access token upon expiration.

JWT provides secure communication, eliminates the overhead of session storage on the server, and is perfectly suited to single-page applications such as ours.

b) Security Features:

Role-based access control with Django permissions and user groups.

Token expiration management and secure use of cookies (with universal-cookie on the frontend).Token blacklisting on logout to avoid reuse.

4.2.4 Database Layer

a) SQLite3:

SQLite3 is a light-weight, server-less, stand-alone SQL database engine. It keeps the entire database in one file and has no setup requirements, which makes it perfect for prototyping and small projects.

During the development stage, SQLite3 was employed to manage all database operations—user creation to storage of meetings. Django's ORM facilitated smooth interaction without having to write raw SQL queries. While the system can be easily ported to PostgreSQL or MySQL for production, SQLite was an extremely efficient database for testing and internal deployment.

b) Database Models:

- UserType: Stores roles such as Admin, Master Admin, and User.
- Materials: Stores nomenclature of documents that generally used.
- Fund: Stores department details.
- Meeting: Stores meeting information, associated users, and documents.

Assignment, Notes, and Notification: Store respective entities that are associated with meetings.

4.2.5 Tools and Utilities

a) Vite

Vite is a new frontend build tool that greatly enhances development performance. Unlike other bundlers, Vite delivers source files as native ES modules during development, with more rapid updates and live reloading.

In our platform, Vite was utilized to create the React project, speed up build time, and configure environment variables. Vite's production-ready builds were deployed on Netlify for speedy, secure delivery.

b) Postman

Postman is a development and testing tool for APIs that makes it easy to send HTTP requests and inspect responses. It is capable of saving collections of requests, specifying environments, and testing authentication pipelines.

We utilized Postman heavily in the backend development stage to test endpoints, verify response schemas, stub authenticated requests using JWT, and verify API behavior prior to frontend integration.

c) Git and GitHub

Git is a version control system based on distributed versions used to record changes in code. GitHub, a hosting platform in the cloud for Git repositories, supports collaborative development.

Git and GitHub were employed to:

- Ensure version control and prevent loss of code.
- Work across frontend and backend teams.
- Coordinate feature branches, pull requests, and code reviews.

d) Canva / Figma

Canva and Figma are UI/UX design software for wireframing and prototyping application designs. Canva was utilized for documentation visuals such as banners and diagrams. Figma was utilized to develop mockups of the user interface prior to implementation to ensure that they were aligned with stakeholder expectations.

Typical designs developed:

User dashboard mock-up, Role-based sidebar UI, Meeting form wireframe, Sequence and data flow diagrams.

e) Draw.io

Draw.io is a robust, free, and open-source web-based diagramming software that enables developers, designers, and project managers to graphically map out ideas, systems, workflows, and architectures. It is widely used in software development projects to draw system architecture diagrams, flowcharts, database schemas, network diagrams, and user interface mock-ups.

4.2.6 Frontend Implementation

Frontend is built in React.js with Vite for hot module replacement and fast bundling. The app is segmented into reusable components like:

- Navbar.jsx: Renders user information, role-based navigation
- Sidebar.jsx: Dynamic sidebar that renders links depending on the user role
- Dashboard.jsx: Retrieves user-specific data after login
- MeetingList.jsx / MeetingInfo.jsx: Renders meeting data from the backend
- Login.jsx: Fullscreen login page coupled with Axios and Universal-Cookie

Dynamic routing is handled through React Router DOM with authentication status and role-based protected routes. Conditional rendering is used to ensure that only authenticated users can view specific components.

UI Design uses a neat, responsive design with role-awareness for rendering. For example:

- Admin views Meeting Creation and Assignment buttons
- Users only view Join Meeting and Upload Docs sections
- Admin and users enjoy the functionality of real time communication through notes module.

4.2.7 Backend Implementation

The Django backend is organized into several apps for separation of concerns:

- users: handles login, registration, and roles
- meetings: handles meeting CRUD operations
- assignments: task creation, assignment, and completion tracking
- notifications: in-app notifications for meetings and assignments
- notes: meeting notes and real time communication between admin and users

The REST APIs are built using Django REST Framework with class-based and function-based views. Serializers guarantee structured and secure data exchange. Pagination and filtering are done using DRF's built-in tools.

Endpoints follow RESTful principles:

Endpoint	Method	Access	Description
/auth/login/	POST	Public	Authenticates user
/auth/logout/	POST	Auth	Logs out user
/dashboard/	GET	Auth	Fetches logged-in user data
/meetings/	GET/POST /DELETE	Admin	Upload meeting notes/documents
/assignments/	GET/POST	Admin	Task assignment to users
/notes/	GET/POST	Auth	Chat between admin and other users

4.2.8 API Communication & Authentication

Frontend-backend communication is done through Axios with token handling through Universal-Cookie. During login, a JWT token is obtained and stored in the cookie for session persistence.

a) Authentication Modes:

Session-based: For admin-level access in browser tabs

JWT (SimpleJWT): For API access and frontend interaction

b) Axios interceptors are set to:

Add Authorization headers to all API requests

Redirect user to login if token is invalid or expired

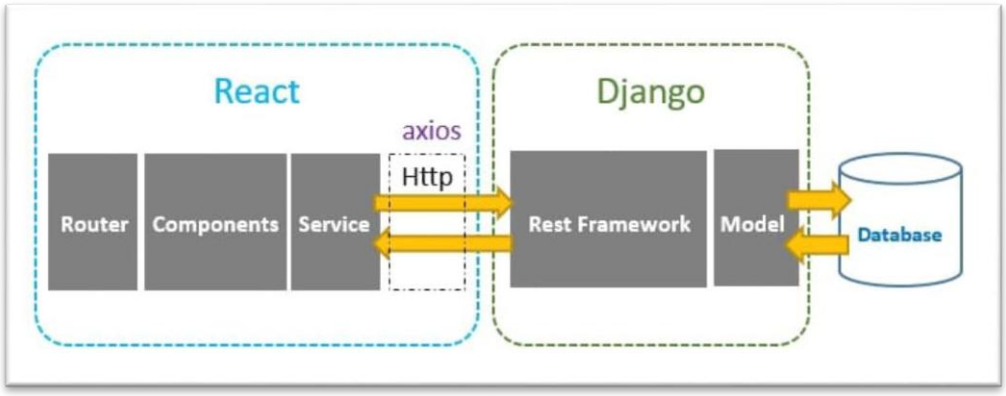


Figure 4.2.8.1 — Frontend-backend communication

4.2.9 Role-Based Access Control (RBAC)

One of the most important things about the system is preventing users from accessing features not allowed by their role:

- Master Admin: Can access creation of funds, admin assignment, have high level of access than admin and other users
- Admin: Can access creation of users, meeting and assignment management
- User: Only allowed for participation and submission of documents
- RBAC is implemented at:
- Backend: With DRF permissions and custom decorators
- Frontend: Through conditionally rendering components by role saved in context or cookie

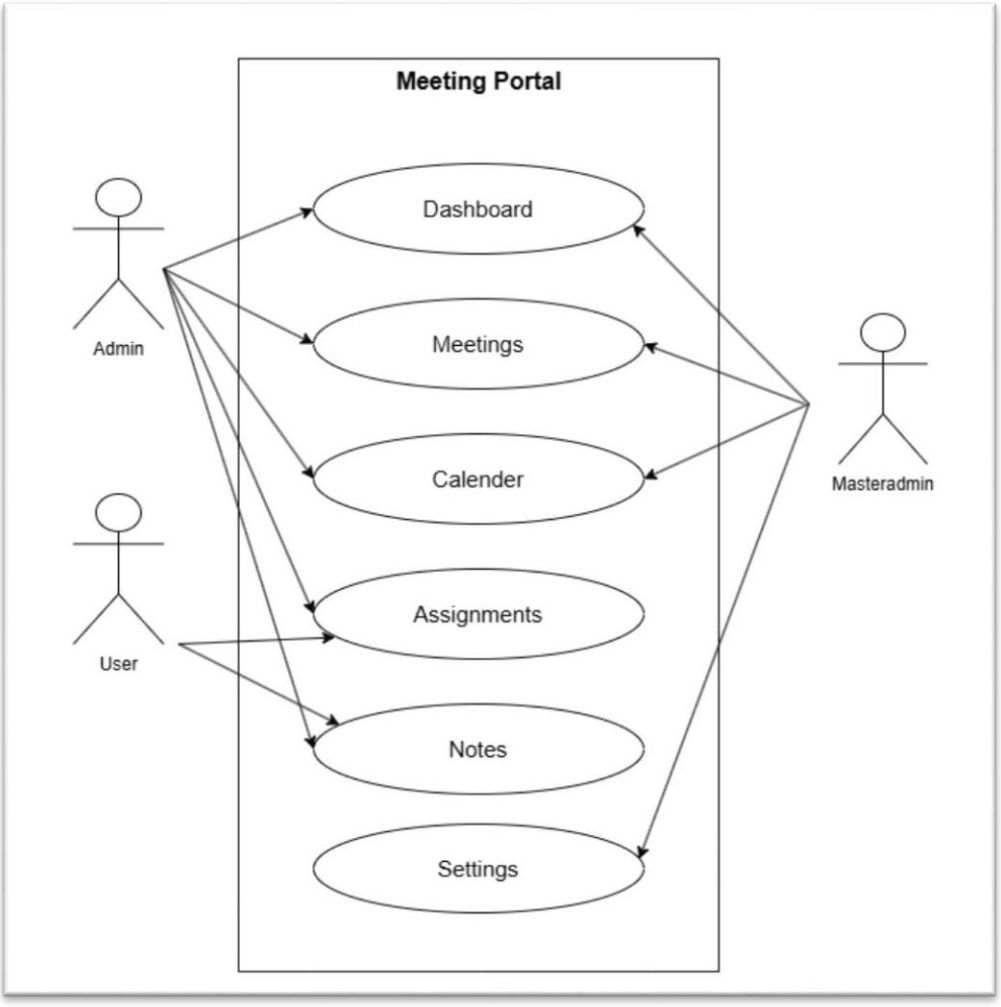


Figure 4.2.9.1 — Role Based Access

4.2.10 Database Design

The relational database schema involves the following major entities:

- User (username, email, role, fund_id)
- Fund (title, created_by, logo)
- Meeting (title, agenda, date, created_by, fund_id)
- Assignment (task_title, assigned_to, deadline, related_meeting)
- Notes (meeting_id, uploaded_by, doc_file)

4.2.11 Entity Relationship Diagram (ERD)

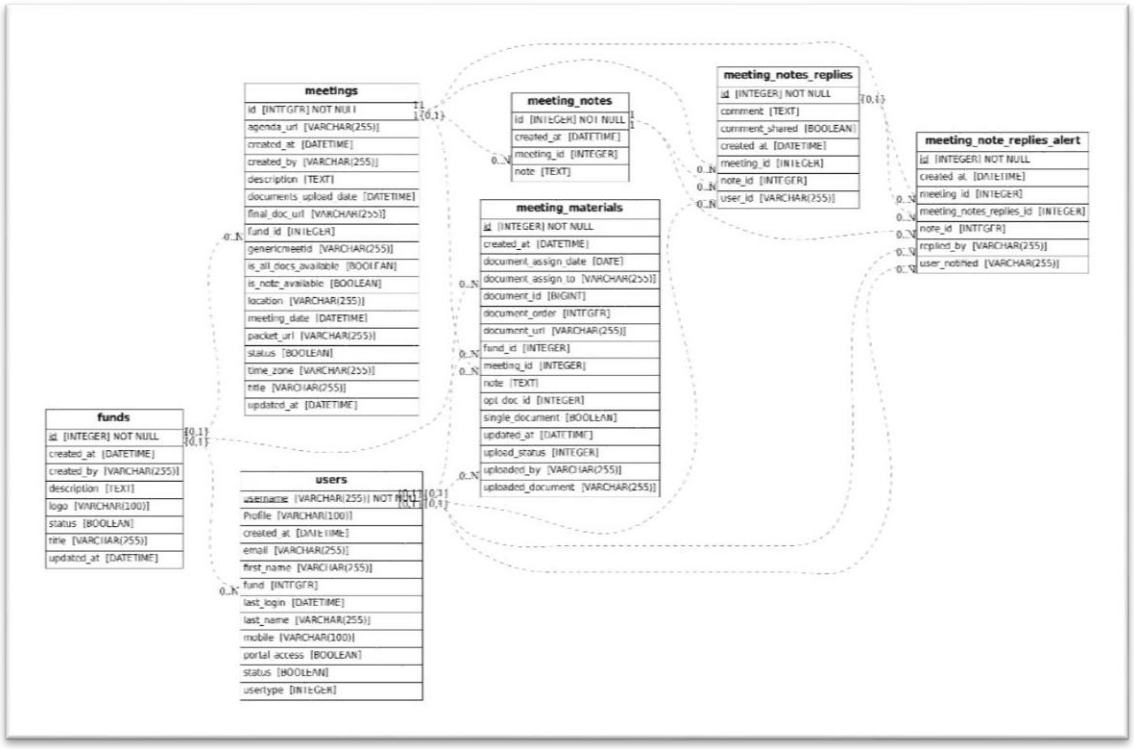


Figure 4.2.11.1 — Database Class Diagram

The Entity Relationship Diagram (ERD) is a critical component of our system design. It visually represents the logical structure of the database and defines how various entities in the system are related to one another. Our project—"Development of NextGen Meeting Portal"—involves multiple users interacting with meeting schedules, documents, and notes, all managed in a structured and secure manner.

The ERD captures the core components of the system including user management, fund (department) structure, meetings, meeting materials, meeting notes, and communication through replies. Below is a detailed explanation of each entity and the relationships that exist among them.

a) User

The User entity represents all the individuals interacting with the system. This includes MasterAdmins, Admins (HODs), and general Users.

Attributes:

- id: Primary key
- username, email, first_name, last_name, mobile
- usertype: Defines the role (e.g., Admin, User, MasterAdmin)
- fund_id: Foreign Key referencing the Fund entity
- portal_access: Indicates whether the user can access the portal
- status: Indicates whether the account is active
- created_at: Timestamp for account creation

Relationships:

- Each user is associated with a fund (department)
- A user can create or participate in multiple meetings
- A user can upload meeting materials
- A user can post replies to meeting notes

b) Fund

The Fund entity represents departments or organizational units within the system. Each fund is typically headed by an Admin and consists of multiple users.

Attributes:

- id: Primary key
- title, description
- logo: Image of the department logo
- status: Indicates if the fund is active
- created_by : User who created the fund
- created_at : Timestamp of creation

Relationships:

- A fund can have multiple users
- A fund can schedule multiple meetings
- A fund can host various materials associated with different meetings

c) Meeting

The Meeting entity contains information about scheduled meetings within a fund.

Attributes:

- id: Primary key
- title, description, meeting_date, location
- meeting_start_time, meeting_end_time
- meeting_type: Could be physical or virtual
- created_by: User who created the meeting
- created_at: Timestamp of meeting creation
- fund_id: Foreign Key referencing the Fund entity

Relationships:

- A fund can host multiple meetings
- A meeting can have several materials, notes, and replies
- A meeting is linked to a specific fund and created by a user

d) Meeting Material

The MeetingMaterial entity manages all documents related to a meeting.

Attributes:

- id: Primary key
- document_title, document_url
- uploaded_document: The actual uploaded file
- note: Description or comment about the document
- uploaded_by: User who uploaded the document
- uploaded_on: Date and time of upload
- meeting_id: Foreign Key referencing the Meeting entity
- fund_id: Foreign Key referencing the Fund entity

Relationships:

- Each meeting can have multiple associated materials
- Materials are linked to the user who uploaded them

e) Meeting Note

The MeetingNote entity captures notes taken during or after meetings.

Attributes:

- id: Primary key
- note: The content of the note
- meeting_id: Foreign Key referencing the Meeting entity
- created_at: Timestamp of creation

Relationships:

- A meeting can have multiple notes
- Notes may be replied to by various users

f) Meeting Notes Replies

The MeetingNotesReply entity enables collaborative communication by allowing users to reply to specific notes.

Attributes:

- id: Primary key
- comment: The reply text
- user_id: Foreign Key referencing the User entity
- meeting_id: Foreign Key referencing the Meeting entity
- note_id: Foreign Key referencing the MeetingNote entity
- created_at: Timestamp of reply

Relationships:

- Each note can have multiple replies
- Replies are linked to both the user and the original note

g) Meeting Note Reply Alerts

The MeetingNoteReplyAlert entity handles system notifications that alert users when a reply is made to their note.

Attributes:

- id: Primary key
- meeting_notes_replies_id: Reference to the reply made
- note_id: The note that was replied to
- replied_by: User who made the reply
- user_notified: User who received the alert

- created_at: Timestamp

Relationships:

- Alerts are triggered upon a new reply
- Each alert notifies the original note owner

Summary of Relationships

Entity	Relationship Description
User – Fund	Many users belong to one fund
Fund – Meeting	One fund can have many meetings
Meeting – Material	One meeting can include multiple materials
Meeting – Note	One meeting can include multiple notes
Note – Reply	One note can have multiple replies
User – Reply	One user can make multiple replies
Note Reply – Alert	One reply can trigger one or more alerts

Let me know if you'd like this in DOCX or PDF format, or if you'd like me to help generate a visual version of the diagram as well.

4.2.12 Testing & Deployment

Testing:

- Unit testing with Django's built-in test framework
- Postman used for API validation
- Frontend tested manually and with browser dev tools

Deployment:

- Frontend built using Vite and deployed via Netlify
- Backend deployed on pythonanywhere
- CORS configured for domain communication

CHAPTER 5

RESULT

AND

DISCUSSION

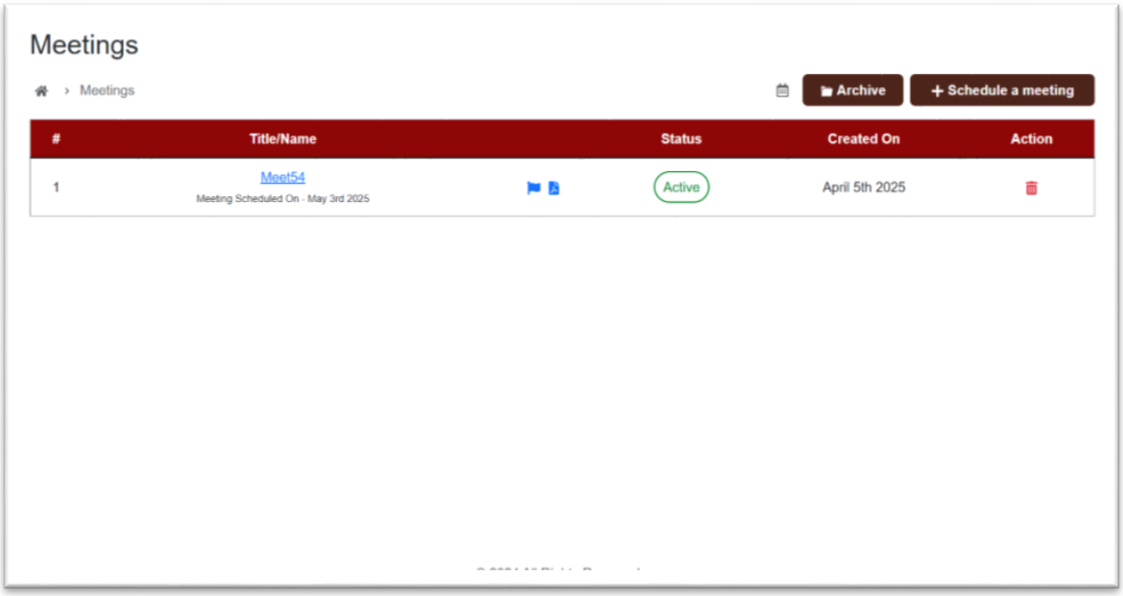
RESULT AND DISCUSSION

5.1 Role of Administrative Users in Meeting Creation

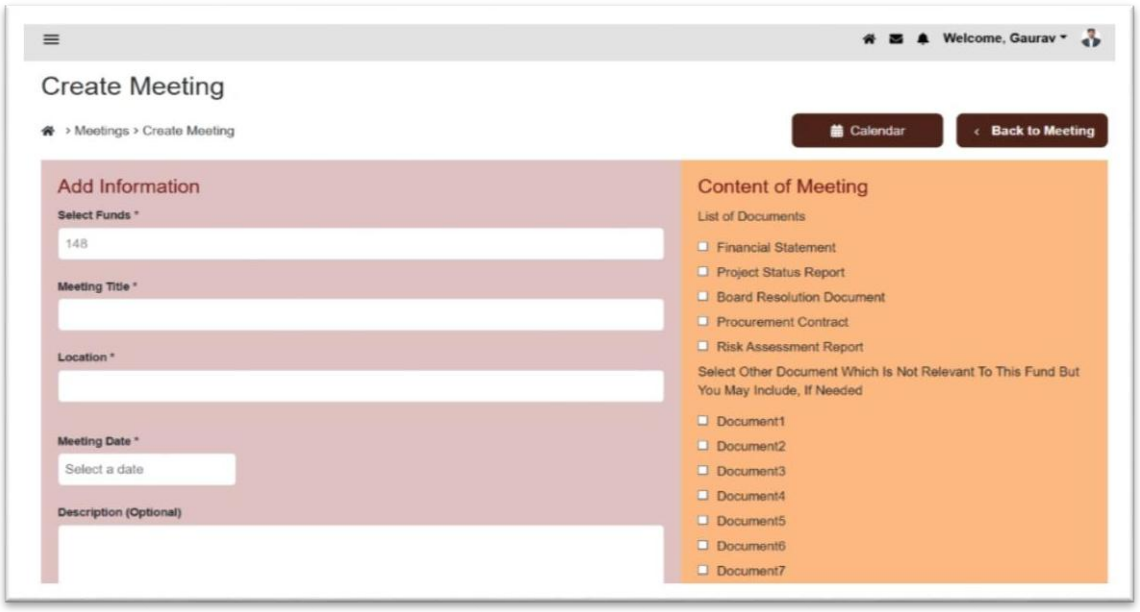
The system successfully implements a hierarchical user management structure where both the Masteradmin and Admin users hold the authority to initiate new meetings. This design aligns with the project's objective of decentralizing meeting operations while maintaining administrative oversight. Upon logging into the portal, these privileged users are presented with an intuitive interface that allows them to create a meeting by specifying parameters such as title, associated fund, date, time, and required documentation.

From a design perspective, the meeting creation module is visualized through a triplet of interface states: the initial meeting creation form, a listing of previously scheduled meetings, and a status tracker that monitors meeting progression. This layered interface enhances usability and provides clear navigation cues to the user.

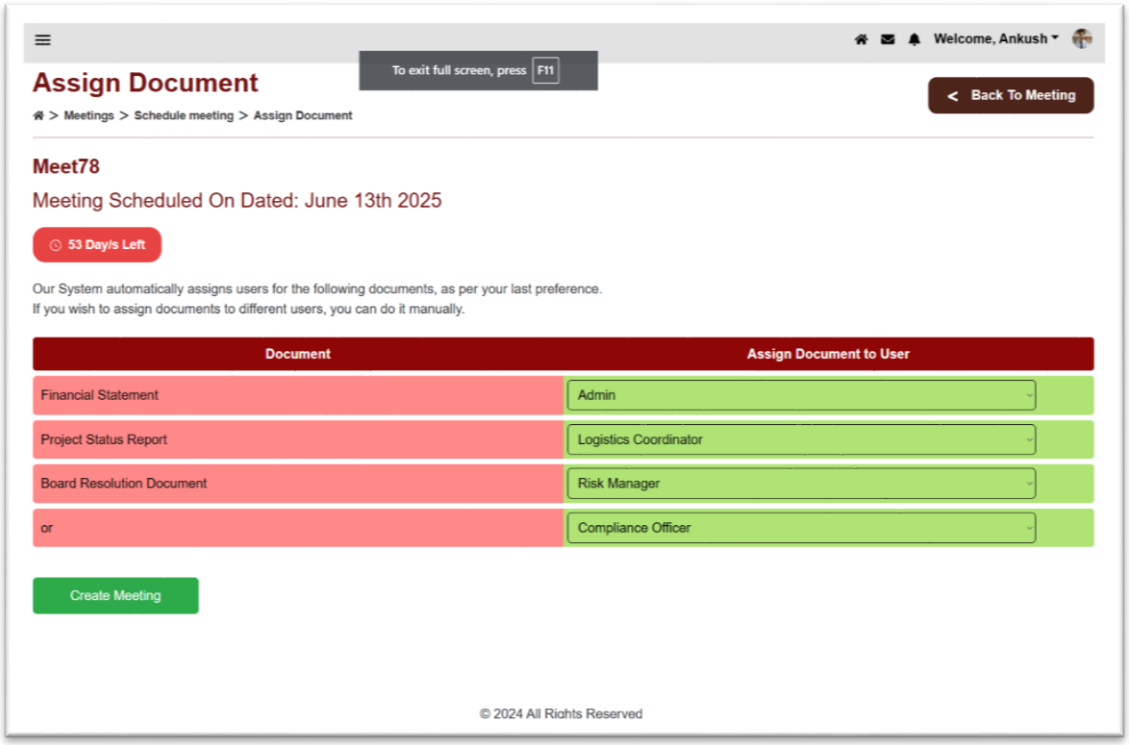
This functionality ensures that the platform can accommodate the dynamic needs of educational or organizational environments where regular meetings are scheduled and monitored by different administrative levels. It also offers scope for future enhancements such as scheduling recurring meetings or inviting external stakeholders.



Screenshot 5.1.1: Meeting List



Screenshot 5.1.2: Create Meeting

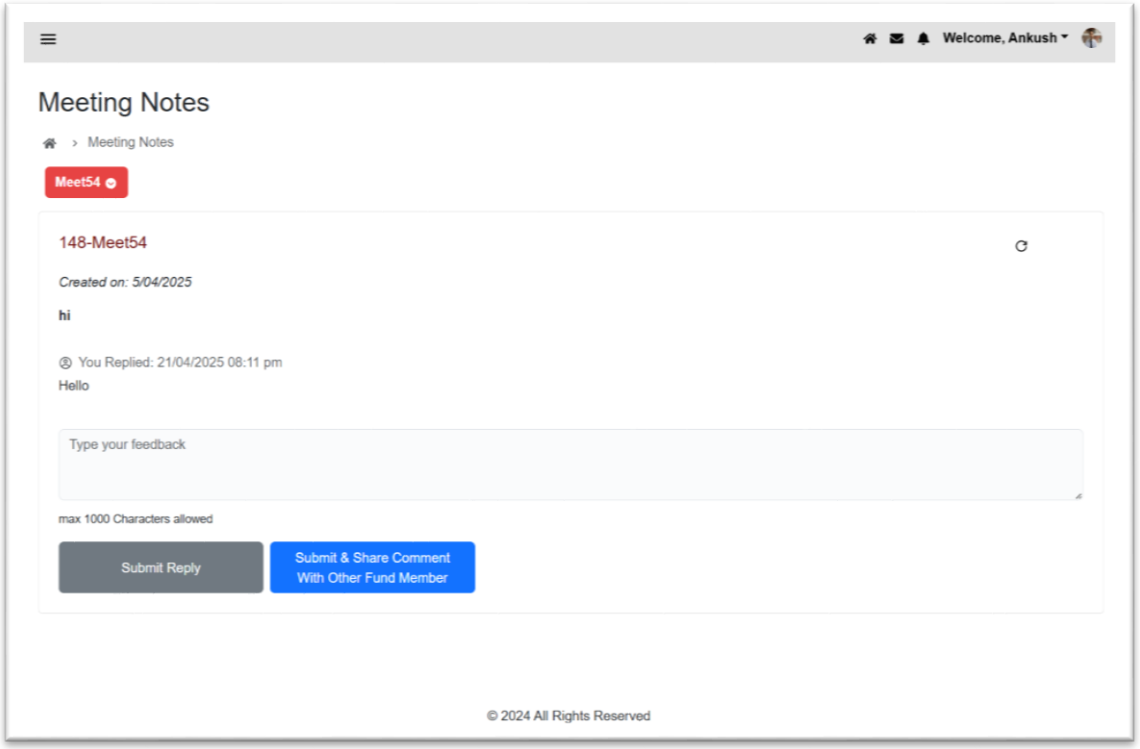


Screenshot 5.1.3: Scheduling Meeting

5.2 Implementation of Intra-Meeting Communication Through Chat

To support collaborative discourse during the meeting cycle, a dedicated communication channel was incorporated into the system. This is represented in the platform as a Chat or Notes module, which enables the meeting initiator (or owner) to begin a structured conversation with users belonging to the same fund group.

This module provides a chatbox-like interface that supports threaded conversations. It fosters asynchronous communication which is vital when participants are not available simultaneously. From a technical standpoint, the backend APIs were designed to associate chat messages with specific meetings and users, while ensuring that only authorized fund users could access their respective meeting chats.



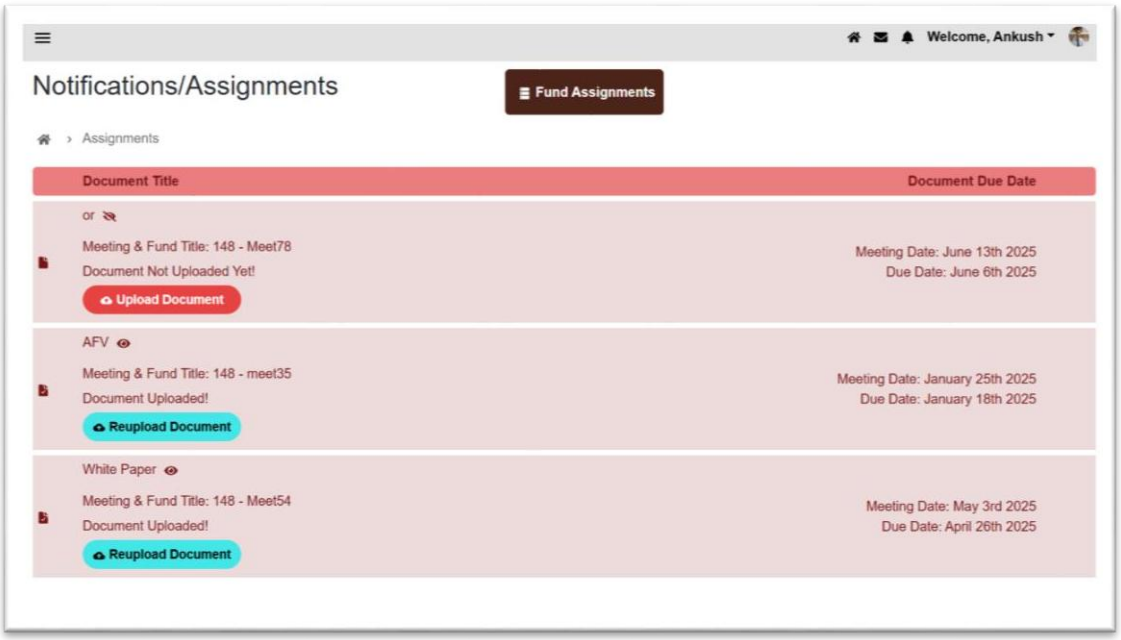
Screenshot 5.2.1: Meeting Notes

5.3 User Interaction via Assignment Submissions

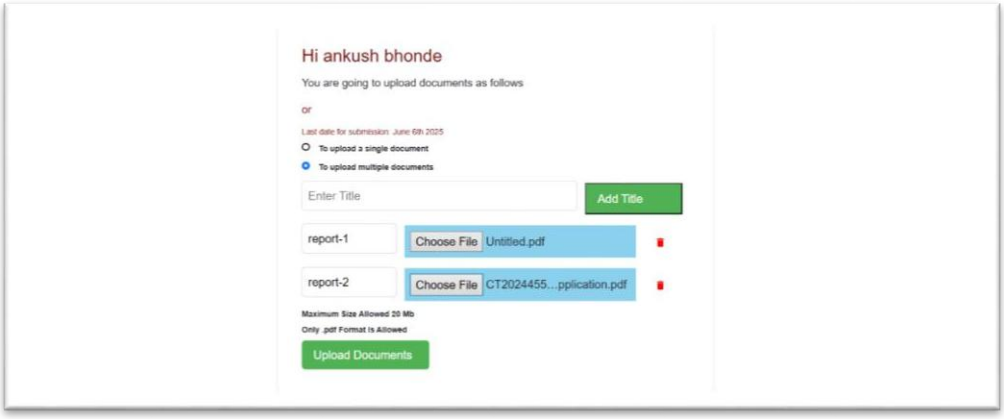
Another key result of the system's implementation is the assignment submission module, which enables participants of a meeting to upload relevant documents prior to the finalization of the agenda. The module is crafted with a user-centric design that includes validation checks for file types, upload status indicators, and an assignment list view for tracking submissions.

The interface progression is depicted through a cluster of three distinct user flows: uploading an assignment, listing all user submissions, and viewing individual document statuses. These components are synchronized with the meeting timeline to ensure timely documentation. The system architecture also supports dynamic file naming conventions and server-side storage using Django's file handling utilities.

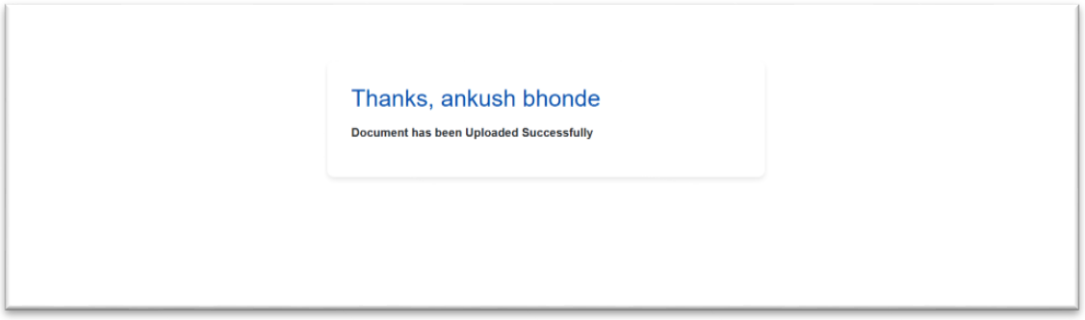
Functionally, this ensures that users remain accountable for their tasks while providing the meeting owner with a centralized repository of materials required for further agenda compilation. This module bridges the gap between user participation and content generation.



Screenshot 5.3.1: User Assignment List



Screenshot 5.3.2: User Assignment Submission

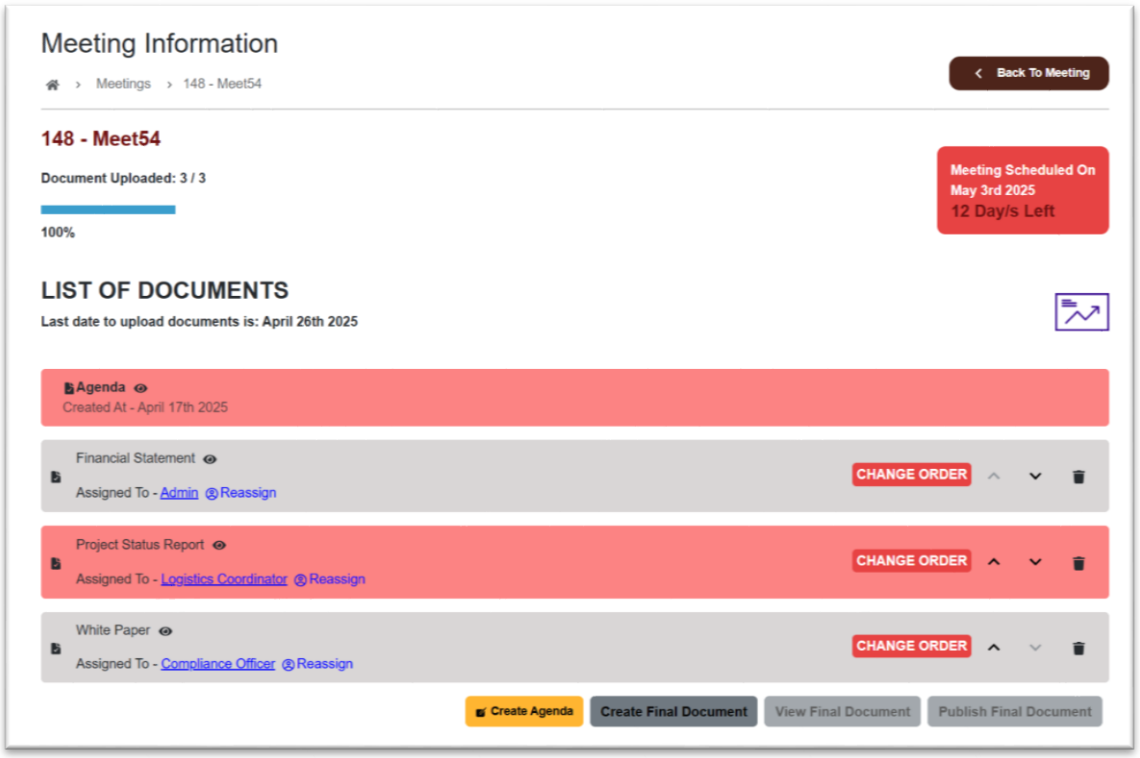


Screenshot 5.3.3: User Gratitude Page

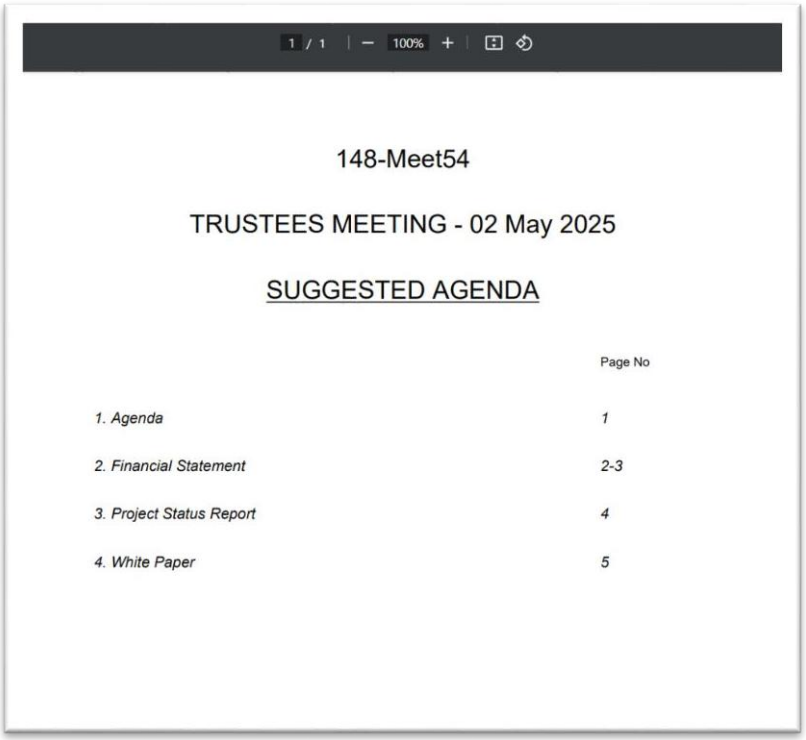
5.4 Agenda Generation and Final Documentation

Once all relevant documents have been submitted by the participants, the meeting owner transitions to the final phase — agenda creation and generation of the final meeting document. The agenda page in the application acts as an indexed overview of all submissions. It displays file titles alongside their corresponding page numbers, forming a table-of-contents-style layout that supports easy navigation and future referencing.

The process involves reviewing uploaded files in the Meeting Monitor, tagging or categorizing them as required, and finally initiating the automatic agenda generation using PDF manipulation utilities. This results in a consolidated document that includes all user submissions, indexed with proper pagination.



Screenshot 5.4.1: Meeting Information



Screenshot 5.4.1: Meeting Agenda

CHAPTER 6

CONCLUSION

CONCLUSION

6.1 CONCLUSION

The Meeting Portal project is a functional and effective solution for organizations that want to streamline the way they gather documents and work together. With role-based access and a collection of helpful modules, it simplifies managing meetings, delegating tasks, and communicating with team members. From scheduling meetings to sending notifications, everything is organized in a neat way, which keeps everyone informed and promotes a sense of responsibility. The project was constructed to address an everyday problem: keeping documents pertaining to meetings organized, easily accessible, and in control. With a simple web application, team members can easily review agendas, reserve meeting space online, and review minutes that are of interest to them so that they remain aware and active.

What makes it even more useful is the inbuilt calendar that reminds teams of deadlines and meeting times. The system automatically sends reminders in time so that nothing slips up, and everyone plans and collaborates more effectively. Those notifications aren't simply about preventing last-minute disasters they actually make the teams better communicators and keep them on schedule. Through its emphasis on accessibility and simplicity of organization, the Meeting Portal enhances productivity, fortifies collaboration, and facilitates an optimized working environment.

6.2 FUTURE SCOPE

Future development of the Meeting Portal project will include adding sophisticated security features for protecting sensitive documents and user data, consistent with industry requirements and legislation. In addition, deploying the platform to run on all devices, for example, mobile and tablet apps, will enhance user accessibility and engagement.

With real-time collaboration features, users will be able to co-author documents with ease, further streamlining the effectiveness and efficiency of the meeting preparation process. With these developments, Meeting Portal will be a complete solution for organizations who aim to streamline their document collection and collaboration processes.

REFERENCES

REFERENCES

1. C. Obagbuwa, O. J. Oyelade, O. O. Oladipupo, D. O. Aborisade, and I. T. Ewejobi, "Design and Implementation of Meetings Document Management and Retrieval System," *Journal of information Science*, vol. XX, no. X, pp. 1-14, 2012.
2. S. Saranga, D. Ranaa, S. Patela, and D. Savaliy, "Meetings through the cloud: Privacy-preserving scheduling on mobile devices,"*Journal of Information Security and Applications*, vol. XX, no. X, pp. 1-14, 2023
3. S. Thalawattha and D. Vidanagama, "A Survey on Web-based Meeting Scheduling Application," in *Conference Paper*, Jan. 2021.
4. C. H. Kao and S. T. Liu, "Development of a Document Management System for Private Cloud Environment," *Procedia – Social and Behavioral Science*, vol. 73, pp. 424-429, 2013.
5. Djedović, E. Žunić, D. Alić, S. Omanović, and A. Karabegović, "Optimization of the Business Processes Via Automatic Integration with the Document Management System," in *2016 IEEE International Conference on Business Informatics (CBI)*, 2016, pp. 117-122.
6. E. Badouel and M. Tchoupé Tchendji, "Merging Hierarchically-Structured Documents in Workflow Systems," *Electronic Notes in theoretical Computer Science*, vol. 203, pp. 3-24, 2008.
7. V. Bagire, J. Byarugaba, and J. Kyogabiirwe, "Organizational meetings: management and benefits," *Journal of Management Development*, vol. 34, no. 8, pp. 960-972, 2015.
8. Djedović, E. Žunić, D. Alić, S. Omanović, and A. Karabegović, "Optimization of the Business Processes Via Automatic Integration with the Document Management System," in 2016 *IEEE International Conference on Business Informativs (CBI)*, 2016, pp. 117-122.

9. S. Saranga, D. Ranaa, S. Patela, and D. Savaliy, "Document Management System Empowered by Effective Amalgam of Blockchain and IPFS," *Journal of Information Security and Applications*, vol. XX, no. X, pp. 1-14, 2023.
10. C.-N. Yang, P.-Y. Tsai, and Y. Liu, "A (k, n) secret document sharing with meaningful shares," Journal of Information Security and Applications, vol. 62, 2021, Art. no. 102973.
11. T. Athaya, S. Munira, A. Zaman, S. A. Hossain, and Col. A. B. M. Humayun Kabir, "A Proposed Algorithm and Architecture for Automated Meeting Scheduling and Document Management," in 2018 21st International Conference of Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 2018, pp. 1-6.
12. Zunino and M. Campo, "Chronos: A multi-agent system for distributed automatic meeting scheduling," Expert Systems with Applications, vol. 36, no. 7, pp. 7011-7018, 2009. doi: 10.1016/j.eswa.2008.08.024.

DISSEMINATION
OF WORK

DISSEMINATION OF WORK

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

Development of Next-Gen Meeting Portal

Dr. R. A. Zamare[1], Bhuvnesh Kale[2], Gaurav Kaple[3], Gaurav Dhale[4], Ankush Bhonde[5]
[1]Assistant professor, Dept. of CSE, Shri Sant Gajanan Maharaj College of Engineering, Shegaon-444203, India.
[2,3,4,5]UG Scholar, Dept. of CSE, Shri Sant Gajanan Maharaj College of Engineering, Shegaon-444203, India.
Emails: *razamare@ssgmce.ac.in[1], bhuvneshk2910@gmail.com[2], gkaple15@gmail.com[3], gauravdhale7404@gmail.com[4], ankushbhonde2805@gmail.com[5].*

Abstract
The Meeting Portal is a web-based platform designed to streamline the gathering, organization, and retrieval of documents. Unlike traditional meeting management systems that focus primarily on scheduling and discussions, Meeting Portal emphasizes document security, role-based access control, and automated agenda creation. It features a hierarchical structure where Master-admins can oversee multiple departments (Funds), each managed by an administrator responsible for assigning user roles. The platform enhances document management and ensures compliance through calendar-based scheduling, automated notifications, and submission tracking. Built with Django and React.JS, this scalable and user-friendly platform improves document traceability, accessibility, and collaboration. Key advancements include enhanced organizational collaboration, secure document handling, and a fund-based hierarchy.

Keywords: *Agenda; Document; Meeting; Fund; Role-Based Access.*

1. Introduction

With the fast-changing environment of organizational management, the importance of effective communication and efficient processes has never been higher. As companies are more dependent on digital solutions to enable collaboration, the older ways of handling meetings and document processes are on their way out. The use of technology in organizational operations has created new solutions that promote productivity, increase accountability, and improve collaboration among team members [1, 3]. Meeting Portal is one of these solutions and is a high-level platform that streamlines the management of meeting-related documents and offers role-based access to support security and efficiency. Development of NextGen Meeting Portal is constructed with a formalized framework consisting of three main roles: Masteradmin, Admin, and User. Each role has been created to serve specific purposes within the company, enabling distinct task division and access levels. Organizations are increasingly interested in carefully managing their private information both to ensure that it is properly handled, and also to make it more useful to their daily tasks [6]. This hierarchical model of access ensures the integrity of the document handling process, as it allows controlled access to sensitive data. The system consists of several modules such as Dashboard, Meetings, Calendar, Notifications, Notes, Assignments, Funds, User Types, Users, Materials, and Settings, and each module plays a different role which makes the system more efficient. We consider the manipulation of hierarchically structured documents within a complex workflow system [8]. The Dashboard is a central platform for Masteradmins and Admins to view, in one place, all activities in progress, such as a list of meetings and users. The feature adds to the level of supervision and management functionality, enabling administrators to monitor user activity and document submissions efficiently. The Meetings module is also important since it enables scheduling and creating meetings in which the admin can gather documents from different types of users. This module is intended to support meetings that are structured platforms for document

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

gathering and collaboration, not just virtual meetings. In today's world, in an organization every meeting is considered as one of the significant tasks since interactions among the employees is important. Therefore, the proper scheduling of these meetings helps in the completion of the tasks and activities of a group on time [5]. The Calendar module is a complement to the Meetings module in that it shows the agenda and meeting schedule, keeping all the participants aware of future events. This forward-looking scheduling minimizes the chances of missing deadlines or forgotten responsibilities. Notifications are important in keeping users aware of their tasks and deadlines, with reminders being automatically sent based on roles assigned. This aspect is crucial in sustaining accountability in the organization since it keeps users reminded of their duty and deadlines for submitting documents. The Notes module facilitates collaboration between Admins and Users in the form of real-time communication that enables alignment on document needs and meeting goals. The Assignments module allows users to submit documents within given timeframes, which ensures responsibility and timely upload. This organization in the submission of documents is critical to the maintenance of the information flow as well as to guaranteeing that all required materials are uploaded for review before meetings. The Materials module enables the creation and maintenance of documents needed for meetings. This is a critical feature since it makes it possible for the Masteradmin to specify the documents that will be required for every meeting, so that all attendees are well prepared, and the information required is easily accessible [9]. The workflow of the Meeting Portal starts with the Masteradmin defining Materials and User Types according to company requirements. Next, a Fund is created, associating related materials with respective user roles. This organized method not only increases the efficacy of document gathering but also informs everyone about their responsibilities and deadlines.

In addition, Meeting is an important part of daily life to discuss and share information. There are different kinds of methods, approaches, and techniques that have been used to hold meetings [7]. Meeting Portal facilitates optimized decision-making and business efficiency since it ensures the gathering and arranging of all critical documents in an orderly manner. Once all the assignments have been submitted by the users, Admin can prepare an agenda with details of the meeting, including meeting name and list of all the meeting documents together with their page numbers. Such an agenda works as a road map for the meeting, conducting discussions and confirming that all such topics are addressed. After the agenda has been prepared, the admin can produce a final document combining all documents that have been submitted with the agenda as the cover page. The final document is important to keep a record of the meeting and make the information discussed accessible to all attendees. The Meeting Portal not only makes it possible to collect documents but also organizes these documents and makes them available for reference purposes.

2. Literature Review

In order to address the challenges of efficiently organizing, storing, and retrieving meeting papers, the article offers the design and implementation of a Meetings Document Management and Retrieval System. It emphasizes the necessity of an object-oriented and indexing-based online application to enhance document security and retrieval. The system's goal is to improve organizational collaboration and decision-making by giving users immediate access to meeting agendas, minutes, and related materials. The authors emphasize how hypertext can be used to browse interconnected material, which eventually results in better meeting record management [1]. In response to the growing demand for scheduling systems that secure personal data, the paper provides a thorough explanation of privacy-preserving scheduling algorithms for mobile devices. It is an expansion of research on distributed constraint fulfilment, private set intersection, and secure multi-party computation, highlighting the drawbacks of traditional methods that either compromise user privacy or have scalability issues. In order to maximize privacy and efficiency for mobile scheduling scenarios, the authors provide three new methods based on homomorphic encryption that allow users to agree on mutual availability without disclosing their personal

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

schedules [2]. An overview of web-based meeting scheduler applications is provided in the paper, along with a discussion of common problems with regular meeting planning, such as poor communication and time waste. While concentrating on a single application to integrate crucial features like participant availability, agenda distribution, and reminders, it examines several algorithms and systems suggested for automating the scheduling process, such as conflict detection and optimization techniques. The study reveals a gap in existing solutions, underscoring the possibility of a new online application that effectively addresses these issues [3]. Effective communication and time management are two of the main problems with traditional meeting planning that are outlined in this paper's thorough literature review of web-based meeting scheduling solutions. It covers the variety of technologies and strategies, such as automated methods and conflict detection, that are suggested to optimize scheduling efficacy. To improve overall meeting efficiency, the review highlights the shortcomings of existing applications, which frequently lack certain functionalities. This highlights the need for a single solution that includes essential features like agenda sending, participant availability, and reminders [4]. The management and efficacy of meetings in Ugandan organizations are covered in the article "Organizational meetings: management and benefits" by Bagire et al. (2015), with an emphasis on the crucial role that policy plays in determining the outcomes of meetings. It critiques the existing literature, which is primarily from Western contexts, and emphasizes the significance of management strategies that are tailored to the local situation while taking cultural factors into account. The study demonstrates that while meetings are generally regarded as beneficial for communication and conflict resolution, issues such as subpar management techniques and a failure to take appropriate action compromise their efficacy, and further research in the African setting is required [7]. As with previous studies by Tu and Hsu and Liu et al., the work presents a novel (k, n) secret document sharing (SDS) method that enhances share management by employing meaningful shares as opposed to noise-like shares. This facilitates the transfer of additional information through these significant shares in addition to making it simpler for participants to recognize and manage shares. To overcome the shortcomings of earlier methods for secret document sharing, the system proposed in this study has better embedding efficiency while maintaining the same computing complexity as Liu et al.'s SDS. By offering a practical and effective structure for sharing confidential documents, this work advances the profession [10].

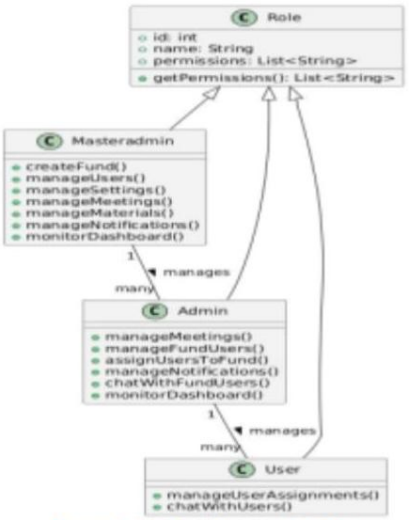


Figure 1 Role based access (RBAC)

In order to enhance efficiency and security, the study presents a comprehensive examination of existing document management and automatic meeting scheduling approaches in organizational contexts. It references several important studies that address scheduling optimizations and temporal conflicts, including the Distributed Multi-Event Scheduling framework and systems based on Open Constraint Programming. The evaluation also discusses information security in meeting management,

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

emphasizing secure document handling and encryption techniques to prevent data leaks. In general, the literature emphasizes how important it is to improve meeting management systems' automation and organization [11]. Figure 1 shows Role based access (RBAC).

3. Research Methodology

The research methodology for the study adopts a systematic procedure to design and develop an effective meeting document management system. The methodology is segmentalized into various phases ensuring systematic development and testing of the Meeting Portal system.

3.1 Research Design

This research applies the applied research philosophy, seeking the development of an organizational document management system that operates on the concept of role-based document control. The work is built along a qualitative as well as a quantitative approach that involves the inclusion of system development and user acceptance methods. It seeks to complement the use of conventional document management systems with a streamlined process of document submission, approval, and retrieval using organizational hierarchy.

3.2 Data Collection Methods

To establish a thorough perception of the problem in current document management systems, the following methods of data gathering were utilized:

- **Literature Review:** An in-depth review of currently available document management systems, i.e., Paperless Office, Content Management Systems, and Business Document Management, was carried out to determine their shortcomings in dealing with meeting documents. This led to the establishment of the research gap and outlining the objectives of the system.
- **System Requirement Analysis:** The information gathered was reviewed to establish the functional and non-functional requirements of the Meeting Portal system being proposed. Some of the important requirements were hierarchical management of users, tracking document submissions, and automatic generation of meeting agendas

3.3 System Development Approach

Agile Software Development Model was used for system implementation. This model supports iterative development, testing, and improvement based on user input. The major steps involved in the development process are:

- **Requirement Analysis:** Determining the major features like role-based access, fund-based hierarchy, agenda preparation, document submission, notifications, and final document generation.
- **System Architecture Design:** Developing a modular design for efficient handling of documents and managing user roles. The system comprises several modules such as Meetings, Assignments, Notes, Dashboard, and Calendar, to promote scalability and ease of maintenance.
- **Development:** It was developed on a Django backend and React.js frontend to guarantee scalability, real-time responsiveness, and secure handling of data. The backend is responsible for handling user authentication, document storage, and meeting workflows, while the frontend offers an interactive and user-friendly interface.
- **Testing & Validation:** The system was thoroughly tested, comprising unit testing, integration testing, and user acceptance testing (UAT). Testing included ensuring secure role-based access, document integrity, and effective execution of workflow.

3.4 Implementation and Evaluation:

The final system was implemented in a controlled setting to be tested. User input was gathered to assess usability, efficiency, and reliability. The assessment considered:

- Effectiveness of role-based access control in managing meeting documents.
- Ease of document submission and retrieval for various user roles.
- Automation effectiveness in agenda generation and document consolidation.
- Overall user satisfaction with the system's functionality. Figure 2 shows Meeting Lifecycle.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

Figure 2 Meeting Lifecycle

4. Result

Meeting Portal is a state-of-the-art, role-based meeting management platform that consolidates document gathering and meeting planning in a modular architecture. Figure 4 shows Meeting Agenda. With three unique roles—Masteradmin, Admin, and User—the portal isolates functions into administrative modules (Dashboard, Meetings, Calendar, Notifications, Settings, Materials, User type, and Funds) and collaborative modules (Notes and Assignments) to make sure that sensitive operations are controlled by authorized staff only. Masteradmin creates core elements like document templates (Materials) and user roles (User types), whereas Admins book meetings by choosing materials, assigning roles (secondary and backup users included) and monitoring document submission. This design not only improves efficiency and accountability in operations but also strengthens security by restricting access to key functions. Initial assessments show that the Meeting Portal successfully minimizes administrative overhead and presents scalability and integration potential with current enterprise software, laying a strong groundwork for future improvements like in-depth analytics and adaptive notification systems. Figure 3 shows Meeting monitoring screen.

Figure 3 Meeting Monitoring Screen

Figure 4 Meeting Agenda

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

Conclusion

The Meeting Portal project offers a solid solution to organizations wanting to automate their document collection and collaboration process. Providing role-based access and a suite of modules, it is effective in meeting management, assignment, and user-to-user interaction. The systematic organization of meetings, assignments, and notifications makes every participant active and responsible, thereby creating a better-organized and efficient workflow. The Meeting Portal project was created to meet the urgent need for proper organization, storage, and convenient access to meeting documents in organizations. Through the development of a web application, the system enables users to check meeting agendas conveniently and sign up for meetings online, making it easier to participate. Users can browse minutes of interest with ease, improving their ability to remain informed and active. Furthermore, the adoption of a calendar-based scheduling tool facilitates time management through the use of reminders on future deadlines and meeting schedules to inform all parties involved. By doing so, it avoids tardiness in filing documents and encourages better coordination across team members. Automated reminders and alerts also encourage time efficiency with reminders to stakeholders to avoid last-minute deadlines being missed. The focus on accessibility and organization not only maximizes workflow effectiveness but also enhances collaboration between workers, ultimately culminating in increased productivity and organizational performance.

Future Scope

The future direction of the Meeting Portal project involves the incorporation of advanced security features to safeguard sensitive documents and user information, in accordance with industry standards and regulations. Furthermore, extending the platform to be accessible on all devices, such as mobile and tablet applications, will improve user convenience and interaction. Having real-time collaboration capabilities will enable users to co-author documents effortlessly, further enhancing the effectiveness and efficiency of the meeting preparation process. These changes will make Meeting Portal a full-fledged solution for organizations that want to optimize their document collection and collaboration processes.

References

[1]. C. Obagbuwa, O. J. Oyelade, O. O. Oladipupo, D. O. Aborisade, and I. T. Ewejobi, "Design and Implementation of Meetings Document Management and Retrieval System," Journal of information Science, vol. XX, no. X, pp. 1-14, 2012.

[2]. S. Saranga, D. Ranaa, S. Patela, and D. Savaliy, "Meetings through the cloud: Privacy-preserving scheduling on mobile devices,"Journal of Information Security and Applications, vol. XX, no. X, pp. 1-14, 2023

[3]. S. Thalawattha and D. Vidanagama, "A Survey on Web-based Meeting Scheduling Application," in Conference Paper, Jan. 2021.

[4]. C. H. Kao and S. T. Liu, "Development of a Document Management System for Private Cloud Environment," Procedia – Social and Behavioral Science, vol. 73, pp. 424-429, 2013.

[5]. A. Djedović, E. Žunić, D. Alić, S. Omanović, and A. Karabegović, "Optimization of the Business Processes Via Automatic Integration with the Document Management System," in 2016 IEEE International Conference on Business Informatics (CBI), 2016, pp. 117-122.

[6]. E. Badouel and M. Tchoupé Tchendji, "Merging Hierarchically Structured Documents in Workflow Systems," Electronic Notes in theoretical Computer Science, vol. 203, pp. 3-24, 2008.

[7]. V. Bagire, J. Byarugaba, and J. Kyogabiirwe, "Organizational meetings: management and benefits," Journal of Management Development, vol. 34, no. 8, pp. 960-972, 2015.

[8]. A. Djedović, E. Žunić, D. Alić, S. Omanović, and A. Karabegović, "Optimization of the Business Processes Via Automatic Integration with the Document Management System," in 2016 IEEE International Conference on Business Informativs (CBI), 2016, pp. 117-122.

[9]. S. Saranga, D. Ranaa, S. Patela, and D.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 879-885
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0126

Savaliy, "Document Management System Empowered by Effective Amalgam of Blockchain and IPFS," Journal of Information Security and Applications, vol. XX, no. X, pp. 1-14, 2023.

[10]. C.-N. Yang, P.-Y. Tsai, and Y. Liu, "A (k, n) secret document sharing with meaningful shares," Journal of Information Security and Applications, vol. 62, 2021, Art. no. 102973.

CERTIFICATES OF PUBLICATION

e-ISSN No: 2584-2137

OPEN ACCESS

INTERNATIONAL RESEARCH JOURNAL ON ADVANCED ENGINEERING HUB (IRJAEH)

Email: editor@irjaeh.com

Available online at : https://irjaeh.com/index.php/journal

CERTIFICATE OF PUBLICATION

IRJAEH Is Hereby Awarding This Certificate To

Ankush Bhonde

In Recognition of The Publication of The Manuscript Entitled

Development of Next-Gen Meeting Portal

Published In Volume 03 Issue 03 March 2025.

Dr. M. Subramanian
Editor in Chief, IRJAEH,
Coimbatore,India

e-ISSN No: 2584-2137

OPEN ACCESS

INTERNATIONAL RESEARCH JOURNAL ON ADVANCED ENGINEERING HUB (IRJAEH)

Email: editor@irjaeh.com

Available online at : https://irjaeh.com/index.php/journal

CERTIFICATE OF PUBLICATION

IRJAEH Is Hereby Awarding This Certificate To

Bhuvnesh Kale

In Recognition of The Publication of The Manuscript Entitled

Development of Next-Gen Meeting Portal

Published In Volume 03 Issue 03 March 2025.

Dr. M. Subramanian

Editor in Chief, IRJAEH,

Coimbatore,India

e-ISSN No: 2584-2137

OPEN ACCESS

INTERNATIONAL RESEARCH JOURNAL ON ADVANCED ENGINEERING HUB (IRJAEH)

Email: editor@irjaeh.com

Available online at : https://irjaeh.com/index.php/journal

CERTIFICATE OF PUBLICATION

IRJAEH Is Hereby Awarding This Certificate To

Gaurav Dhale

In Recognition of The Publication of The Manuscript Entitled

Development of Next-Gen Meeting Portal

Published In Volume 03 Issue 03 March 2025.

Dr. M. Subramanian

Editor in Chief, IRJAEH,

Coimbatore,India

e-ISSN No: 2584-2137

OPEN ACCESS

INTERNATIONAL RESEARCH JOURNAL ON ADVANCED ENGINEERING HUB (IRJAEH)

Email: editor@irjaeh.com

Available online at : https://irjaeh.com/index.php/journal

CERTIFICATE OF PUBLICATION

IRJAEH Is Hereby Awarding This Certificate To

Gaurav Kaple

In Recognition of The Publication of The Manuscript Entitled

Development of Next-Gen Meeting Portal

Published In Volume 03 Issue 03 March 2025.

Dr. M. Subramanian
Editor in Chief, IRJAEH,
Coimbatore,India

e-ISSN No: 2584-2137

OPEN ACCESS

INTERNATIONAL RESEARCH JOURNAL ON ADVANCED ENGINEERING HUB (IRJAEH)

Email: editor@irjaeh.com

Available online at : https://irjaeh.com/index.php/journal

CERTIFICATE OF PUBLICATION

IRJAEH Is Hereby Awarding This Certificate To

Dr. R. A. Zamare

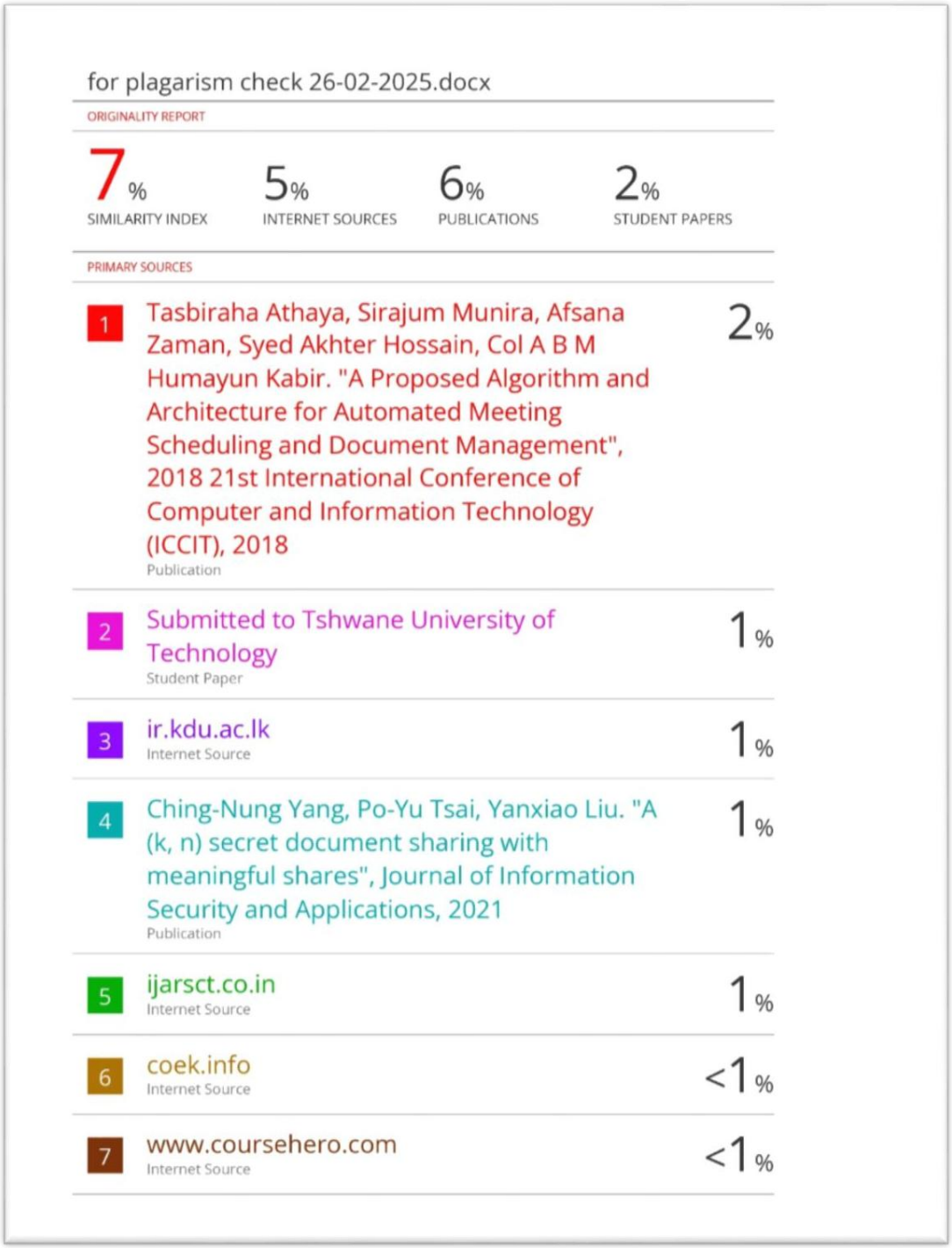
In Recognition of The Publication of The Manuscript Entitled

Development of Next-Gen Meeting Portal

Published In Volume 03 Issue 03 March 2025.

Dr. M. Subramanian
Editor in Chief, IRJAEH,
Coimbatore,India

RESEARCH PAPER PLAGARISM REPORT



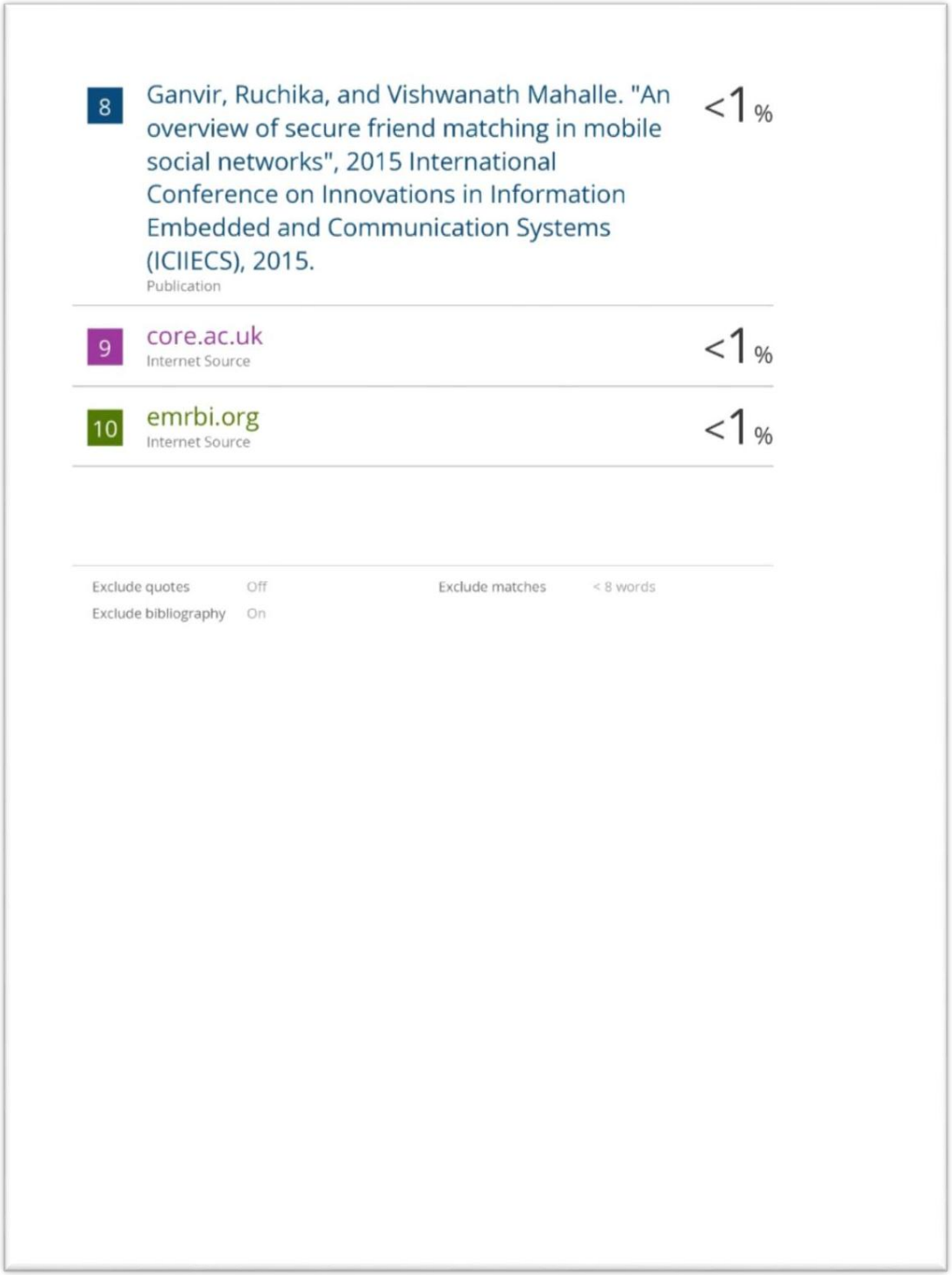
for plagarism check 26-02-2025.docx

ORIGINALITY REPORT

7%	5%	6%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Tasbiraha Athaya, Sirajum Munira, Afsana Zaman, Syed Akhter Hossain, Col A B M Humayun Kabir. "A Proposed Algorithm and Architecture for Automated Meeting Scheduling and Document Management", 2018 21st International Conference of Computer and Information Technology (ICCIT), 2018 Publication	2%
2	Submitted to Tshwane University of Technology Student Paper	1%
3	ir.kdu.ac.lk Internet Source	1%
4	Ching-Nung Yang, Po-Yu Tsai, Yanxiao Liu. "A (k, n) secret document sharing with meaningful shares", Journal of Information Security and Applications, 2021 Publication	1%
5	ijarsct.co.in Internet Source	1%
6	coek.info Internet Source	<1%
7	www.coursehero.com Internet Source	<1%



8	Ganvir, Ruchika, and Vishwanath Mahalle. "An overview of secure friend matching in mobile social networks", 2015 International Conference on Innovations in Information Embedded and Communication Systems (ICIIECS), 2015. Publication	<1 %
9	core.ac.uk Internet Source	<1 %
10	emrbi.org Internet Source	<1 %

Exclude quotes	Off	Exclude matches	< 8 words
Exclude bibliography	On		

PROJECT GROUP MEMBERS

Name: Ankush Deepak Bhonde

Address: Deshmukh Fail Akola, 444001

Email: ankushbhonde2805@gmail.com

Mobile no: +91 7249700781

Name: Bhuvnesh Hariom Kale

Address: Keshav Nagar, Khamgaon, 444303

Email: bhuvneshk2910@gmail.com

Mobile no: +91 7499553500

Name: Gaurav Rajendra Dhale

Address: Sai nagar, Amravati, 444607

Email: gauravdhale09@gmail.com

Mobile no: +91 7507216007

Name: Gaurav Vijay Kaple

Address: Choubhara Chowk Jalgaon Jamod, 443402

Email: gkaple15@gmail.com

Mobile no: +91 8767329349