A

Project Report

on

Project Pulse: College Project Showcase Platform

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

Parth Kinkar
(PRN: 213120189)
Abhishek Dhoran
(PRN: 213120371)
Ajinkya Nagare
(PRN: 213120216)
Shivani Digole
(PRN: 213120274)

Under the Guidance of

K. P. Sable

Asst. 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. Parth Kinkar, Mr. Abhishek Dhoran, Mr. Ajinkya Nagare and Miss. Shivani Digole** 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 "**Project Pulse: College Project Showcase Platform**" 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.

KPSable
9/5/2025

Asst. Prof. K. P. Sable
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. Parth Kinkar, Mr. Abhishek Dhoran, Mr. Ajinkya Nagare and Miss. Shivani Digole** 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 "**Project Pulse: College Project Showcase Platform**" 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

KP Sable

Name and Signature

Date: 9/5/2025

External Examiner

Dr. A. A. Bardekar

Name and Signature

Date: 9/5/2025

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 **Asst**. **Prof. K. P. Sable** for his 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.

Projectors
Parth Kinkar (22BD310769)
Abhishek Dhoran (22BD310730)
Ajinkya Nagare (22BD310731)
Shivani Digole (22BD310794)

Contents

Abstract

In today‘s rapidly evolving educational landscape, there is a growing disconnect between academic training and industry expectations, leaving students unprepared for real-world challenges. Furthermore, limited access to faculty guidance for projects hampers students' ability to maximize their potential. Our project, Project Pulse, addresses this gap by providing a centralized platform where students can showcase their completed and ongoing projects, connect with faculty for mentorship, and engage with potential employers based on their skills and project achievements. By fostering transparency, mentorship, and industry relevance, ProjectPulse empowers students to take ownership of their learning journey, supports faculty in nurturing innovation, and helps recruiters discover talent in a more meaningful way. This platform not only enhances student visibility but also redefines how academic projects are perceived in the broader professional landscape.

Keywords: Project showcase, Industry alignment, Academic mentorship, Career opportunities, Project Pulse

List of Abbreviations

Abbreviation	Description
RBAC	Role Based Access Control
ORM	Object Relation Mapping
CDN	Contact Delivery Network
CSV	Comma Separated Values
DOM	Document Object Model
JWT	Jason Web Token
UI	User Interface
MTV	Model Template View
MVC	Model View Controller
API	Application Program Interface
CSRF	Cross-Site Request Forgery

List of Figures

List of Screenshot

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1 PREFACE

Projects are an essential part of a student's learning journey, helping them develop practical skills and gain hands-on experience. However, many students struggle to connect what they learn in the classroom with real-world applications. The lack of exposure to industry expectations and limited access to faculty mentorship often make it challenging for students to fully realize their potential. Despite the importance of showcasing projects to enhance career prospects, students often find it difficult to present their work effectively to academic mentors, industry professionals, and potential employers. One of the biggest hurdles students face is the absence of a centralized platform where they can document, track, and share their projects in a structured way. Without proper visibility, many brilliant ideas and efforts go unnoticed, limiting opportunities for collaboration, mentorship, and career growth. Additionally, faculty members may struggle to connect with students' work, making it harder to provide valuable guidance and industry insights. To tackle these challenges, we created ProjectPulse, a digital platform designed to empower students by giving them a space to showcase their completed and ongoing projects. This platform fosters collaboration by connecting students with faculty and potential employers who can offer mentorship and career advice based on their project achievements. ProjectPulse serves as a bridge between academia and industry, ensuring students feel more prepared to take on real-world challenges.

With ProjectPulse, students can build strong professional profiles by showcasing their skills and expertise through their project work. The platform creates networking opportunities and encourages a culture of knowledge sharing and growth, ultimately improving employability and academic engagement. By integrating features such as mentorship support and industry interaction, ProjectPulse is set to cultivate a dynamic community where talent and innovation thrive. This paper will explore the design, functionality, and potential impact of ProjectPulse, emphasizing how digital solutions can enhance project visibility, strengthen connections between academia and industry, and empower students to take control of their professional growth.

1.2 MOTIVATION

The evolution of education in the digital age has unveiled several persistent challenges that call for innovative interventions. While academic institutions have embraced digital tools to facilitate learning and skill development, a critical gap remains in how students document, showcase, and receive recognition for their project-based work. In the absence of structured visibility and centralized repositories, many high-potential student projects remain underexposed, depriving learners of opportunities for mentorship, collaboration, and career advancement.

Moreover, the disconnect between academic curricula and industry requirements has become increasingly apparent. Students often struggle to translate theoretical concepts into real-world applications, leading to a mismatch between graduates' skillsets and employers' expectations. This gap is further widened by limited access to faculty guidance, sporadic industry interactions, and a lack of digital infrastructure that promotes continuous learning and professional growth.

These challenges are compounded by the absence of platforms where faculty, and recruiters can efficiently explore student projects, assess capabilities, or provide timely feedback. As a result, both students and educators lose out on valuable opportunities to engage in meaningful dialogue, share expertise, and drive innovation within the academic environment.

The motivation for developing ProjectPulse arises from the need to empower students with a centralized digital platform that not only allows them to document and showcase their projects but also connects them with a broader academic and professional ecosystem. By offering features such as mentorship integration, and recruiter engagement, ProjectPulse aims to bridge the gap between academia and industry, while promoting transparency and visibility in student accomplishments.

Inspired by the principles of collaborative learning and real-time interaction, ProjectPulse provides a structured space where innovation is recognized, feedback is streamlined, and students are encouraged to take ownership of their professional journey. Just as blockchain ensures trust and security in digital healthcare, ProjectPulse introduces trust, transparency, and connectivity into the student project ecosystem, creating a culture of continuous improvement and real-world readiness.

1.3 PROBLEM STATEMENT

Current education systems lack alignment with industry demands, leaving students unprepared. Access to faculty guidance for projects is limited. A centralized platform is needed to display student projects, faculty facilitator and connect students with career opportunities.

1.4 OBJECTIVES

1) Bulk Registration of Students and Faculty with Secure Access Management
2) Project Uploading, Group Collaboration, and Faculty Endorsements
3) Skill-Based Student Discovery for Recruiters and Faculty-Student Interaction
4) Performance Tracking and Portfolio Development for Students

1.5 SCOPE OF PROJECT

1) Developing a centralized web-based platform to enable students to showcase their academic, personal, and collaborative projects for academic and professional visibility.
2) Implementing a modular and scalable backend architecture with secure database integration to manage user registrations, project uploads, and platform interactions.
3) Enabling role-based access control to deliver personalized experiences and ensure secure data segregation for students, faculty, and recruiters.
4) Building a responsive and intuitive frontend interface using React.js and Next.js to ensure seamless navigation, real-time updates, and multi-device compatibility.
5) Allowing students to upload detailed project information including technologies used, outcomes, team members, and supporting documents, with tagging and categorization features for improved discoverability.
6) Incorporating advanced search and filtering mechanisms to help users find relevant projects based on technology stack, skill domains, or academic year.
7) Facilitating engagement through a mentorship module where faculty can provide feedback, guidance, and project endorsements.
8) Ensuring platform security through encrypted data storage, secure login mechanisms, and strict role-based authentication to prevent unauthorized access.
9) Supporting recruiters in discovering potential candidates by browsing student profiles and projects aligned with industry needs.

10) Promoting academic collaboration, professional networking, and real-world visibility for students through an integrated, interactive, and secure digital ecosystem.

1.6 ORGANIZATION OF PROJECT

Chapter 1: This chapter introduces the core idea behind Project Pulse – a centralized college project showcase platform. It describes the significance of such a system in academic and industrial domains, emphasizing how it benefits students by offering exposure and recruiters by providing an innovative skill-based hiring approach. The chapter also highlights the scope, objectives, and key challenges that inspired the development of the platform.

Chapter 2: The literature survey presents a detailed study of existing platforms and tools used for project presentation and skill showcasing, such as GitHub, LinkedIn, and Behance. It reviews academic papers and case studies to understand trends in academic project documentation, visibility, and how they contribute to student growth. The chapter also explores the gaps in existing systems and how Project Pulse aims to address them.

Chapter 3: This chapter discusses the project methodology adopted during the development of Project Pulse. It elaborates on the agile approach used for planning and execution, details the system architecture, and outlines the modules such as student dashboard, recruiter portal, and analytics. The chapter also discusses the technology stack chosen, including Firebase, ReactJS, and Tailwind CSS, and the rationale behind their selection.

Chapter 4: The implementation chapter explores in detail the technical realization of Project Pulse. It explains the frontend and backend integration, design strategies for UI/UX, authentication and project upload systems, recruiter filtering algorithms, and analytics dashboard. This chapter breaks down the actual development process, highlighting design iterations, testing strategies, and the tools used throughout.

Chapter 5: The results and discussion chapter evaluates the outcomes of deploying the Project Pulse platform. It covers metrics such as user engagement, number of projects uploaded, recruiter feedback, and student participation. It also includes a discussion on challenges faced, how they were mitigated, and what insights were gained from feedback sessions with users. Suggestions for feature enhancements and usability improvements are also discussed.

Chapter 6: The final chapter concludes the project by summarizing key accomplishments, highlighting its impact on student visibility and recruitment processes. It also reflects on the learning experience and outlines possible future enhancements like AI-powered recommendations, inter-college integration, and support for multimedia-rich project presentations. This chapter provides both closure and direction for continued development of the platform.

CHAPTER 2

LITERATURE REVIEW

LITERATURE REVIEW

2.1 INTRODUCTION TO PROPOSED SYSTEM

Project Pulse is a college project showcase platform designed to enable students to display their academic projects, connect with potential recruiters, and receive feedback and ratings[1]. The platform offers an interactive environment where students can upload their projects, categorize them, and showcase their work in a structured format. Recruiters can browse through student projects based on specific filters such as technology stack, project type, and ratings[2]. This will facilitate seamless connections between students and industry professionals, making it easier for recruiters to find talent based on the projects students have worked on. The system is designed with two primary goals:

Showcasing Student Projects: Allowing students to showcase their creativity, innovation, and technical expertise[3].

Connecting Recruiters with Talented Individuals: Enabling recruiters to find candidates based on their project work[4], simplifying the recruitment process. Additionally, the platform will offer analytics to both recruiters and students, providing insights into the engagement and feedback on each project. Students can use this feedback for improvement, while recruiters can assess candidates based on actual work[5].

2.2 STAKEHOLDERS AND ROLES

The key stakeholders involved in Project Pulse and their respective roles are:

- Students: The primary users of the platform, responsible for uploading and managing their projects. They engage with recruiters and use feedback to enhance their skills[6].
- Recruiters: Hiring professionals or organizations looking for talent based on students' project work. They evaluate candidates based on project ratings and other details provided on the platform[7].

- Platform Administrators: Responsible for maintaining and managing the platform. They ensure smooth operations, moderate content, and handle any issues related to project submissions, ratings, or user engagement[8].
- Mentors/Faculty: These individuals can also serve as guides or evaluators for projects. They may provide feedback or endorse student projects[9], helping students improve and giving recruiters a clearer understanding of the project's academic background.

2.3 PROJECT EVALUATION AND RATING SYSTEM

The Project Evaluation and Rating System is a core component of the Project Pulse platform, playing a crucial role in how projects are assessed, recognized, and used for recruitment purposes. Unlike traditional systems where students are graded solely by faculty members within an academic setting, Project Pulse introduces a more dynamic and transparent evaluation process[10]. Once a project is uploaded to the platform, it becomes available for review by multiple parties including peers, mentors, and recruiters. This multi-dimensional evaluation approach ensures that feedback is not limited to academic performance alone but also reflects real-world relevance, creativity, and technical proficiency.

The rating system is designed to be intuitive and fair. Users can provide ratings based on parameters such as innovation, technical depth, presentation quality, user interface design, and practical usability[11]. These parameters help break down the strengths and weaknesses of a project, offering a more detailed understanding than a single grade or score. The ratings and feedback provided by recruiters carry significant weight, as they are often based on industry standards and expectations. This makes the system not only motivational for students but also highly valuable for companies seeking skilled individuals.

Moreover, the platform uses these ratings to generate a project leaderboard or ranking system, which helps highlight top-performing projects. This visibility can be highly beneficial for students, as highly-rated projects are more likely to catch the attention of recruiters and mentors. Over time, this evaluation system encourages a culture of continuous improvement, where students refine their skills based on constructive criticism and strive to meet higher industry benchmarks.

2.4 LARGE DATA HANDLING

In Project Pulse, large data handling is essential due to the extensive amount of content and user data managed by the platform. Students upload various project-related files including documentation, images, videos, and source code, all of which contribute to growing data volumes. The backend of Project Pulse, built using Django, plays a crucial role in efficiently storing, organizing, and serving this data. Django's robust ORM (Object Relational Mapping) system helps manage complex database operations with ease, while cloud-based storage solutions and content delivery networks (CDNs) ensure quick access to heavy media files. To maintain speed and performance, the system also uses techniques such as file compression, lazy loading, and pagination for browsing large sets of project data[12].

In addition to individual uploads, bulk student data is collected directly from colleges through CSV files, which are imported and processed by the Django backend. This allows administrators to quickly onboard large numbers of students and their project details into the system. Once uploaded, Django handles the data parsing, validation, and database entry processes, making the workflow both efficient and scalable. The backend also tracks real-time interactions such as project views, ratings, and comments, enabling analytics and personalized recommendations. With this structured approach to data handling, Project Pulse ensures reliable performance and smooth scalability, even as the user base and data size continue to grow.

2.5 COST CONCERNS AND ALTERNATIVE SOLUTIONS

One of the major cost concerns in Project Pulse is managing and storing large volumes of data, especially media files like videos and images, which can significantly increase server and cloud storage expenses over time[13]. Additionally, handling real-time analytics and providing fast, reliable access to users may require more advanced and scalable hosting solutions, further adding to the operational costs. To address these challenges, alternative solutions can include using open-source technologies and frameworks like Django to minimize software licensing fees, integrating cost-effective cloud storage options such as AWS S3 with tiered storage plans, and employing third-party services like GitHub or YouTube for hosting code and demo videos externally. Implementing data compression, caching, and archiving older or less-accessed content can also help reduce storage and bandwidth usage, ultimately lowering costs while maintaining performance.

2.6 INTEGRATION OF INSIGHTS

The integration of insights in Project Pulse plays a key role in enhancing user experience and decision-making for both students and recruiters. By collecting and analyzing interaction data—such as project views, ratings, comments, and recruiter engagement—the platform generates meaningful insights that are displayed through interactive dashboards[14]. These insights help students understand how their projects are performing, which skills are in demand, and where they can improve. For recruiters, the insights highlight trending technologies, top-rated projects, and student profiles that align with their hiring needs. Using Django’s backend capabilities along with data visualization libraries or tools like Chart.js or D3.js, these analytics can be seamlessly integrated into the user interface, making the platform more intelligent, user-centric, and impactful.

CHAPTER 3

METHODOLOGY

METHODOLOGY

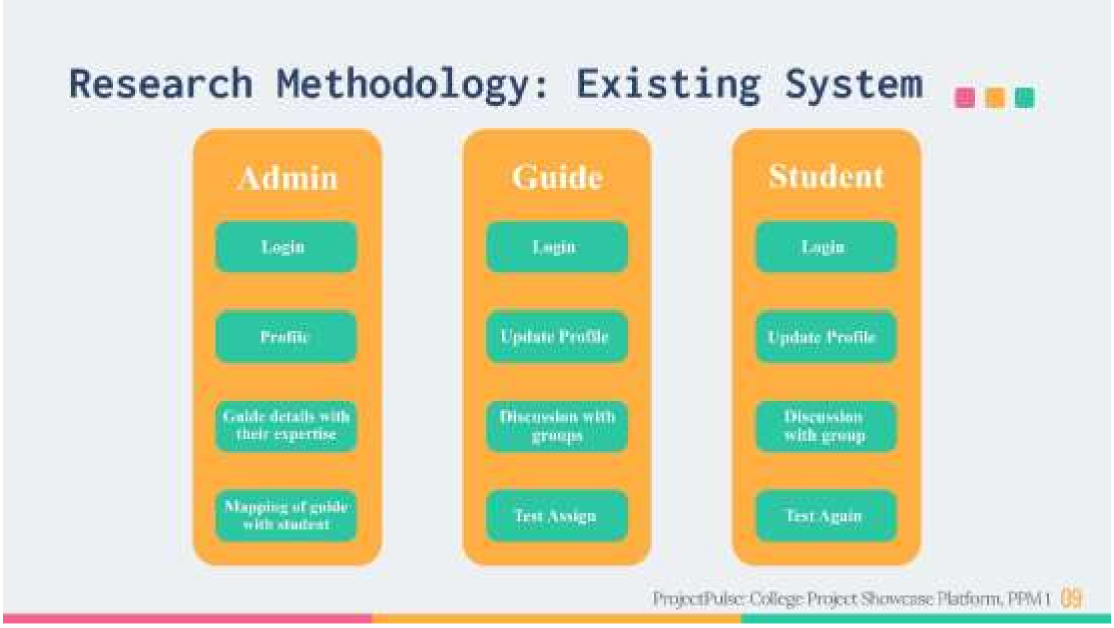
3.1 EXISTING SYSTEM

1. Admin Module: The Admin plays a crucial role in managing user accounts and assigning faculty guides to students. The admin logs into the system to manage student and faculty data. Profile management allows the admin to update system-level access controls. The admin registers faculty members and records their area of expertise. Based on the guide‘s expertise, the admin manually assigns students to specific faculty members.

2. Guide (Faculty) Module: Faculty members are responsible for mentoring students, monitoring project progre0ss, and evaluating their work. Login: Guides access their accounts to manage student projects. Profile Update: Faculty members add or update their expertise and specialization. Discussion with Groups: Faculty interacts with student teams in scheduled sessions. Test Assignment: Guides assign tests and evaluate student responses.

3. Student Module: Students use the platform primarily to receive guidance, discuss projects, and complete assignments. Login: Students access their assigned projects and faculty mentors. Profile Update: Students update academic details and project interests. Discussion with Groups: Communication is limited to scheduled faculty-student interactions. Test Reattempt: Students must retake tests if they fail the first attempt.

The existing system lacks an automated project showcase, restricting students from independently displaying their work to the public or recruiters, limiting recognition and career opportunities. Manual faculty-student mapping by admins is inefficient and not optimized based on expertise. Limited International Journal of Scientific Research in Science and Technology (www.ijsrst.com) 653 interaction and collaboration confine faculty-student communication to scheduled discussions, delaying feedback and project progress. Additionally, recruiters lack direct access to student projects, making hiring inefficient. These challenges highlight the need for an automated system to enhance project visibility, streamline mentorship, and improve recruiter engagement.



Research Methodology: Existing System

Admin

Login

Profile

Guide details with their expertise

Mapping of guide with student

Guide

Login

Update Profile

Discussion with groups

Test Assign

Student

Login

Update Profile

Discussion with group

Test Again

ProjectPulse: College Project Showcase Platform, PPM 1 09

Figure 3.1 Existing System

3.2 PROPOSED SYSTEM

1. User Registration and Login: The system starts when a user logs in based on their role: Students log in to create and showcase projects. Colleges (Faculties/Admins) register, oversee student progress, and provide feedback. Recruiters log in to explore project-based candidate profiles. Admin Responsibilities: The admin logs into the system securely. Registers new users (students, faculty, and recruiters). Stores user details in the database and generates unique user IDs. Ensures smooth platform functionality and data security.

2. Student Module: Students use the platform for project collaboration and showcasing their work. Steps: Student Login: A student logs in and accesses their profile. Create Project: The student initiates a new project, adding a description, technologies used, and objectives. Collaborate & Form Teams: Students can invite peers and faculty members to collaborate. Chat with Faculty & Teammates: Integrated communication tools allow real-time discussions, doubt clearance, and knowledge exchange. Complete & Post Project: Once finalized, the project is submitted for review and feedback.

3. College Module (Faculty & Admin): The faculty and college administrators play a critical role in guiding students and ensuring high-quality project development. Steps: Sign-Up & Register Faculty & Students: The institution registers faculty members and students onto the platform. Solve Doubts: Faculty members provide insights and

assistance to students in resolving technical or conceptual doubts. Rate Projects & Give Feedback: Faculty evaluates the projects based on innovation, execution, and presentation. Feedback is recorded to help students refine their projects. Feedback Integration: The feedback mechanism helps students iterate on their projects for better quality and innovation.

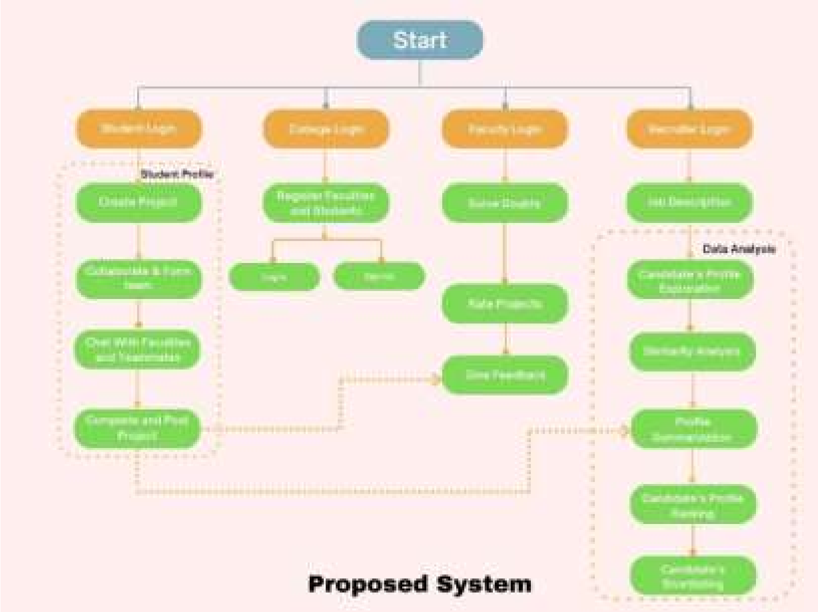


Figure 3.2 Proposed System

4. Recruiter Module: Recruiters explore the platform to find suitable candidates based on their project work. Steps: Recruiter Login & Job Description Posting: Recruiters log in and define the skills they are looking for in candidates. Data Analysis & Profile Exploration: The system performs data analysis to match recruiter requirements with student profiles. It explores project submissions, skills, and innovation levels. Similarity Analysis: The system compares candidates based on project relevance, technology stack, and problem-solving approaches. Profile Summarization & Ranking: The system generates a summarized view of each candidate‘s work Profiles are ranked based on recruiter requirements. Candidate Shortlisting: The recruiter finalizes potential candidates for interviews or further discussions.

3.3 MODULES

1. User Management

This module forms the foundation of the platform’s security and role-based access control.

It includes the following submodules:

Authentication: Registration and login using JWT tokens to ensure secure sessions. Passwords are stored securely using hashing.

Student Management: Admins can create students either individually (through a form) or via Excel bulk upload. Students can edit their profile, add skills, upload resumes, and update contact details.

Department Management: Admins can add, edit, and delete departments. Each student and teacher is linked to a department, ensuring data is organized branch-wise.

Teacher Management: Admins can manage teacher accounts (create/edit/delete) and assign them to specific departments. Teachers have restricted access compared to Admins and may be used in future expansions like project evaluation or mentorship.

Goal: Ensure all users have correct access according to their role while keeping the system flexible for management.

2. Project Showcase

The project showcase module allows students to display their work to peers, teachers, and recruiters.

Students can create projects by filling out details like project title, description, technologies used, GitHub link, deployment link, and collaborators.

Multiple students can collaborate on a single project entry, with a user-friendly selection for adding team members.

Uploaded projects can be viewed publicly (depending on future access settings) to help recruiters and other students explore work easily.

Students can edit or delete their projects if needed.

Goal: Provide students a professional portfolio-like space inside the platform itself, making their achievements visible.

3. Search and Filtering

This module enables smart discovery of students and projects:

Students can be searched and filtered based on name, department, year of study, skills, and interested technologies.

Projects can be searched by project title, technologies used, and department.

Search results are dynamically loaded to enhance user experience without needing page reloads.

Goal: Allow recruiters, teachers, and other students to quickly find relevant profiles and projects, saving time and improving usability.

4. Admin Dashboard

The Admin Dashboard acts as the command center for College Admins:

Displays key statistics such as total number of students, teachers, projects, and departments.

Provides quick-access links to management pages for creating or editing students, teachers.

Optionally extendable to include analytics like most popular technologies among students, project upload activity, etc.

Built for ease of navigation so that Admins can monitor the platform without needing technical knowledge.

Goal: Make the administrative side of the platform as smooth, fast, and intuitive as possible.

5. Recruiter Access

The recruiter module is specifically built to enable industry professionals to discover talent easily:

Recruiters can register and login as a special user type with access to public student and project data.

Browse student profiles by skills, technologies, and year.

View showcased projects to assess the capabilities of students based on real-world work.

Optionally, recruiters could also bookmark students/projects they are interested in (expandable feature).

Goal: Make it easy for recruiters to find and assess students based on skills and practical work rather than just academic scores.

3.4 ENTITIES IN THE SYSTEM

1. Student: The Student is a primary user of the system who can register, log in, and upload project details. Each student profile includes information such as name, email, enrolment number, branch, year, and resume. Students can update their profiles, add new projects, and view ratings or feedback on their work.

2. Project: The Project entity represents the core content of the platform. Each project includes a title, description, technologies used, GitHub link, documentation, and media (images or demo videos). Projects are associated with the student who created them and are visible to recruiters and other users.

3. Recruiter: The Recruiter is an external user who can view student projects for hiring purposes. Recruiters can create accounts, search for projects based on technologies or domains, view detailed project analytics, and contact students.

4. Admin: The Admin manages the overall platform. This entity is responsible for verifying student accounts, approving projects, managing reported content, and

maintaining system security. The admin can also access analytics dashboards to

5. Ratings & Reviews: This entity stores feedback given by other users or recruiters on each project. It includes numerical ratings, text-based reviews, and timestamps. It helps to rank projects and provides credibility to the students' work.

6. Technologies: The Technologies entity maintains a list of all technologies used in projects (e.g., Python, React, Django, Firebase). Each project is linked to one or more technologies, which can be used as filters for search and recommendation.

7. Analytics: This entity stores data about user interactions with the platform, including project views, downloads, ratings, and recruiter interest. It supports the analytics feature that helps both students and admins gain insights into performance.

8. Messages / Notifications: This entity handles internal communication, such as notifications to students about project ratings, admin approvals, recruiter messages, and platform updates. It ensures timely and relevant communication across user roles.
9. Shortlisted Projects: This is a relationship entity used by recruiters to bookmark or shortlist projects they are interested in. It links recruiters to projects and helps them manage their hiring pipeline within the platform.

10. User Sessions / Authentication: This entity deals with login/logout activity, session management, and password security for all types of users (students, admins, recruiters). It ensures authorized access and user tracking.

CHAPTER 4

IMPLEMENTATION

IMPLEMENTATION

The Student Showcase Platform is implemented as a full-stack web application using React for the frontend and Django (with Django REST Framework) for the backend. The system includes JWT-based authentication, role-based access control, and structured APIs for handling student profiles, project uploads, and recruiter interactions.

4.1 SYSTEM ARCHITECTURE:

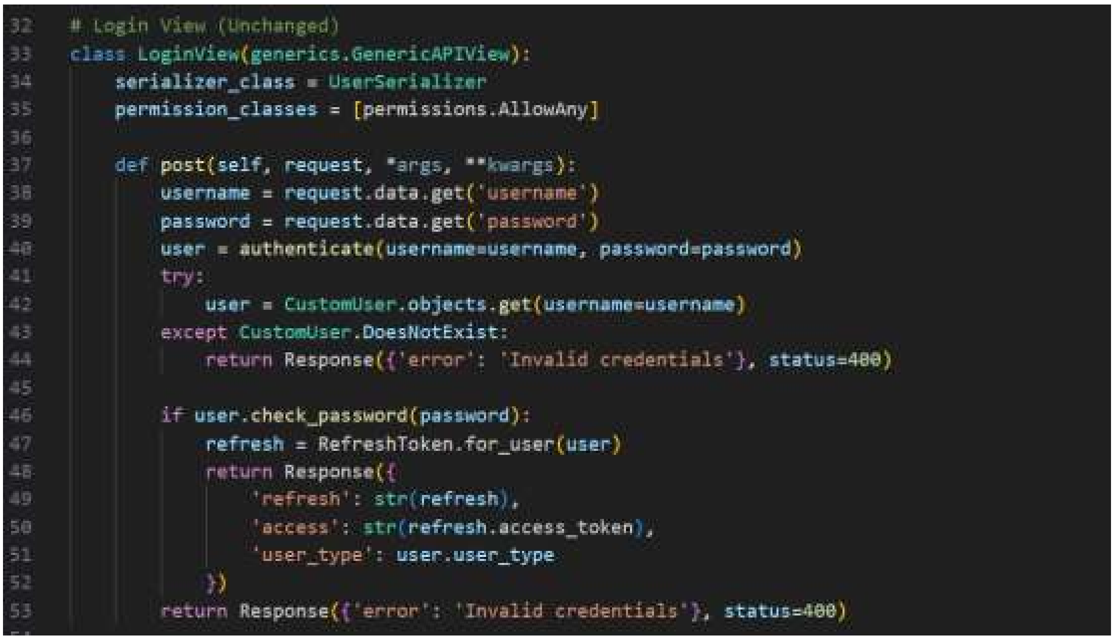
The backend architecture is built using Django and Django REST Framework, which powers all major functionalities such as user authentication, CRUD operations for projects, and user profile management. The frontend is developed using React, leveraging component-based structure for dynamic rendering and smooth routing. JWT tokens are used to authenticate and authorize requests securely. The system architecture includes:

- A Custom User model to differentiate roles (student, teacher, recruiter, admin)
- Model associations for students, teachers, projects, and departments

RESTful APIs for modular and scalable endpoints.

4.2 USER REGISTRATION AND AUTHENTICATION

The authentication module supports registration and login for four user types: student, teacher, recruiter, and admin. On registration, the user is assigned a role, and role-specific models (e.g., Student, Teacher) are instantiated. JWT-based login tokens are generated using the SimpleJWT package, which are stored in local storage on the client side. APIs are protected using authentication and permission classes to ensure only authorized users can access or modify data.

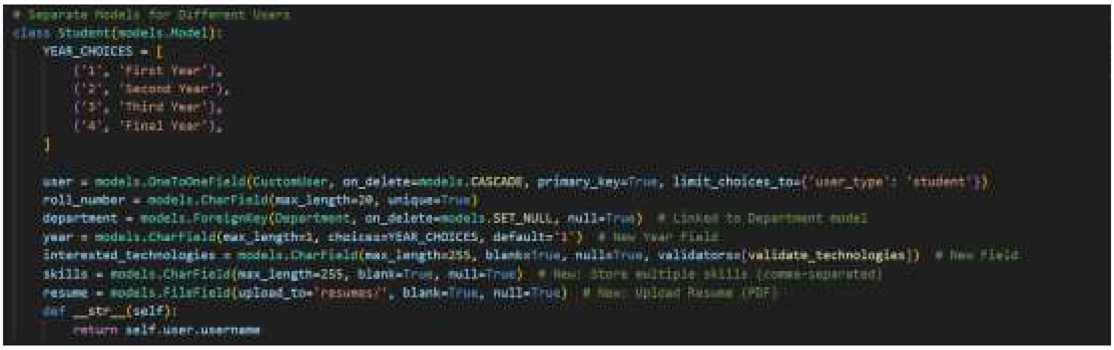


```
# Login View (Unchanged)
class LoginView(generics.GenericAPIView):
    serializer_class = UserSerializer
    permission_classes = [permissions.AllowAny]

    def post(self, request, *args, **kwargs):
        username = request.data.get('username')
        password = request.data.get('password')
        user = authenticate(username=username, password=password)
        try:
            user = CustomUser.objects.get(username=username)
        except CustomUser.DoesNotExist:
            return Response({'error': 'Invalid credentials'}, status=400)

        if user.check_password(password):
            refresh = RefreshToken.for_user(user)
            return Response({
                'refresh': str(refresh),
                'access': str(refresh.access_token),
                'user_type': user.user_type
            })
        return Response({'error': 'Invalid credentials'}, status=400)
```

4.3 PROFILE MANAGEMENT SYSTEM

Each student has a detailed profile that includes a profile picture, year, department, interested technologies, resume, and a dynamically updating project list. Students can upload resumes in PDF format and update their skills via the frontend. The profile page displays:

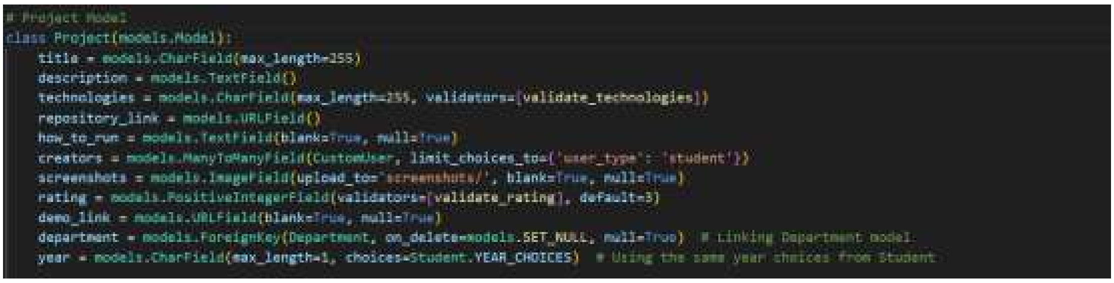
- Profile image and name
- Department and year
- Editable fields like skills and resume
- Associated projects (to be implemented later)



```
# Separate Models for Different Users
class Student(models.Model):
    YEAR_CHOICES = [
        ('1', 'First Year'),
        ('2', 'Second Year'),
        ('3', 'Third Year'),
        ('4', 'Final Year'),
    ]

    user = models.OneToOneField(CustomUser, on_delete=models.CASCADE, primary_key=True, limit_choices_to={'user_type': 'student'})
    roll_number = models.CharField(max_length=20, unique=True)
    department = models.ForeignKey(Department, on_delete=models.SET_NULL, null=True)  # Linked to Department model
    year = models.CharField(max_length=1, choices=YEAR_CHOICES, default='1')  # New Year Field
    interested_technologies = models.CharField(max_length=255, blank=True, null=True, validators=[validate_technologies])  # New Field
    skills = models.CharField(max_length=255, blank=True, null=True)  # New: Store multiple skills (comma-separated)
    resume = models.FileField(upload_to='resumes/', blank=True, null=True)  # New: Upload Resume (PDF)
    def __str__(self):
        return self.user.username
```

4.4 PROJECT UPLOAD AND MANAGEMENT

Students can upload their projects using a dedicated form, including title, description, tech stack, screenshots, and repository/demo links. Each project supports multiple creators (selected from registered students), and is stored in a many-to-many relationship. The backend validates technologies and limits inputs where needed (e.g., max 5 skills). Top and trending projects are sorted based on ratings for easy discoverability.

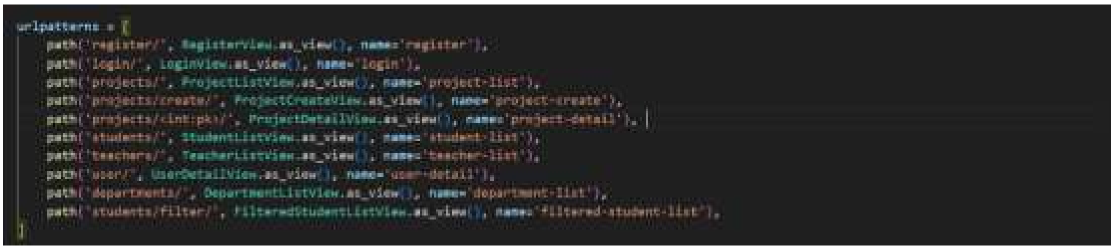


```python
# Project Model
class Project(models.Model):
    title = models.CharField(max_length=255)
    description = models.TextField()
    technologies = models.CharField(max_length=255, validators=[validate_technologies])
    repository_link = models.URLField()
    how_to_run = models.TextField(blank=True, null=True)
    creators = models.ManyToManyField(CustomUser, limit_choices_to={'user_type': 'student'})
    screenshots = models.ImageField(upload_to='screenshots/', blank=True, null=True)
    rating = models.PositiveIntegerField(validators=[validate_rating], default=3)
    demo_link = models.URLField(blank=True, null=True)
    department = models.ForeignKey(Department, on_delete=models.SET_NULL, null=True)  # Linking Department model
    year = models.CharField(max_length=1, choices=Student.YEAR_CHOICES)  # Using the same year choices from Student
```

4.5 API ENDPOINTS AND ROUTING REST API

Api endpoints and routing rest api endpoints are implemented to manage:

- User details (/auth/user/)
- Project listings (/auth/projects/, /auth/project/<id>/)
- Profile editing (/auth/student/profile/)
- File uploads (profile photo, resume)



```python
urlpatterns = [
    path('register/', RegisterView.as_view(), name='register'),
    path('login/', LoginView.as_view(), name='login'),
    path('projects/', ProjectListView.as_view(), name='project-list'),
    path('projects/create/', ProjectCreateView.as_view(), name='project-create'),
    path('projects/<int:pk>/', ProjectDetailView.as_view(), name='project-detail'),
    path('students/', StudentListView.as_view(), name='student-list'),
    path('teachers/', TeacherListView.as_view(), name='teacher-list'),
    path('user/', UserDetailView.as_view(), name='user-detail'),
    path('departments/', DepartmentListView.as_view(), name='department-list'),
    path('students/filter/', FilteredStudentListView.as_view(), name='filtered-student-list'),
]
```

4.6. Technology Stack

1 React

React is employed for building the frontend, offering users a seamless experience with features like smooth page switches, fast loading and added features for extra security, compiling and storing HTML pages in the backend without revealing details to users.

React is a widely-used JavaScript library employed for frontend development, particularly favored for its ability to create dynamic and interactive user interfaces. It enhances user experience by providing features such as smooth page transitions, fast loading times, and additional security measures. One of React's key advantages lies in its virtual DOM (Document Object Model), which enables efficient rendering of UI components and ensures rapid updates without requiring a full page reload. This contributes to the seamless user experience mentioned, as React optimizes the rendering process for improved performance. Moreover, React allows for the creation of reusable UI components, promoting code reusability and maintainability. Additionally, React can be integrated with backend systems to compile and store HTML pages without exposing sensitive details to end-users, thus enhancing security. This combination of features makes React a popular choice for frontend development, empowering developers to craft engaging and secure user interfaces for web applications.

2 Node.js

Node.js is a server-side JavaScript runtime environment that facilitates the execution of JavaScript code outside the browser, enabling efficient and scalable web application development. It is recognized for its non-blocking, event-driven architecture, enhancing the responsiveness of applications.

Node.js is a server-side JavaScript runtime environment renowned for its ability to execute JavaScript code outside of the browser, thus enabling the development of efficient and scalable web applications. One of Node.js's standout features is its non-blocking, event-driven architecture, which significantly enhances application responsiveness. Unlike traditional server-side technologies that utilize a synchronous, blocking approach, Node.js employs an asynchronous model, allowing multiple operations to be executed concurrently without waiting for each one to complete. This non-blocking nature ensures that Node.js applications can handle large numbers of simultaneous connections without becoming sluggish or unresponsive. Additionally, Node.js boasts a rich ecosystem of libraries and frameworks, further simplifying the development process and empowering developers to build robust, high-performance web applications. Overall, Node.js has revolutionized server-side development by offering a lightweight, scalable solution that leverages JavaScript's versatility and popularity.

3 Python

Python is an interpreted, high-level programming language known for its simplicity, readability, and versatility. Created by Guido van Rossum and first released in 1991, Python is popular for its clean, easy-to-understand syntax that emphasizes readability, making it an ideal choice for both beginners and experienced developers. Its use of indentation to define code blocks instead of braces makes the code more structured and readable. This simplicity allows developers to focus on problem-solving rather than language intricacies.

As an interpreted language, Python executes code line by line, facilitating rapid testing and iteration. While this can lead to slower execution compared to compiled languages, Python's performance can be enhanced using optimized libraries or integrating compiled code for performance-critical tasks. Additionally, Python is dynamically typed, meaning developers don’t need to declare variable types explicitly, which adds flexibility but can sometimes lead to runtime errors, though Python’s clear error messages help with troubleshooting.

Python’s extensive standard library provides built-in modules for tasks such as file handling, data manipulation, and network communication, reducing the need for external dependencies. It is also cross-platform, meaning Python programs can run on Windows, macOS, and Linux without modifications, making it highly versatile for developing software that works across different environments. Python is widely used in various domains such as web development, data science, machine learning, automation, and scientific computing. In web development, frameworks like Django and Flask simplify the creation of dynamic web applications. In data science and machine learning, Python’s libraries—such as NumPy, Pandas, Matplotlib, and Scikit-learn—offer powerful tools for data manipulation, analysis, and modelling. Python is also a popular choice for scripting and automating tasks, making workflows more efficient. In scientific computing, libraries like SciPy and SymPy support complex mathematical modelling and simulations, while tools like Jupyter Notebooks provide interactive coding environments.

Python's large, active community contributes to its growth and success, offering extensive resources such as tutorials, documentation, and forums to help developers. Its broad ecosystem of third-party libraries and frameworks further extends its functionality and makes it a versatile choice across industries.

4 Django

Django is a high-level, open-source web framework for building web applications quickly and with minimal hassle. It is written in Python and follows the model-template-views (MTV) architectural pattern, which is similar to the widely used model-view-controller (MVC) pattern. Django was created by Adrian Holovaty and Simon Willison in 2003 and released publicly in 2005. It emphasizes the reuse of components, rapid development, and a clean, pragmatic design, making it one of the most popular frameworks for web development. One of Django's core strengths is its batteries-included philosophy, meaning that it comes with a lot of built-in features that help developers avoid reinventing the wheel. These include an admin interface, authentication system, database ORM (Object-Relational Mapping), URL routing, and form handling. This makes it easy for developers to build full-featured web applications with minimal external dependencies. The admin interface is one of Django's standout features. It allows developers to quickly create a backend interface for managing application data. This automatically generated interface supports data models, making it incredibly useful for managing the content and users of a website without writing extra code.

Django's ORM (Object-Relational Mapping) system provides a way to interact with databases using Python code rather than raw SQL queries. This makes it easier for developers to work with databases by abstracting away complex SQL operations into Python objects, ensuring that they can focus on building their application logic rather than handling database intricacies. The ORM supports various databases, including PostgreSQL, MySQL, SQLite, and Oracle. Django's URL routing system allows developers to define clean, human-readable URLs and associate them with specific views in the application. This feature enhances SEO and user experience while ensuring that web applications are organized and maintainable. Another key feature of Django is its security. The framework provides built-in protections against common web vulnerabilities, such as cross-site scripting (XSS), SQL injection, and cross-site request forgery (CSRF). Django also encourages the use of secure password hashing and provides tools for user authentication, making it easier to build secure web applications. Django follows the DRY (Don’t Repeat Yourself) principle, which encourages code reusability and maintainability. By promoting modular code and providing reusable components, developers can avoid redundant code, leading to more efficient development and easier maintenance.

5 Postman API

Postman is a popular tool used for testing and interacting with APIs (Application Programming Interfaces). It is widely used by developers, testers, and even non-technical users for sending requests to APIs, inspecting responses, and automating various aspects of API workflows. Originally launched as a Chrome extension, Postman has since evolved into a standalone, cross-platform application that supports testing for REST, SOAP, and GraphQL APIs. At its core, Postman simplifies the process of interacting with APIs by providing an intuitive graphical interface. It eliminates the need to write complex code or use command-line tools to test API endpoints, making it accessible for both beginners and experienced developers. With Postman, users can easily create and manage HTTP requests, view the response, and validate API functionality. One of the main features of Postman is its request-building interface, which allows users to craft HTTP requests (GET, POST, PUT, DELETE, etc.) by entering the URL, headers, query parameters, and request body. This is useful for testing various API endpoints and exploring how the server responds to different types of inputs. It also supports authentication protocols like OAuth 2.0, API Key, and Bearer Token, making it easy to test secure APIs.

Postman provides rich functionality for working with collections, which are groups of related API requests. Developers can organize requests into collections, making it easier to run a series of tests or automate repetitive tasks. Postman also supports environments, which allow users to define different sets of variables (such as base URLs, authentication tokens, or request parameters) that can be reused across multiple requests and collections. This is particularly useful for testing APIs in different environments, such as development, staging, and production. Another key feature of Postman is its ability to handle API responses. When a request is sent, Postman displays the response data in various formats, including JSON, XML, and HTML, and allows users to inspect the response body, status code, headers, and cookies. This makes it easy to verify the correctness of an API and troubleshoot any issues. Postman also includes a built-in console that logs all requests and responses, making it easier to debug issues during testing. Automation is another significant strength of Postman. It supports writing tests and scripts using JavaScript, which allows developers to automate API testing by running assertions on response data. Postman’s testing capabilities are useful for checking if the API is behaving as expected. For instance, you can check whether a specific field in the response body is correct or whether the API returns the expected status code. These tests can be run automatically as part of a CI/CD (Continuous Integration/Continuous Deployment) pipeline, improving the efficiency of testing and quality assurance processes.

CHAPTER 5

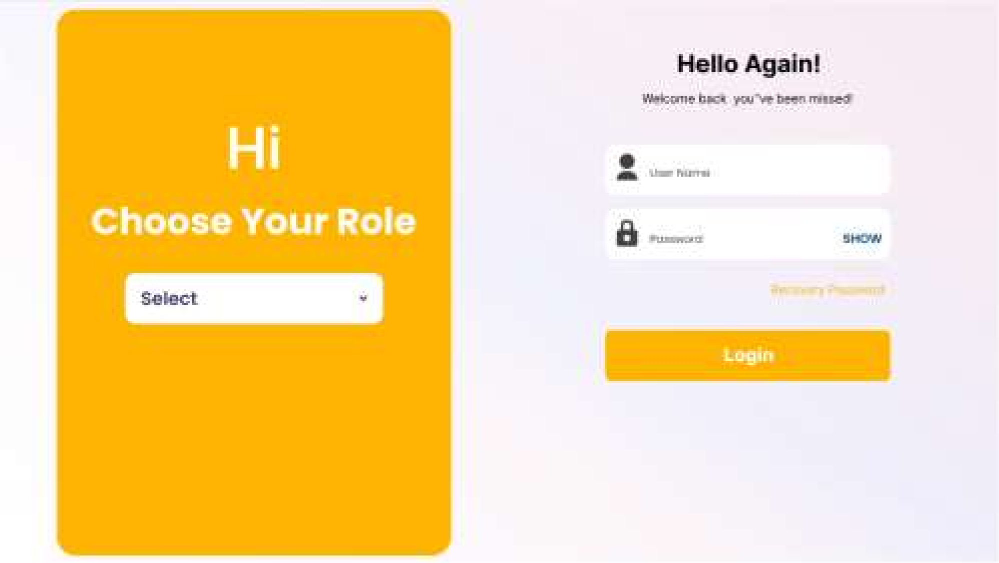
RESULT

AND

DISCUSSION

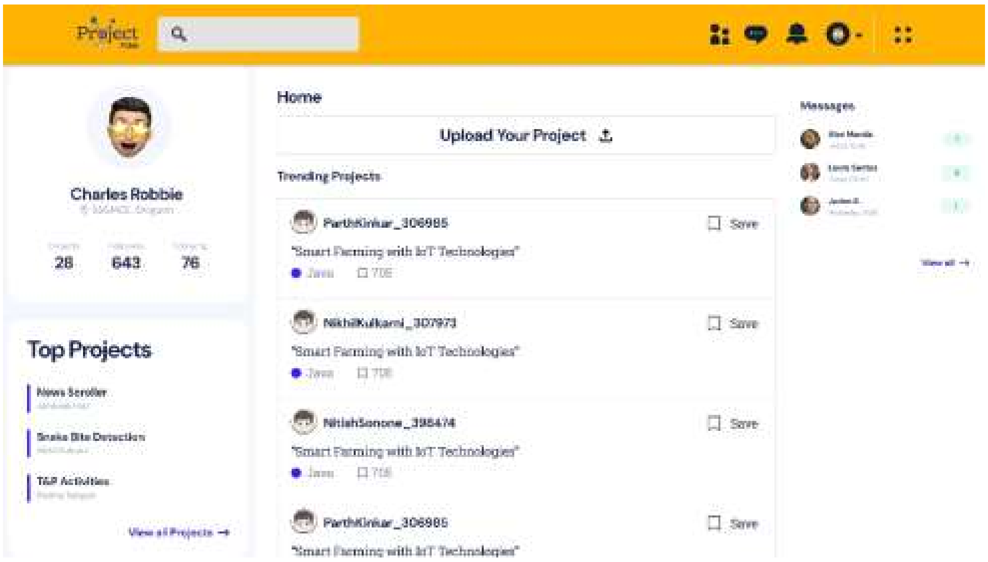
RESULT AND DISCUSSION

5.1 AUTHENTICATION: This page handles secure login for all users—students, faculty, and recruiters. It verifies credentials and redirects users to their role-specific dashboards. It may include options like registration or password reset.



Screenshot 5.1 Authentication page

5.2 STUDENT: Students can view and manage their projects from this dashboard. They can edit profiles, add projects and view feedback or ratings. The page serves as a hub for student activities on the platform.



Charles Robbie

28 643 76

News Scroller

Snake Bite Detection

T&P Activities

ParthKinkar_306985
"Smart Farming with IoT Technologies"
Java 706

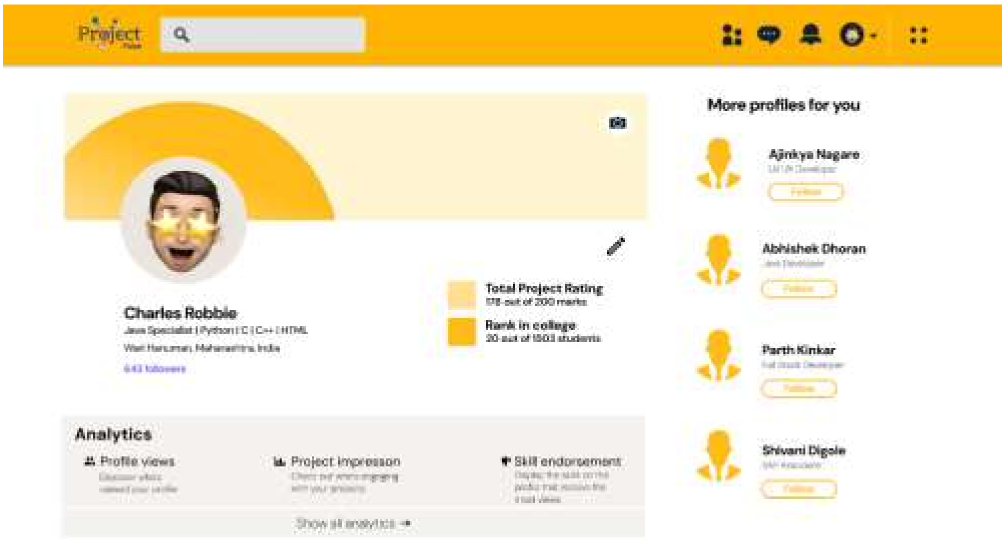
NikhilKulkarni_307973
"Smart Farming with IoT Technologies"
Java 706

NitishSonone_398474
"Smart Farming with IoT Technologies"
Java 706

ParthKinkar_306985
"Smart Farming with IoT Technologies"

Screenshot 5.2 Student Page

5.3 FACULTY: Faculty can review student projects, track performance, give feedback, and use ratings for academic evaluation. They can also monitor project trends, ensure quality submissions, and identify top-performing students.



Charles Robbie
Java Specialist | Python | C | C++ | HTML
Wari Hanuman, Maharashtra, India
643 followers

Total Project Rating
178 out of 200 marks

Rank in college
20 out of 1503 students

Analytics

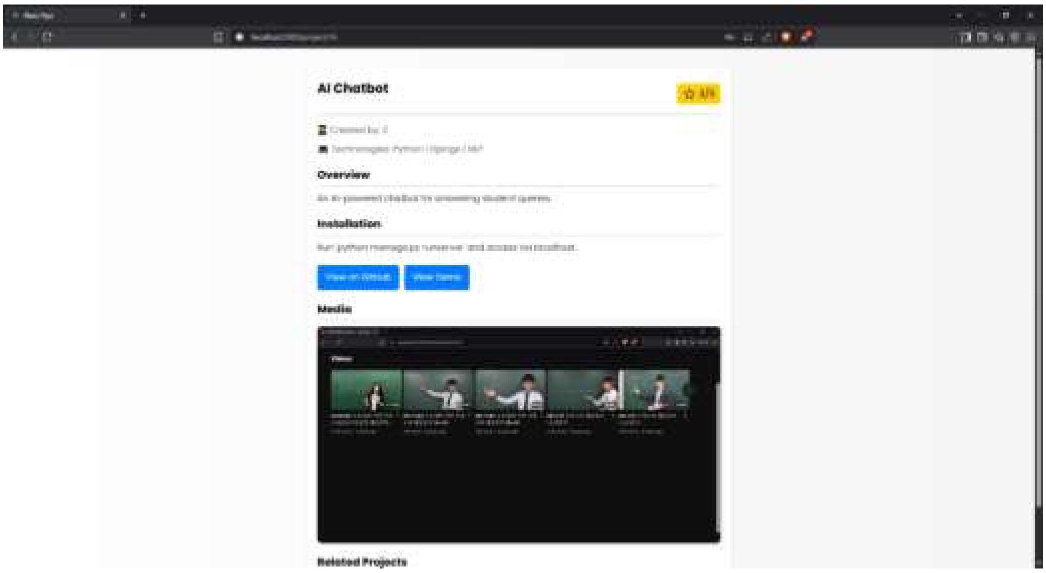
Profile views
Discover who's viewed your profile

Project impression
Check out who's engaging with your projects

Skill endorsement
Display the skills on the profile that receive the most views

Screenshot 5.3 Faculty Page

5.4 PROJECT: Displays all details of an individual project, including title, description, technologies used, and team members. Users can also see ratings and comments. GitHub or external links may be available for deeper exploration.



AI Chatbot

Created by: 2

Technologies: Python | Django | NLP

Overview

An AI-powered chatbot for answering student queries.

Installation

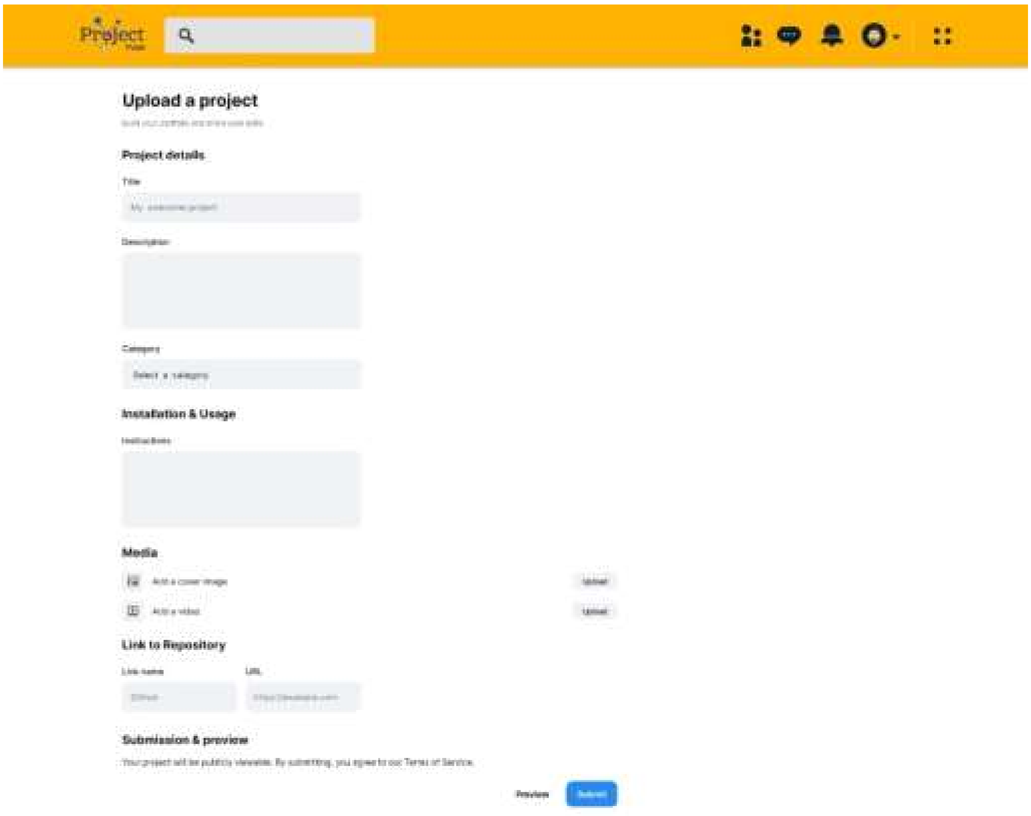
Run `python manage.py runserver` and access via localhost.

Media

Related Projects

Screenshot 5.4 Search and Sort Page

5.5 UPLOAD: Students can upload their project details like project title, description, Installation guide, screenshots, github repository link.



Upload a project

[illegible]

Project details

Title

My awesome project

Description

Category

Select a category

Installation & Usage

Instructions

Media

Add a cover image

Add a video

Link to Repository

Link name

GitHub

URL

https://example.com

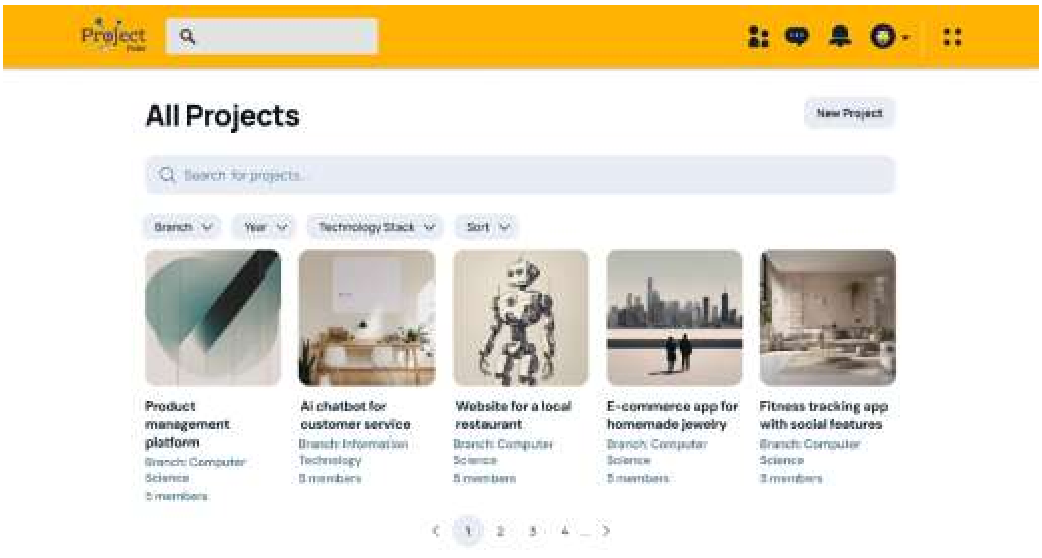
Submission & preview

Your project will be publicly viewable. By submitting, you agree to our Terms of Service.

Submit

Screenshot 5.5 Upload Page

5.6 All Projects : This page displays all the submitted and approved projects in one place for easy viewing. Recruiters can browse through these projects and evaluate them for hiring opportunities. A search and filter feature is provided to help sort projects based on different criteria. It makes project discovery and talent scouting convenient and efficient.



All Projects

Product management platform
Branch: Computer Science
5 members

AI chatbot for customer service
Branch: Information Technology
8 members

Website for a local restaurant
Branch: Computer Science
5 members

E-commerce app for homemade jewelry
Branch: Computer Science
5 members

Fitness tracking app with social features
Branch: Computer Science
3 members

Screenshot 5.6 All Projects

CHAPTER 6

CONCLUSION

CONCLUSION

6.1 CONCLUSION

The ProjectPulse platform was developed and tested in a real-world academic environment, enabling students, faculty, and industry professionals to register projects, track progress, and engage in mentorship. The platform significantly enhanced project visibility, bridging the gap between student work and industry expectations. Faculty and industry experts found that ProjectPulse streamlined workflow by centralizing project records and improving mentorship. Easy access to student projects allowed faculty to provide timely guidance, while structured project sharing attracted professionals, fostering valuable feedback and networking opportunities. Students reported that the showcase feature simplified project presentation, eliminating visibility challenges and allowing them to focus on refining their work. Data analysis showed a steady increase in project registrations and completions, with students benefiting from mentorship and timely feedback. Employers found the platform useful for identifying skilled candidates, strengthening the connection between academia and industry.

6.2 FUTURE SCOPE

As ProjectPulse evolves, several enhancements can be integrated to expand its functionality and engagement among students, recruiters, and educators. These future developments will further strengthen the platform's role in bridging academia and industry.

1. Hackathons and Competitions A dedicated section for hackathons where students can participate in coding and problem-solving challenges. Competitions hosted by universities and industries to encourage innovation and skill-building.

2. Project Guide Tutorials and Workshops A knowledge hub featuring tutorials and mentorship programs to help students improve their project development skills. Live and recorded workshops on emerging technologies, research methodologies, and project management best practices.

3. Industry-Provided Real-Time Problem Statements Companies and startups can post real-world challenges for students to solve, simulating a real work environment.

Students who provide innovative solutions can receive rewards such as internships, scholarships, or cash prizes.

4. AI-Based Skill Matching and Career Guidance AI-driven analysis to suggest projects based on students' skills and career aspirations. Personalized recommendations for job opportunities and recruiter connections.

5. Enhanced Collaboration and Networking Improved team formation features that help students find collaborators based on their interests and expertise. Forums and discussion boards for knowledge exchange, guidance, and networking with industry professionals. By implementing these features, ProjectPulse can become a comprehensive platform that not only showcases student projects but also fosters innovation, learning, and career growth.

REFERENCES

REFERENCES

[1] N. Wald and T. Harland, “Graduate attributes frameworks or powerful knowledge?,” Journal of Higher Education Policy and Management, vol. 41, no. 4, pp. 361–374, May 2019, doi: 10.1080/1360080x.2019.1613310.

[2] E. Malherbe and M.-A. Aufaure, “Bridge the terminology gap between recruiters and candidates: A multilingual skills base built from social media and linked data,” 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Aug. 2016, doi: 10.1109/asonam.2016.7752295.

[3] M. Abelha, S. Fernandes, D. Mesquita, F. Seabra, and A. T. Ferreira-Oliveira, “Graduate Employability and Competence Development in Higher Education—A Systematic Literature Review using PRISMA,” Sustainability, vol. 12, no. 15, p. 5900, Jul. 2020, doi: 10.3390/su12155900.

[4] S. Chaudhuri, U. Dayal, and V. Ganti, “Database technology for decision support systems,” Computer, vol. 34, no. 12, pp. 48–55, Jan. 2001, doi: 10.1109/2.970575.

[5] O. Moscoso-Zea, Andres-Sampedro, and S. Luján-Mora, “Datawarehouse design for educational data mining,” in 2016 15th International Conference on Information Technology Based Higher Education and Training, ITHET 2016, Sep. 2016, pp. 1–6, doi: 10.1109/ITHET.2016.7760754.

[6] Begel, A., Bosch, J., and Storey, M.-A. Social networking meets software development: Perspectives from github, msdn, stack exchange, and topcoder.

Software, IEEE 30, 1 (2013), 52–66.

[7] Farooq, U., Carroll, J. M., and Ganoe, C. H. Supporting creativity with awareness in distributed collaboration. In Proceedings of the 2007 international ACM conference on Supporting group work, ACM (2007), 31–40.

[8] Anderson, E. and Shannon, A. (1988) ‘Towards a conceptualization of mentoring’, Journal of Teacher Education, 39(1), pp. 38-42. http://dx.doi.org/10.1177/002248718803900109

[9] Beckers, J., Dolmans, D., & Van Merriënboer, J. (2016). e-Portfolios enhancing students' self-directed learning: A systematic review of influencing factors. Australasian Journal of Educational Technology, 32(2), 2.

[10] Bernier, A., Larose, S. and Soucy, N. (2005) 'Academic mentoring in college: the interactive role of student's and mentor's dispositions', Research in Higher Education, 46(1), pp. 29-51. http://dx.doi.org/10.1007/s11162-004-6288-5

[11] Christie, H. (2014) 'Peer mentoring in higher education: issues of power and control', Teaching in Higher Education, 19(8), pp. 955-965. http://dx.doi.org/10.1080/13562517.2014.934355

[12] Cook, A. and Rushton, B. (2009) How to recruit and retain higher education students: a handbook of good practice. Oxon: Routledge

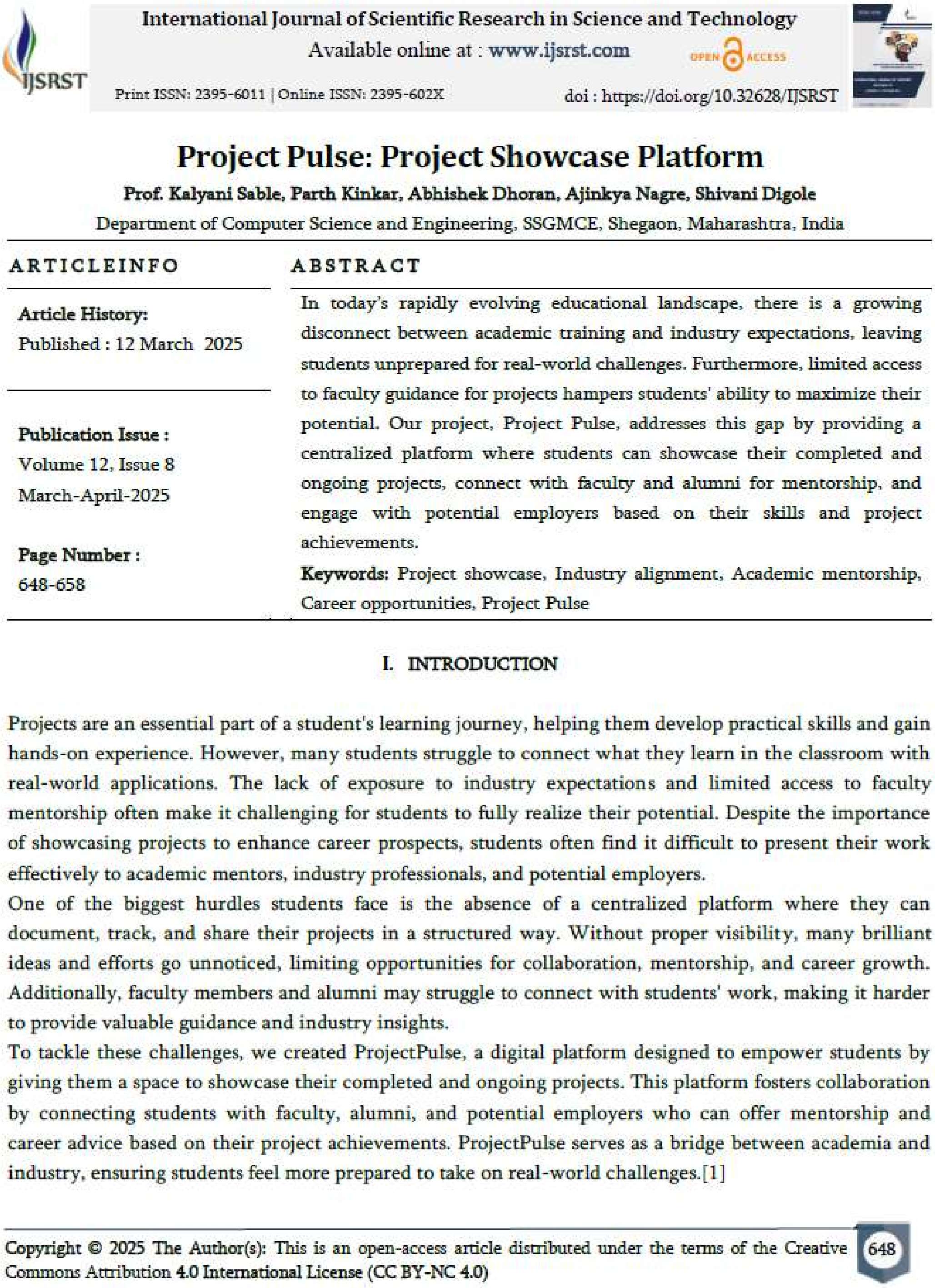
[13] World Economic Forum, "The future of jobs report 2020 | world economic forum." www.weforum.org. https://www.weforum.org/reports/the-future-of-jobs-report-2020/digest (accessed Nov. 26 2021).

[14] E. Malherbe and M. A. Aufaure, "Bridge the terminology gap between recruiters and candidates: A multilingual skills base built from social media and linked data," in Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016, Aug. 2016, pp. 583–590, doi: 10.1109/ASONAM.2016.7752295.

[15] E.-R. Richiţeanu-Năstase and C. Stăiculescu, "The impact of career factors on students professional insertion. what measures to be taken by the university?," Procedia - Social and Behavioral Sciences, vol. 180, pp. 1102–1108, May 2015, doi: 10.1016/j.sbspro.2015.02.216

DISSEMINATIOn OF WORK

DISSEMINATION OF WORK



International Journal of Scientific Research in Science and Technology
Available online at : www.ijsrst.com
OPEN ACCESS
Print ISSN: 2395-6011 | Online ISSN: 2395-602X
doi : https://doi.org/10.32628/IJSRST

Project Pulse: Project Showcase Platform

Prof. Kalyani Sable, Parth Kinkar, Abhishek Dhoran, Ajinkya Nagre, Shivani Digole
Department of Computer Science and Engineering, SSGMCE, Shegaon, Maharashtra, India

ARTICLEINFO

Article History:
Published : 12 March 2025

Publication Issue :
Volume 12, Issue 8
March-April-2025

Page Number :
648-658

ABSTRACT

In today's rapidly evolving educational landscape, there is a growing disconnect between academic training and industry expectations, leaving students unprepared for real-world challenges. Furthermore, limited access to faculty guidance for projects hampers students' ability to maximize their potential. Our project, Project Pulse, addresses this gap by providing a centralized platform where students can showcase their completed and ongoing projects, connect with faculty and alumni for mentorship, and engage with potential employers based on their skills and project achievements.

Keywords: Project showcase, Industry alignment, Academic mentorship, Career opportunities, Project Pulse

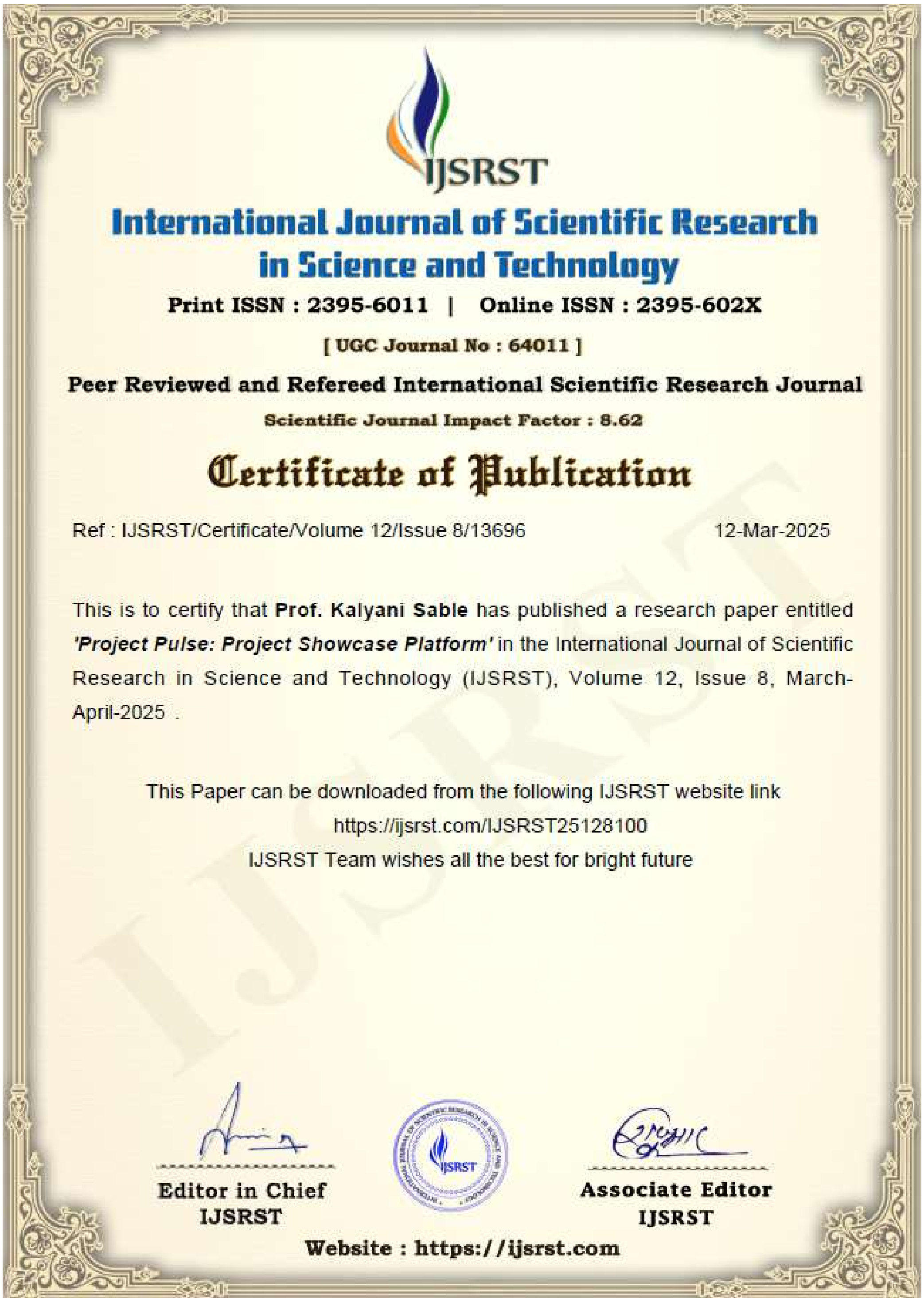
I. INTRODUCTION

Projects are an essential part of a student's learning journey, helping them develop practical skills and gain hands-on experience. However, many students struggle to connect what they learn in the classroom with real-world applications. The lack of exposure to industry expectations and limited access to faculty mentorship often make it challenging for students to fully realize their potential. Despite the importance of showcasing projects to enhance career prospects, students often find it difficult to present their work effectively to academic mentors, industry professionals, and potential employers.

One of the biggest hurdles students face is the absence of a centralized platform where they can document, track, and share their projects in a structured way. Without proper visibility, many brilliant ideas and efforts go unnoticed, limiting opportunities for collaboration, mentorship, and career growth. Additionally, faculty members and alumni may struggle to connect with students' work, making it harder to provide valuable guidance and industry insights.

To tackle these challenges, we created ProjectPulse, a digital platform designed to empower students by giving them a space to showcase their completed and ongoing projects. This platform fosters collaboration by connecting students with faculty, alumni, and potential employers who can offer mentorship and career advice based on their project achievements. ProjectPulse serves as a bridge between academia and industry, ensuring students feel more prepared to take on real-world challenges.[1]

CERTIFICATES



IJSRST

International Journal of Scientific Research in Science and Technology

Print ISSN : 2395-6011 | Online ISSN : 2395-602X

[UGC Journal No : 64011]

Peer Reviewed and Refereed International Scientific Research Journal

Scientific Journal Impact Factor : 8.62

Certificate of Publication

Ref : IJSRST/Certificate/Volume 12/Issue 8/13696 12-Mar-2025

This is to certify that **Prof. Kalyani Sable** has published a research paper entitled ***'Project Pulse: Project Showcase Platform'*** in the International Journal of Scientific Research in Science and Technology (IJSRST), Volume 12, Issue 8, March-April-2025 .

This Paper can be downloaded from the following IJSRST website link

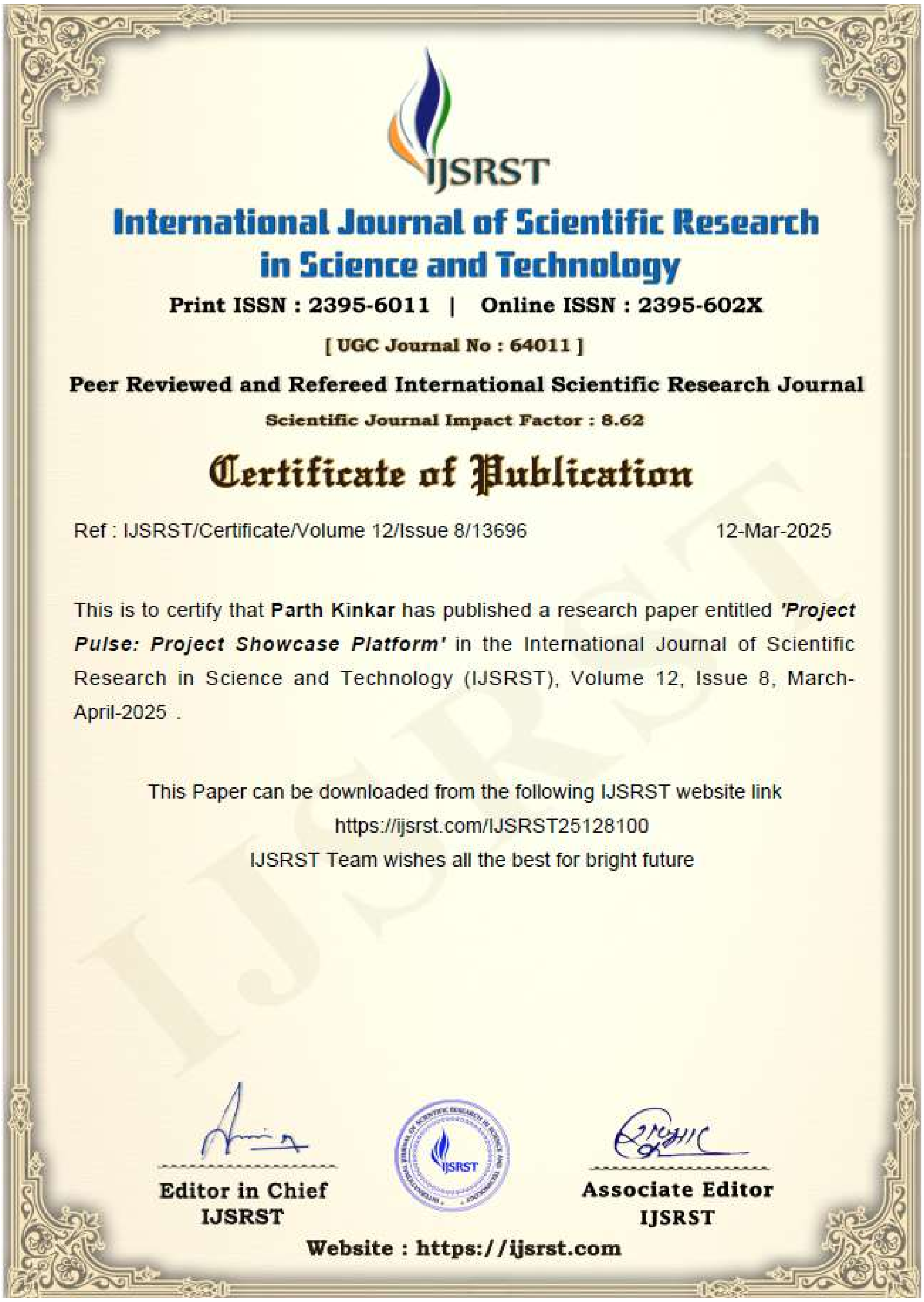
https://ijsrst.com/IJSRST25128100

IJSRST Team wishes all the best for bright future

Editor in Chief
IJSRST

Associate Editor
IJSRST

Website : https://ijsrst.com



IJSRST

International Journal of Scientific Research in Science and Technology

Print ISSN : 2395-6011 | Online ISSN : 2395-602X

[UGC Journal No : 64011]

Peer Reviewed and Refereed International Scientific Research Journal

Scientific Journal Impact Factor : 8.62

Certificate of Publication

Ref : IJSRST/Certificate/Volume 12/Issue 8/13696 12-Mar-2025

This is to certify that **Parth Kinkar** has published a research paper entitled ***'Project Pulse: Project Showcase Platform'*** in the International Journal of Scientific Research in Science and Technology (IJSRST), Volume 12, Issue 8, March-April-2025 .

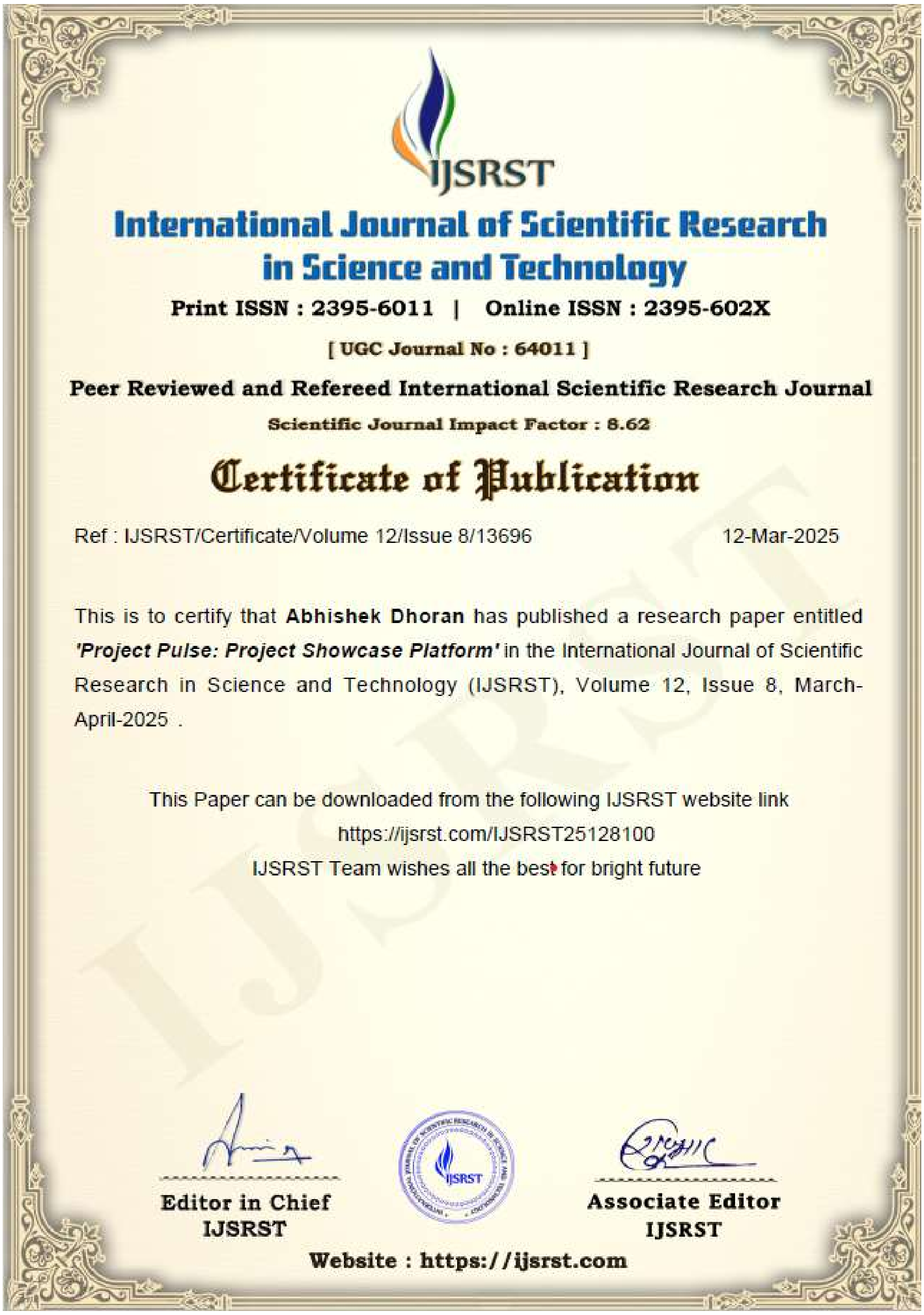
This Paper can be downloaded from the following IJSRST website link
https://ijsrst.com/IJSRST25128100
IJSRST Team wishes all the best for bright future

Editor in Chief
IJSRST

IJSRST

Associate Editor
IJSRST

Website : https://ijsrst.com



IJSRST

International Journal of Scientific Research in Science and Technology

Print ISSN : 2395-6011 | Online ISSN : 2395-602X

[UGC Journal No : 64011]

Peer Reviewed and Refereed International Scientific Research Journal

Scientific Journal Impact Factor : 8.62

Certificate of Publication

Ref : IJSRST/Certificate/Volume 12/Issue 8/13696 12-Mar-2025

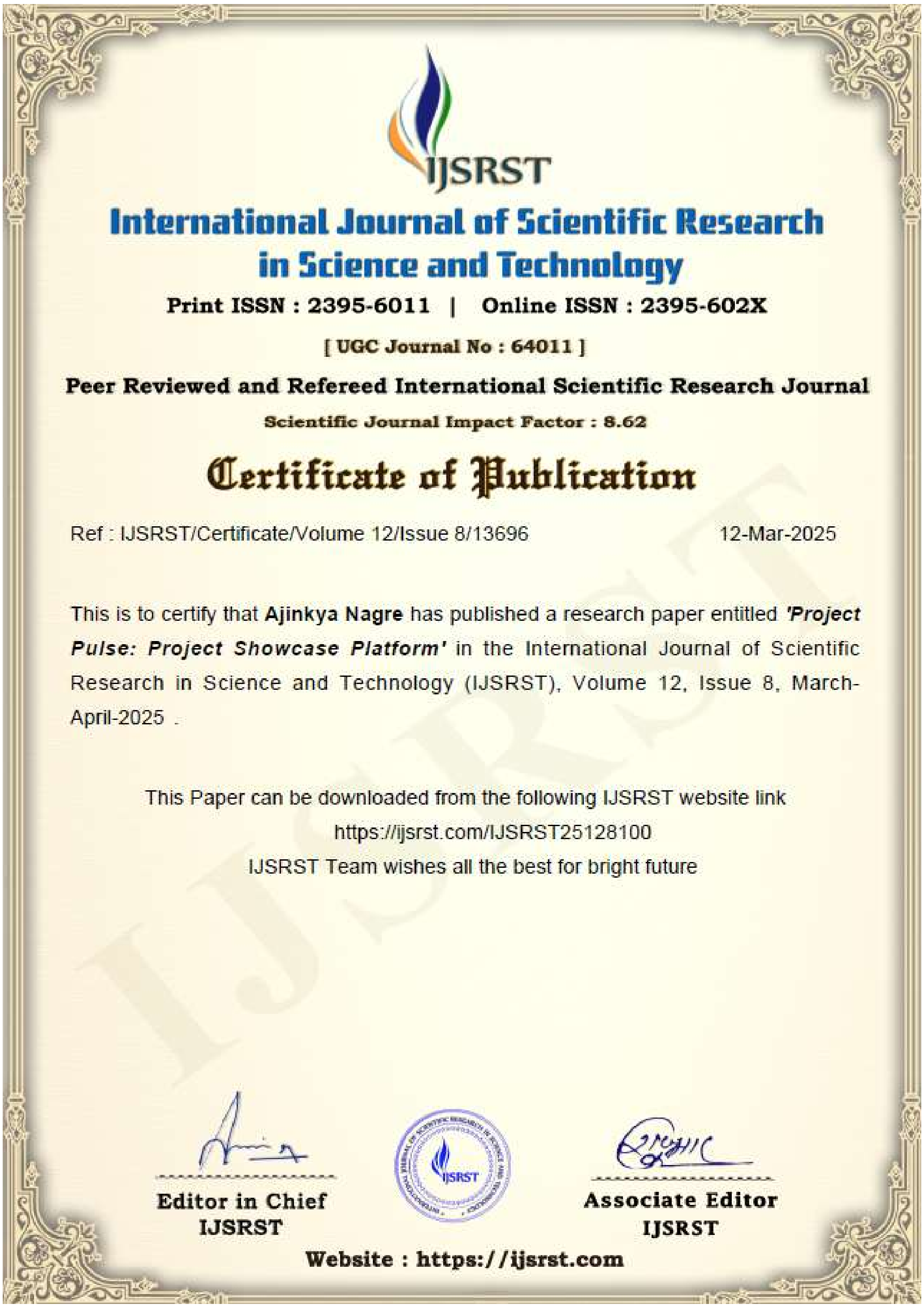
This is to certify that **Abhishek Dhoran** has published a research paper entitled ***'Project Pulse: Project Showcase Platform'*** in the International Journal of Scientific Research in Science and Technology (IJSRST), Volume 12, Issue 8, March-April-2025 .

This Paper can be downloaded from the following IJSRST website link
https://ijsrst.com/IJSRST25128100
IJSRST Team wishes all the best for bright future

Editor in Chief
IJSRST

Associate Editor
IJSRST

Website : https://ijsrst.com



IJSRST

International Journal of Scientific Research in Science and Technology

Print ISSN : 2395-6011 | Online ISSN : 2395-602X

[UGC Journal No : 64011]

Peer Reviewed and Refereed International Scientific Research Journal

Scientific Journal Impact Factor : 8.62

Certificate of Publication

Ref : IJSRST/Certificate/Volume 12/Issue 8/13696 12-Mar-2025

This is to certify that **Ajinkya Nagre** has published a research paper entitled ***'Project Pulse: Project Showcase Platform'*** in the International Journal of Scientific Research in Science and Technology (IJSRST), Volume 12, Issue 8, March-April-2025 .

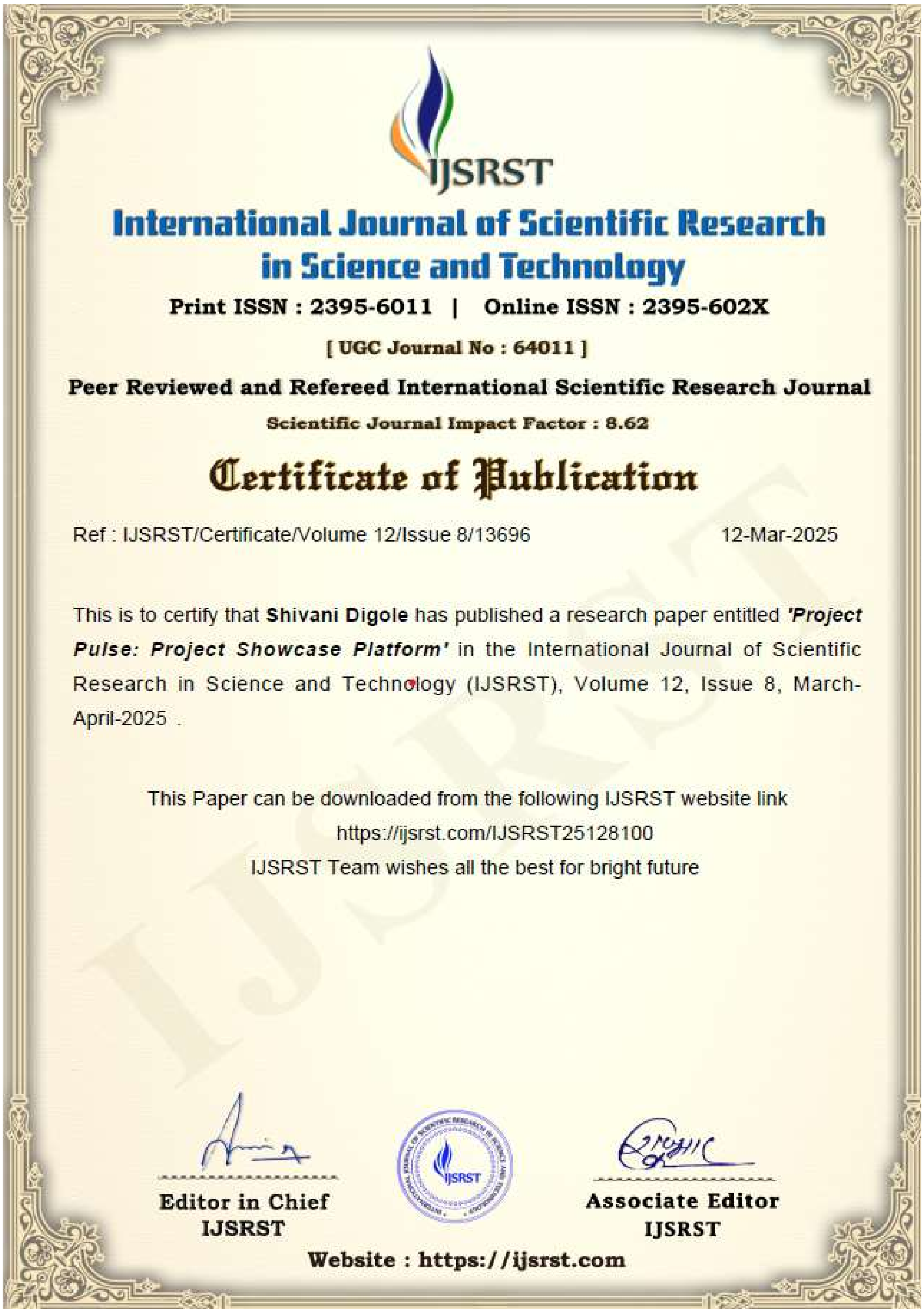
This Paper can be downloaded from the following IJSRST website link
https://ijsrst.com/IJSRST25128100
IJSRST Team wishes all the best for bright future

Editor in Chief
IJSRST

IJSRST

Associate Editor
IJSRST

Website : https://ijsrst.com



IJSRST

International Journal of Scientific Research in Science and Technology

Print ISSN : 2395-6011 | Online ISSN : 2395-602X

[UGC Journal No : 64011]

Peer Reviewed and Refereed International Scientific Research Journal

Scientific Journal Impact Factor : 8.62

Certificate of Publication

Ref : IJSRST/Certificate/Volume 12/Issue 8/13696 12-Mar-2025

This is to certify that **Shivani Digole** has published a research paper entitled ***'Project Pulse: Project Showcase Platform'*** in the International Journal of Scientific Research in Science and Technology (IJSRST), Volume 12, Issue 8, March-April-2025 .

This Paper can be downloaded from the following IJSRST website link
https://ijsrst.com/IJSRST25128100
IJSRST Team wishes all the best for bright future

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN SCIENCE AND TECHNOLOGY
IJSRST

Editor in Chief
IJSRST

Associate Editor
IJSRST

Website : https://ijsrst.com

PROJECT GROUP MEMBERS

Name: Parth Sachin Kinkar

Address: Flat FB/2, Mangladhish Apartment,
Bharti Nagar, Yavatmal-445001

Email: parthkinkar20@gmail.com

Mobile No: 7020999628

Name: Abhishek Vinod Dhoran

Address: Harikiran Society Malkapur-443101,
Dist. Buldhana-444303

Email: avdhoran11@gmail.com

Mobile No: 9075926454

Name: Ajinkya Jagannath Nagare

Address: At. Pom Deulgaon Raja,
Tq. Deulgaon Raja, Dist. Buldhana- 443204

Email: nagareajinkya53@gmail.com

Mobile No.: 9420043409

Name: Shivani Gopal Digole

Address: At. Po. Shahapur, Tq. Khamgaon,
Dist. Buldhana-444303

Email: shivanidigole@gmail.com

Mobile No: 7083552591