VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi: 590 018

A Mini Project Report

On

"Online integrated platform for projects taken up by the students of various colleges"

Submitted in partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering in Computer Science and Engineering

Submitted by

ABHISHEK PANDEY (1VE21CS004)
ATUL RATHORE (1VE21CS023)

Under the Guidance of

LOKESH M
Assistant Professor
Dept. of CSE,
SVCE, Bengaluru.

Accredited by NAAC & NBA*

SVCE BENGALURU

SRI VENKATESHWARA COLLEGE OF ENGINEERING

— Affiliated to VTU, Approved by AICTE, Recognised by UGC u/s 2(f) & 12(B)—

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRI VENKATESHWARA COLLEGE OF ENGINEERING

Affiliated to VTU Belgaum & Approved by AICTE New Delhi) an ISO 9001:2008 Certified, Kempegowda International Airport Road, Vidyanagar, Bengaluru, Karnataka, India-562157

2023– 2024

SRI VENKATESHWARA COLLEGE OF ENGINEERING

Vidyanagar, Bengaluru, Karnataka, India-562157

Department of Computer Science & Engineering

Accredited by NAAC & NBA*

SVCE BENGALURU

SRI VENKATESHWARA COLLEGE OF ENGINEERING

— Affiliated to VTU, Approved by AICTE, Recognised by UGC u/s 2(f) & 12(B)—

CERTIFICATE

This is to certify that Mini Project entitled **"Online integrated platform for projects taken up by the students of various colleges"** is submitted by **ABHISHEK PANDEY [1VE21CS004] and ATUL RATHORE[1VE21CS023]** on partial fulfillment of sixth semester, Bachelor of Engineering in Computer Science and Engineering, Visvesvaraya Technological University for the academic year 2023-2024.

.....	.....	.....
Signature of Guide	**Signature of Co-Ordinator**	**Signature of the HOD**
LOKESH M Assistant Professor Dept. of CSE, SVCE, Bengaluru.	SURESH P Assistant Professor Dept. of CSE, SVCE, Bengaluru.	DR. HEMA M S Head of Dept. of CSE, SVCE, Bengaluru.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without complementing those who made it possible, whose guidance and encouragement made our efforts successful.

My sincere thanks to highly esteemed institution SRI VENKATESHWARA COLLEGE OF ENGINEERING for grooming up me in to be software engineer.

I express our sincere gratitude to **Dr. Nageswara Guptha**, Principal, SVCE, Bengaluru for providing the required facility.

I am extremely thankful to **Dr. Hema M S**, HOD of CSE, SVCE for providing support and encouragement.

I am extremely thankful to the Co-Ordinator **Mr. SURESH P**, Asst. Professor, Dept. of CSE, SVCE for providing support and encouragement.

I am grateful to **Mr. Lokesh M**, Asst. Professor, Dept. of CSE, SVCE who helped me to complete this project successfully by providing guidance, encouragement and valuable suggestion during entire period of the project. I thank all my computer science staff and others who helped directly or indirectly to meet my project work with grand success.

Finally, I am grateful to my parents and friends for their invaluable support guidance and encouragement.

ABHISHEK PANDEY [1VE21CS004]
ATUL RATHORE [1VE21CS023]

ABSTRACT

This work proposes an online integrated platform designed to centralize student projects from various universities and colleges. This platform, aimed at bridging the gap between project-based learning and digital technology, fosters knowledge sharing and collaboration among students with diverse academic backgrounds. The platform will offer functionalities such as project showcases, discussion forums, and collaborative tools, empowering students to exchange ideas, provide feedback, and co-create knowledge in a virtual environment. This initiative not only promotes peer learning and cross-disciplinary engagement but also serves as a valuable repository of student projects, potentially inspiring future endeavors and fostering innovation within the academic community

Contents

CHAPTER-1

INTROCUCTION

Innovation is the key to betterment of education and students in the Indian universities/colleges put a lot of efforts on the projects as a part of the academic requirements.

If a common knowledge platform is created to bring all project works taken up at various levels by the students in Technical / Higher Educational Institutes and Universities throughout the country,

Then it will be a great source of knowledge and also will help the student community to take up unique/innovative project works.

1.1. BACKGROUND

- A Project operate with teams spread across different locations, this dispersion creates challenges in communication, collaboration and coordination among team's members and others on the same project.
- Managing resources, timeline, and various levels of uncertainty are often more complex for the students, which plays a crucial role for project success.
- Overall, the background behind the development of online integrated project management platforms reflects the need to address the evolving challenges and complexities of managing projects for the students, to enhance efficiency, collaboration, and project success.

1.2. OBJECTIVE

To consolidate all project-related information in a single, accessible location

To facilitate seamless communication and collaboration among team members or different colleges students.

To offer real-time visibility into project status, progress and issues by quickly and effectively.

CHAPTER-2

LITERATURE SURVEY

2.1 Microsoft Teams: -

Microsoft Teams is a powerful chat and collaboration platform that combines various features to enhance communication and productivity within organizations.

The key features are:

- ✓ Channels: are dedicated sections within a team to keep conversations organized by specific topics, projects, disciplines—whatever works for your team. Files that you share in a channel (on the Files tab) are stored in SharePoint.
- ✓ Teams: are collections of people, content, and tools surrounding different projects and outcomes within an organization.
- ✓ Through this anyone can use the Messaging, Meeting and Video conferencing.

2.2 Trello: -

Trello is a visual work management tool that empowers teams to collaborate, ideate, plan, manage, and celebrate their work together.

- ✓ Boards: A Trello board represents a place to keep track of information for projects, teams
- ✓ Lists: Lists keep cards (tasks or pieces of information) organized in different stages of progress
- ✓ Cards: Cards are the smallest units on a board. They represent tasks, ideas, or reminders
- ✓ It Works on Kanban methodology i.e. an agile management method.

2.3 GitHub: -

GitHub is a platform that hosts code, providing version control and collaboration features. It enables you and others to work together on projects from anywhere in the world.

Some key Features:

Repositories: Imagine a central storage for all your project files and their history. This is a GitHub repository. It acts like a container that holds all the different versions of your code, documents, images, or any other project files.

Branches: Think of branches as different lines of development on your project. You can create a new branch from the main codebase to work on a specific feature or bug fix without affecting the main project. Branches allow you to experiment and make changes without messing up the core functionality.

Commits: Whenever you make changes to your code or files, you can take a snapshot of those changes and save them with a message describing what was modified. This snapshot is called a commit. It's like capturing a specific state of your project at a particular point in time.

Pull Requests: Once you're happy with your changes in a branch, you can propose merging them back into the main codebase using a pull request. This creates a request for review from other collaborators. They can review your code, discuss any modifications, and ultimately decide whether to merge your changes into the main branch.

CHAPTER-3

WEBSITE DESIGN

3.1 REQUIRENMENT SPECIFICATIONS

3.1.1 SOFTWARE REQUIREMENTS

- ✓ Operating System: Window 10 or above
- ✓ Operating System: MacOS
- ✓ Google Chrome/Mozilla Firefox/Microsoft Edge
- ✓ Visual Studio Code or any python man editor

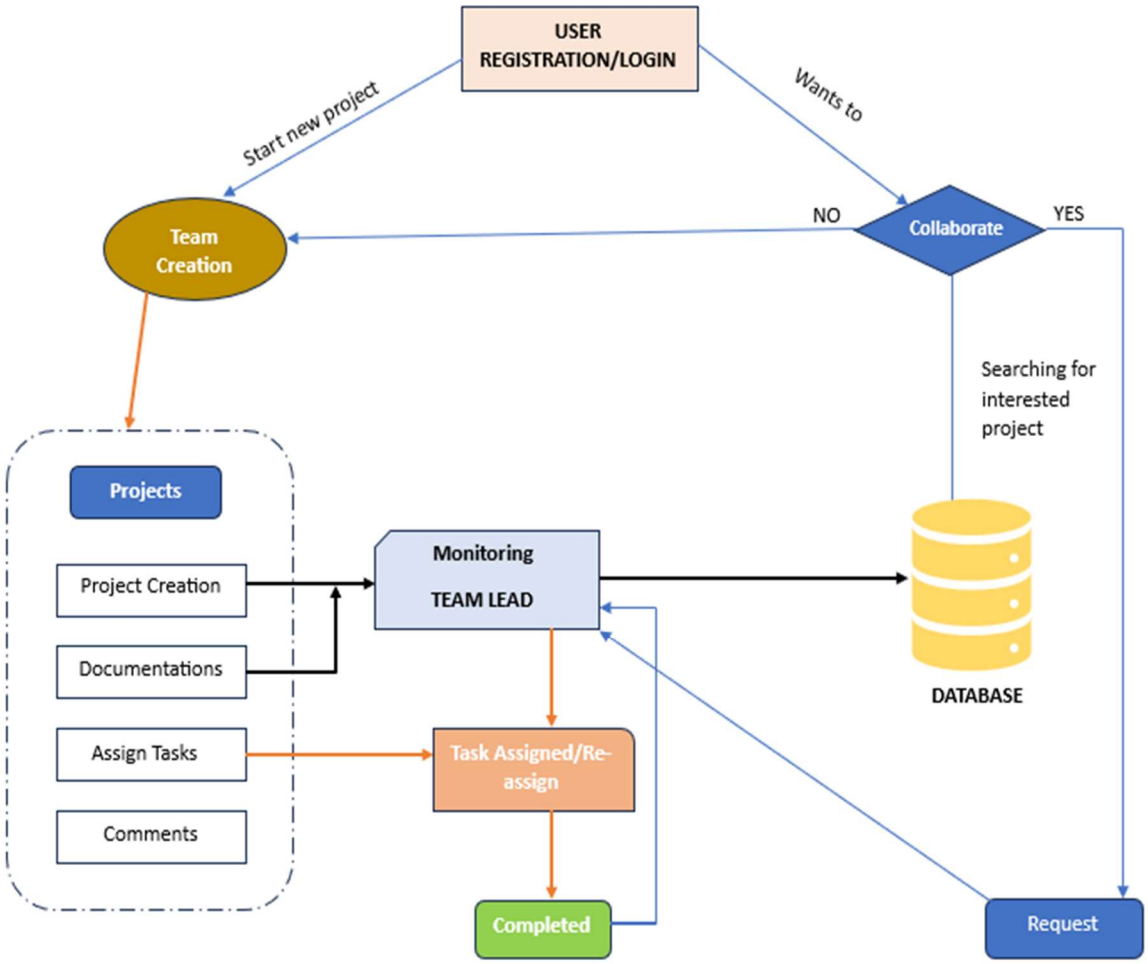
3.1.2 HARDWARE REQUIREMENTS

- ✓ Computer with 1.1 GHz or faster processor
- ✓ Minimum 2GB of RAM or more
- ✓ 2.5 of available hard-disk space
- ✓ 5400 RPM hard drive
- ✓ Higher-resolution display

3.2 CONCEPTUAL DIAGRAM

3.2.1 WORK-ON DIAGRAM

3.2.2 METHODOLOGY DIAGRAM



3.3 IMPLEMENTATION

3.3.1 FRONT END

HTML, **CSS**, and **JavaScript** are the fundamental building blocks of most websites

HTML (Hypertext Markup Language): We can say HTML as the skeleton of your website. It defines the structure and content of a webpage using tags. These tags tell the browser what kind of content to display, like headings, paragraphs, images, and more.

CSS (Cascading Style Sheets): CSS is like the clothing for your website's skeleton (HTML). It controls the visual design of your webpage, including layout, fonts, colors, and animations. With CSS, you can style your website to make it visually appealing and user-friendly.

JavaScript: JavaScript adds interactivity to your website. It's a programming language that can be used to create dynamic effects, respond to user actions, and communicate with servers. This is what makes websites feel more engaging than static webpages.

3.3.2 BACK END

Python: An "implementation" of Python should be taken to mean a program or environment which provides support for the execution of programs written in the Python language, as represented by the Python reference implementation. There have been and are several distinct software packages providing of what we all recognize as Python, although some of those are more like distributions or variants of some existing implementation than a completely new implementation of the language.

3.3.3 DATABASE

SQLite is a lightweight, self-contained, embeddable database engine written in C. Unlike traditional database applications, SQLite doesn't require a separate server process. Instead, it functions as a library that developers can integrate into their applications.

3.3.4 FRAMEWORK

We use Django framework for this mini project.

Key Advantages for using Django:

- ✓ Fast Development**:** Focus on core logic, leaving the framework to handle common web development tasks.
- ✓ Secure by Design**:** Built-in security features mitigate common threats.
- ✓ Scalable Solutions**:** Adapts to accommodate growing projects.
- ✓ Versatile Toolkit**:** Supports a wide range of web application needs.
- ✓ Large Community**:** Extensive documentation, tutorials, and active community support.

CHAPTER-4
SAMPLE CODE

4.1 FRONT END CODE

base.html

```
{% load static %}

<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

  <link rel="manifest" href="{% static 'base/favicons/site.webmanifest' %}">

  <link rel="apple-touch-icon" sizes="180x180" href="{% static
'base/favicons/apple-touch-icon.png' %}">
  <link rel="icon" type="image/png" sizes="32x32" href="{% static
'base/favicons/favicon-32x32.png' %}">
  <link rel="icon" type="image/png" sizes="16x16" href="{% static
'base/favicons/favicon-16x16.png' %}">
  <!-- Bootstrap stylesheet -->
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
integrity="sha384-
9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1
dKGj7Sk" crossorigin="anonymous">

  <!-- Color theme -->
  <!-- <link rel="stylesheet" href="colorTheme.css"> -->
  {% block styles %}
  {% endblock styles %}

  {% block title %}
    <title>National - PMS</title>
  {% endblock title %}
</head>

{% block openBodyTag %}
```

```
<body class="bg-light">
{% endblock openBodyTag %}

  {% include 'base/base_navbar.html' %}

  <!-- Content -->
  {% block content %}
  {% endblock content %}

  {% block footer %}
  {% endblock footer %}

  {% block scripts %}
  {% endblock scripts %}

  <!-- Bootstrap scripts -->
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></
script>
  <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></
script>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

</body>

</html>
```

home.html

```
{% extends 'base/base_background.html' %}

{% load static %}

{% block content %}
<div class="container my-5">
  <div class="row">
    <div class="col-md-6">
      <h1 class="display-4">Welcome to National - Project Management</h1>
```

```
    <p class="lead">Your Ultimate Project Management Solution</p>
    <p>Simplify, Organize, Collaborate</p>
    <a href="{% url 'login' %}" class="btn btn-primary btn-lg">Login</a>
    <a href="{% url 'register' %}" class="btn btn-outline-primary btn-lg ml-
2">Get Started</a>
   </div>
   <div class="col-md-6">
    <img src="{% static 'base/images/project_management.png' %}"
alt="Project Management" class="img-fluid">
   </div>
  </div>
</div>

<div class="container my-5">
  <div class="row">
   <div class="col-md-4">
    <div class="card mb-4">
     <div class="card-body">
      <h2 class="h4">Effortless Project Management</h2>
      <p class="lead">
       Take the complexity out of project management with National - PMS.
Whether you're a solo task manager or part of a team, we've got you covered.
Our user-friendly platform simplifies the process.
      </p>
     </div>
    </div>
   </div>
   <div class="col-md-4">
    <div class="card mb-4">
     <div class="card-body">
      <h2 class="h4">Stay Organized</h2>
      <p class="lead">
       Keep your work meticulously organized with Teams, Projects, and
Tasks. Easily track deadlines, milestones, and project progress to ensure
nothing gets overlooked.
      </p>
     </div>
    </div>
   </div>
   <div class="col-md-4">
```

```
      <div class="card mb-4">
        <div class="card-body">
          <h2 class="h4">Real-time Collaboration</h2>
          <p class="lead">
            Collaborate seamlessly with your team, no matter where they are. Our
real-time communication tools enable you to share updates, messages, and files
instantly, boosting productivity.
          </p>
        </div>
      </div>
    </div>
  </div>
</div>

{% endblock content %}
```

login.html

```
{% extends 'base/base_single_card_sea_bg.html' %}

{% load crispy_forms_tags %}

{% block title %}
  <title>Login | National - PMS</title>
{% endblock title %}



{% block cardContent %}
  <div class="card-header">
    <h2>Login</h2>
  </div>
  <div class="card-body">
    <form method="post" novalidate>
      {% csrf_token %}

      {{ form}}

      <button type="submit" class="btn btn-primary d-block mx-auto">Log
in</button>
    </form>
  </div>
```

```
 <div class="card-footer">
  <small class="d-block">
    Don't have an account? <a href="{% url 'register' %}" class="text-success">
Register an account here!</a> or <a href="{% url 'demo-user-login' %}"
class="text-warning" >login as a demo user</a> to try it out.
  </small>
  <small class="text-muted d-block">
   <a href="{% url 'password_reset' %}" class="text-danger">Forgot Your
Password?</a>
  </small>
 </div>
{% endblock cardContent %}
```

4.1 BACK-END CODE

views.py(user)

```
from django.conf import settings
from django.contrib import messages
from django.contrib.auth import authenticate, login, mixins
from django.contrib.auth.models import User
from django.contrib.messages.views import SuccessMessageMixin
from django.shortcuts import redirect, render, reverse
from django.views import generic
import ast
from collections import Counter
from . import forms
from tasks.models import Task
from projects.models import Project
from teams.models import Team
class UserCreateView(SuccessMessageMixin, generic.CreateView):
    model = User
    form_class = forms.UserRegisterForm
    template_name = "users/user_register.html"
    success_message = "Your account was successfully created."

    def get_success_url(self):
        return reverse("dashboard")

    def form_valid(self, form):
        super().form_valid(form)
        user = self.object
```

```
        user.profile.roll_number = form.cleaned_data.get("roll_number")
        user.profile.college_name = form.cleaned_data.get("college_name")
        user.profile.university_name = form.cleaned_data.get("university_name")
        user.profile.save()
        user = authenticate(
            self.request,
            username=form.cleaned_data["username"],
            password=form.cleaned_data["password1"],
        )
        login(self.request, user)
        return redirect(self.get_success_url())

class UserDetailView(
    mixins.LoginRequiredMixin, generic.DetailView,
):
    model = User
    template_name = "users/user_detail.html"
    slug_url_kwarg = "username"
    slug_field = "username"
    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)
        context["user_profile"] = self.get_object().profile  # Retrieve the user's
profile
        context["college_name"] = self.get_object().profile.college_name
        context["university_name"] = self.get_object().profile.university_name
        return context

class UserUpdateView(
    mixins.LoginRequiredMixin, mixins.UserPassesTestMixin,
generic.UpdateView
):
    model = User
    form_class = forms.UserChangeForm
    template_name = "users/user_update.html"

    def get_object(self, queryset=None):
        user_pk = self.request.user.pk
        return User.objects.get(pk=user_pk)

    def get_success_url(self):
```

```
        return reverse("profile", args=[self.request.user.username])

    def test_func(self):
        return not self.request.user.profile.is_demo_user

class UserDeleteView(
    mixins.LoginRequiredMixin, mixins.UserPassesTestMixin,
generic.DeleteView
):
    model = User
    template_name = "users/user_confirm_delete.html"

    def get_object(self, queryset=None):
        user_pk = self.request.user.pk
        return User.objects.get(pk=user_pk)

    def get_success_url(self):
        return reverse("home")

    def test_func(self):
        return not self.request.user.profile.is_demo_user

def demo_user_login_view(request):
    if request.method == "POST":
        user = authenticate(
            request,
            username=settings.DEMO_USER_USERNAME,
            password=settings.DEMO_USER_PASSWORD,
        )

        if user is not None:
            if user.profile.is_demo_user:
                login(request, user)
                return redirect(settings.LOGIN_REDIRECT_URL)
            else:
                msg = (
                    "The user you have provided is not a demo user. Please "
                    + "set a different demo user in your settings or change "
                    + "the current user to a demo user."
                )
```

```
            raise PermissionError(msg)

        messages.error(request, "Demo user account not setup, contact admin to
setup a demo account")

    return render(request, template_name="users/login_demo_user.html")

class DashboardView(mixins.LoginRequiredMixin, generic.TemplateView):
    template_name = "users/user_dashboard.html"

    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)

        # Fetch all projects
        projects = Project.objects.all()
        #print(projects)
    # Extract and count technologies used
        technology_counts = Counter()
        project_completed_counts = 0
        project_active_counts = 0
    # Iterate through projects and count technologies
        for project in projects:
            technologies_used_str = project.technologies_used
            project_status = project.completed

            #print("Technologies used for project {}: {}".format(project.id,
technologies_used))
            if technologies_used_str is not None:
                #print(f"Technologies Used for project {project.id}:
{technologies_used_str}")
                technologies_used_list = ast.literal_eval(technologies_used_str)
                for tech in technologies_used_list:
                    technology_counts[tech] = technology_counts.get(tech, 0) + 1
            if project_status is not None:
                if project_status:
                    project_completed_counts+=1
                else:
                    project_active_counts+=1

        #print(project_completed_counts)
```

```
        # print(project_active_counts)
        #print(" project {}: {} {}".format(project.id, technologies_used_list,
technology_counts))
    # Prepare the data for the chart
        chart_labels =list(technology_counts.keys())

        chart_data = list(technology_counts.values())

        context["chart_labels"] = chart_labels
        context["chart_data"] = chart_data
        context["project_completed_counts"] = project_completed_counts
        context["project_active_counts"] = project_active_counts

        context["task_list"] = (
            Task.objects.filter(completed=False)
            .filter(assigned_to=self.request.user)
            .order_by("date_due", "-priority_level")[:7]
        )

        context["unassigned_task_list"] = (
            Task.objects.filter(completed=False)
            .filter(team__leader=self.request.user)
            .filter(assigned_to=None)
            .order_by("date_due", "-priority_level")[:7]
        )


        context["project_list"] = (
            Project.objects.filter(completed=False)
            .filter(team__members=self.request.user)
            .order_by("date_due")
        )

        context["team_list"] = Team.objects.filter(members=self.request.user)

        return context

from django.contrib.auth import logout

def logout_view(request):
```

```
    logout(request)
    # Redirect to your desired page after logout
    return render(request,'users/logout.html')
```

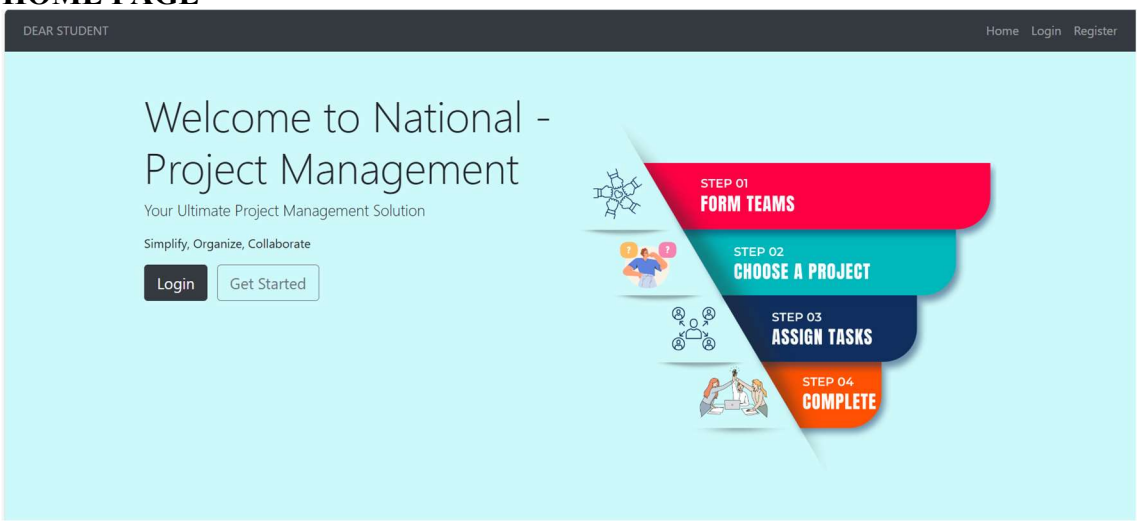
models.py(user)

```
    from django.db import models
from django.contrib.auth.models import User
from projects.models import Project

class Profile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    roll_number = models.CharField(max_length=50, blank=True, null=True)
    college_name = models.CharField(max_length=100, blank=True, null=True)
    university_name = models.CharField(max_length=100, blank=True,
null=True)
    is_manager = models.BooleanField(default=False)
    is_demo_user = models.BooleanField(default=False)

    def __str__(self):
        return self.user.username

    def get_connections(self):
        """A queryset of users who share a team with the user object."""
        q = User.objects.filter(pk=self.user.pk)
        for team in self.user.team_set.all():
            q = q | team.members.all()
        return q.distinct()

    def get_related_projects(self):
        q = Project.objects.none()
        for team in self.user.team_set.all():
            q = q | team.project_set.all()
        return q.distinct()
```

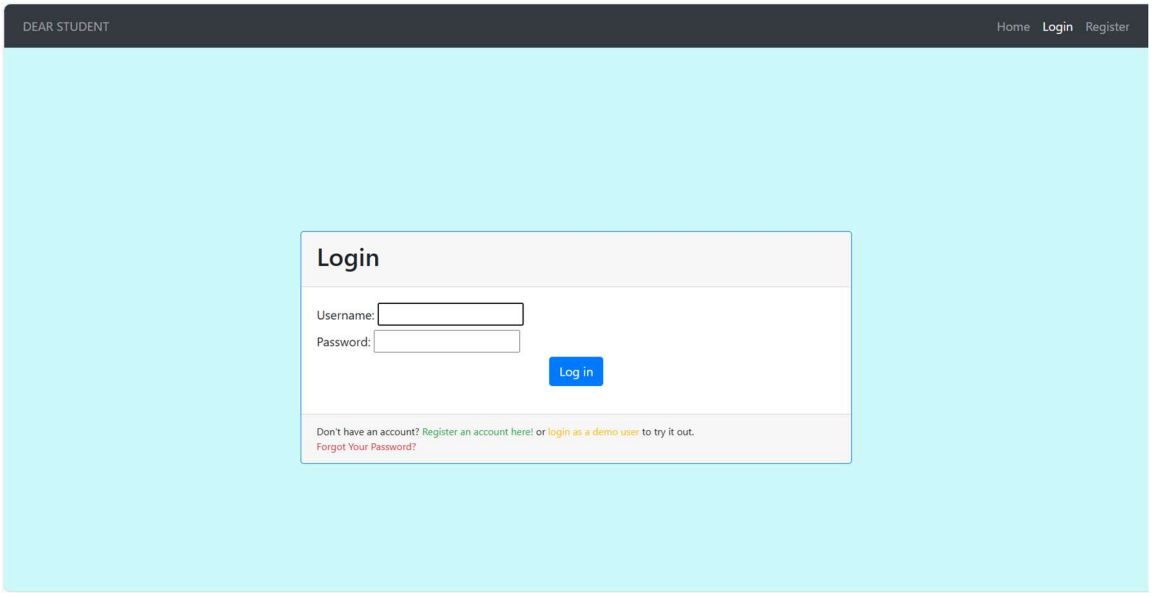
CHAPTER-5
USER INTERFACE

5.1 SCREEN SHOTS

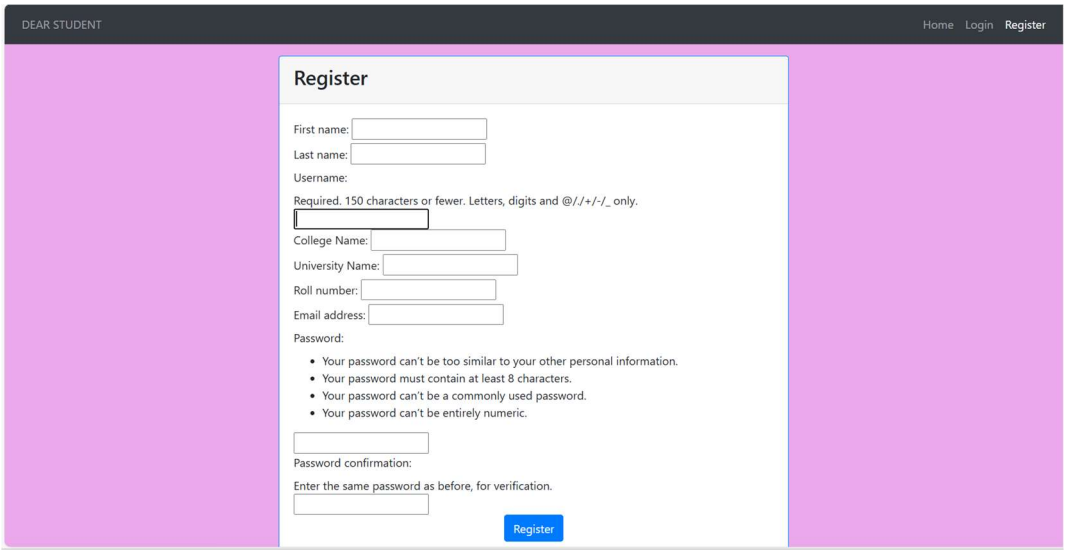
HOME PAGE



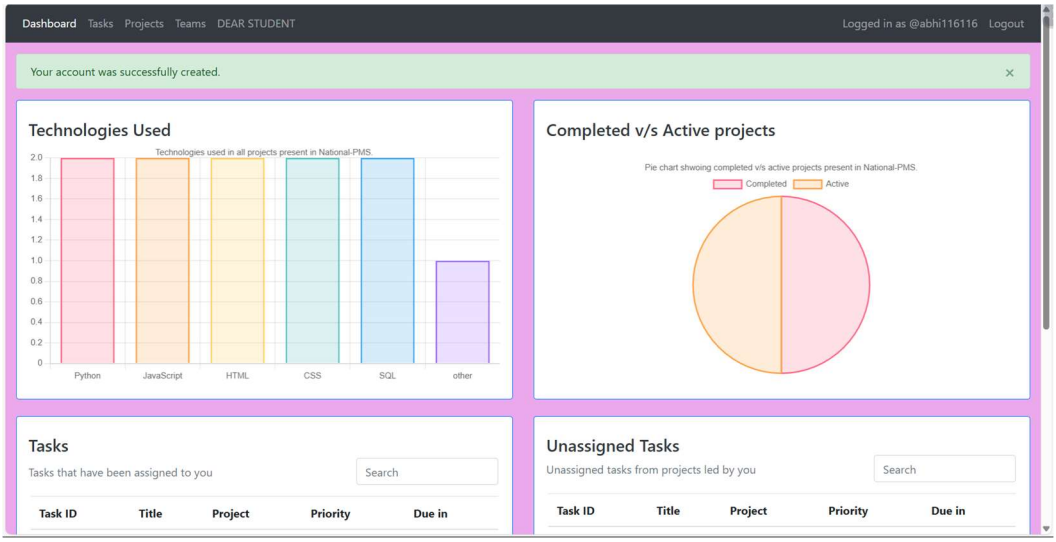
LOGIN PAGE



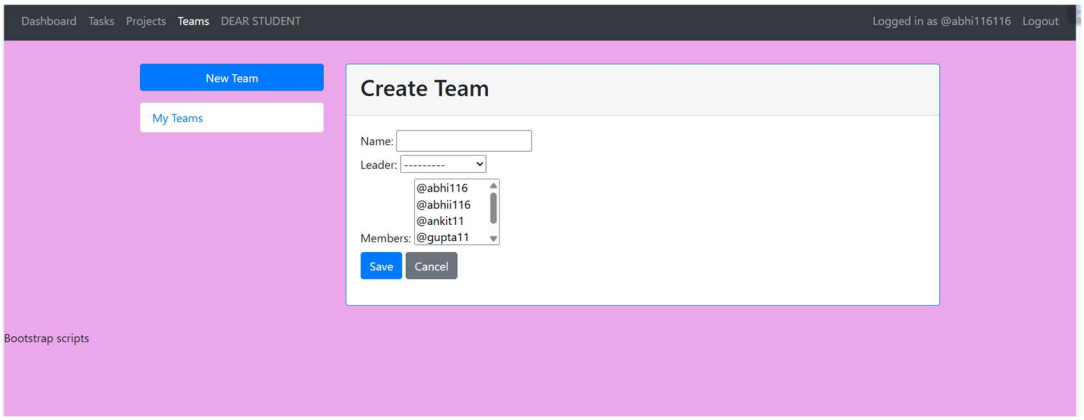
REGISTER PAGE



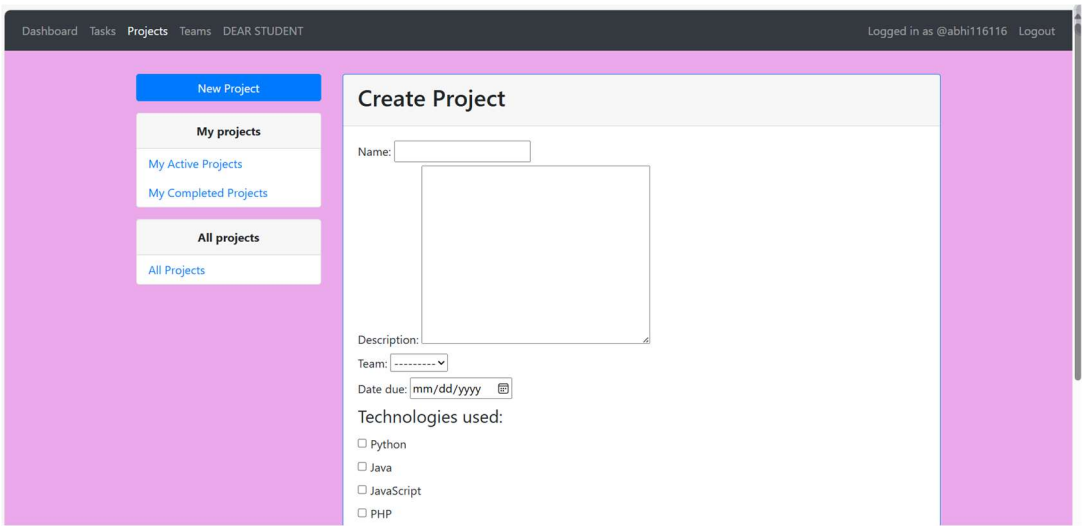
DASHBOARD



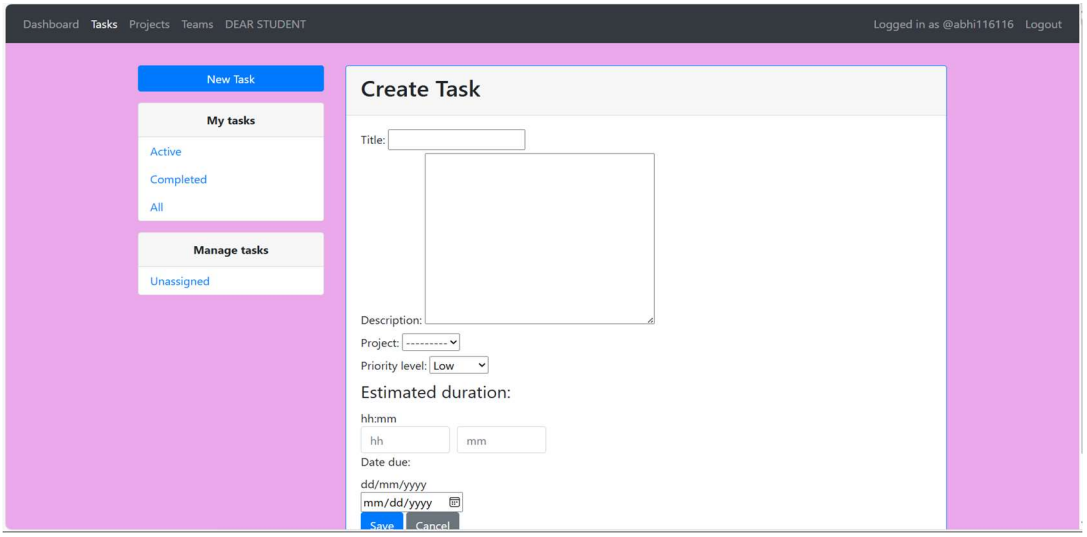
TEAM CREATION



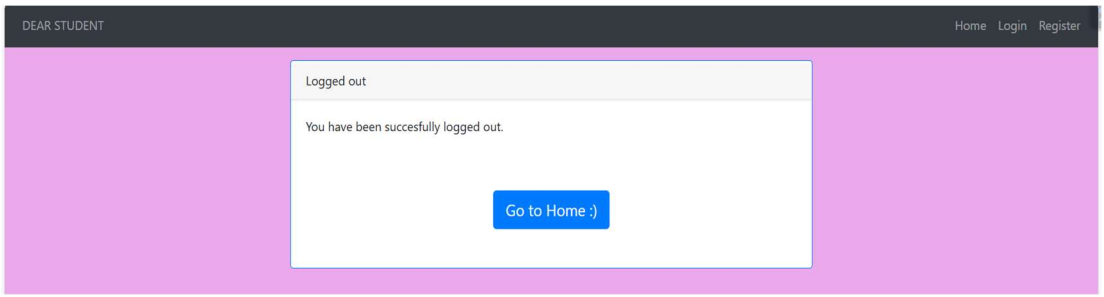
PROJECT CREATION



TASKS ASSIGN



LOGOUT



CHAPTER -6

CONCLUSION

An online integrated platform for student projects across various colleges offers a multitude of benefits for students, faculty, and even the educational landscape as a whole. Students gain access to a vast repository of project ideas, resources, and potential collaborators. Faculty can leverage the platform to identify trends, encourage inter-institutional collaboration, and potentially discover new areas for research. On a broader scale, such a platform fosters knowledge sharing, promotes innovation, and strengthens the overall educational ecosystem.

CHAPTER - 7

FUTURE SCOPE & ENHANCEMENT

Our online project platform is a great start, but there's always room to grow! Here are some ideas to make it even more awesome in the future:

Gamification: Imagine earning points and badges for completing project milestones or collaborating with others. This could make project work more fun and engaging!

AI Power: An AI assistant could suggest relevant resources, connect students with compatible collaborators, or even provide feedback on project ideas.

Virtual Reality (VR): Imagine exploring project concepts or conducting experiments in a VR environment! This could make learning more immersive and interactive.

Industry Linkages: Partnering with companies could allow students to work on real-world projects and gain valuable industry experience.

Global Collaboration: Break down geographical barriers! Features for translation and communication could enable students from around the world to work together on projects.

CHAPTER - 8

REFERENCES

- Django-documentation
- Stackoverflow.com
- Docs.python.org
- Wscube Tech Channel & Website
- Bootstrap.com
- https://www.researchgate.net/publication/356633631_AN_APPRAISAL_OF_MICROSOFT_TEAMS_AND_ITS_FEATURES

DrillBit

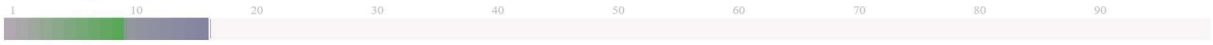
The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

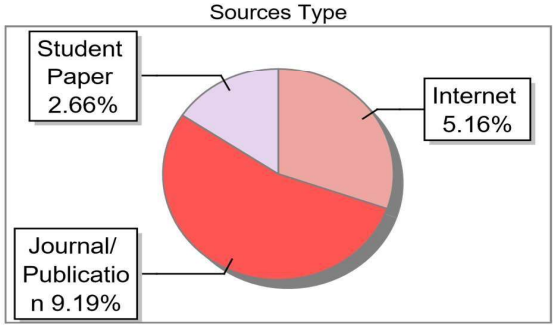
Author Name	Abhishek Pandey
Title	Online integrated platform for projects taken up by the students of various colleges"
Paper/Submission ID	2187630
Submitted by	virupaksha.msc12@gmail.com
Submission Date	2024-07-31 13:44:30
Total Pages, Total Words	26, 3084
Document type	Project Work

Result Information

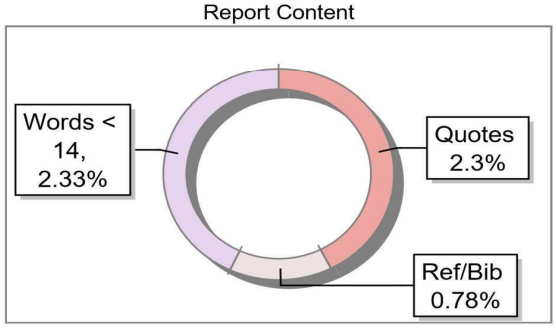
Similarity **17 %**



Sources Type



Report Content



Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Source: Excluded < 14 Words	Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

17	9	B
SIMILARITY %	MATCHED SOURCES	GRADE

A-Satisfactory (0-10%)
B-Upgrade (11-40%)
C-Poor (41-60%)
D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	proceeding.conferenceworld.in	5	Publication
2	ciie.uok.edu.in	4	Publication
3	sih.gov.in	2	Internet Data
4	Submitted to Visvesvaraya Technological University, Belagavi	2	Student Paper
5	sih.gov.in	1	Internet Data
6	sih.gov.in	1	Internet Data
7	Submitted to Visvesvaraya Technological University, Belagavi	1	Student Paper
8	Thesis Submitted to Shodhganga, shodhganga.inflibnet.ac.in	1	Publication
9	avxlive.icu	1	Internet Data