

# **Student Technical Skill Development Planner Based On Student Feedback**

## **Using Latest Trends & Techniques.**

Community Service Project Report Submitted to the Faculty of Engineering of

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA, KAKINADA**

In partial fulfilment of the requirements for the award of the Degree of

**BACHELOR OF TECHNOLOGY**

IN

**INFORMATION TECHNOLOGY**



**By :**

**Maganti Sai Veni**

(23481A12D9)

**Panthangi Gopi chand**

(23481A12H8)

**Laddika Nithusha**

(23481A12D3)

**Padamata Navadeep**

(23481A12H0)

Under the Supervision of

**Dr .M V L N Raja Rao,**M. Tech., Ph.D.,

Professor.

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**SESHADRI RAO GUDLAVALLERU ENGINEERING COLLEGE**

(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)

SESHADRIRAO KNOWLEDGE VILLAGE, GUDLAVALLERU – 521356

**ANDHRA PRADESH**

**2025-2026**

# **Program Book**

## **for**

# **Community Service Project**



**Name of the Student:** Maganti Sai Veni

**Name of the College:** Seshadri Rao Gudlavalleru Engineering College

**Registration Number:** 23481A12D9

**Period of CSP:** **From:** 19-05-2025 **To:** 30-06-2025 and

**From:** 14-07-2025 **To:** 26-07-2025

**Name & Address of the Community/Habitation:** Seshadri Rao Gudlavalleru Enginnering College, Gudlavalleru, Gudlavalleru Mandal, Krishna District, Andhra Pradesh 521356.

# **Community Service Project Report**

Submitted in accordance with the requirement for the degree of Bachelor of Technology

**Name of the College:** Seshadri Rao Gudlavalleru Engineering College

**Department:** Information Technology

**Name of the Faculty Guide:** Dr. M. V. L. N. Raja Rao, M. Tech., (Ph. D)

**Duration of the CSP:** From : 19-05-2025 To: 30-06-2025

From: 14-07-2025 To: 26-07-2025

**Name of the Student:** Maganti Sai Veni

**Programme of Study:** Bachelor of Technology.

**Year of Study:** III B. Tech, I Semester.

**Register Number:** 23481A12D9

**Date of Submission:**

## **Student Declaration**

We, Maganti Sai Veni, Reg. No. 23481A12D9 student of CSP Program of the Department of Information Technology, Seshadri Rao Gudlavalleru Engineering College do here by declare that I have completed the mandatory community service from 19-05-2025 to 30-06-2025 and from 14-07-2025 to 26-07-2025 about Community Services Growth and Recognition in Gudlavalleru, Krishna district of Andhra Pradesh under the Faculty Guideship of **Dr. M. V. L. N. Raja Rao**, Professor. Department of **Information Technology** in **Seshadri Rao Gudlavalleru Engineering College, Gudlavalleru.**

(Signature and Date)

## **Endorsements**

Faculty Guide: **Dr. M. V. L. N. Raja Rao**

Master of Trainer(S): **Mr Md. Rafi Khan**

Head of the Department: **Dr.D.N.V.S.L.S. Indira**

Principal: **Dr. B. Karuna Kumar**

## **Certificate from official of the community**

This is to certify that Maganti Sai Veni, Reg. No.23481A12D9 of Seshadri Rao Gudlavalleru Engineering College underwent community service project in Gudlavalleru, Krishna District of Andhra Pradesh From 19-05-2025 To 30-06-2025 and 14-07-2025 To 26-07-2025. The overall performance of the community service volunteer during her community service is found to be satisfactory.

**Authorized Signatory with Date and Seal**

## **ACKNOWLEDGEMENTS**

We are deeply indebted to several individuals whose guidance, encouragement, and support made this project a success.

First and foremost, we express our heartfelt gratitude to **Dr. M. V. L. N. Raja Rao**, Professor, Department of Information Technology, for his invaluable guidance, supervision, and motivation throughout the project.

We also extend our sincere thanks to **Dr. D. N. V. S. L. S. Indira**, Head of the Department, Information Technology, for her enthusiastic encouragement and support and helpful feedback during the analysis. Her valuable suggestions and advice were very important in improving the project.

We are grateful to our beloved Principal, **Dr. B. Karuna Kumar**, for providing a great support for us in completing our project and giving us the opportunity for doing project.

We are truly thankful to **Mr. Ch.Venkateswara Rao**, CSP Coordinator, for his constant support, clear guidance, and encouragement, which helped us complete every part of the project smoothly.

Our special thanks to the both teaching and Non-Teaching Staff of our department. Thanks to our family members, and friends who directly or indirectly contributed to the timely completion of this project.

### **Project Associates:**

M. Sai Veni (23481A12D9)

## **ABSTRACT**

Student Technical Skill Development Planner Based on Latest Trends & Techniques is a web-based application designed to assist academic departments in effectively planning, managing, and evaluating Student Development Programs (SDPs). The system maintains a centralized database of previous programs, capturing essential details such as topics covered, student participation, feedback, resource persons, and outcomes. This data-driven approach enables departments to make informed decisions and improve the quality and relevance of future SDPs.

By adopting this feedback-based and analytics-driven system, institutions can enhance student engagement, ensure transparency, and maintain continuous improvement in skill development initiatives. Furthermore, the system contributes to bridging the gap between academic learning and industry requirements, thereby preparing students for both academic excellence and professional competence.

Overall, the Student Technical Skill Development Planner promotes a collaborative, efficient, and systematic approach to managing technical training programs, fostering a culture of lifelong learning and innovation among students.

# INDEX

<b>CONTENTS</b>	<b>Page</b>
<b>Chapter 1:</b> Executive Summary	1
<b>Chapter 2:</b> Overview of the Community	2
<b>Chapter 3:</b> Community Service Part	4
<b>Chapter 4:</b> Weekly Report	6
<b>Chapter 5:</b> Outcomes Description	22
Questionnaire	22
Problems identified in the Community	24
Short-term and Long-term action plan for possible outcomes	25
Describe the problem you identified	26
Report	27
Code for website	28
User Interface	51
<b>Chapter 6:</b> Recommendations and Conclusions	53
Student Self-Evaluation	54
Evaluation by the person in-charge	55
Community Service Project Survey Photos	57
PO and CO mapping	64

# **CHAPTER 1: EXECUTIVE SUMMARY**

## **Website Overview:**

The Student Technical Skill Development Planner is a web-based platform designed to manage and evaluate Student Development Programs (SDPs).

It helps departments plan programs based on the latest technologies and student feedback. The website provides separate login access for students and faculty.

Students can enroll in programs, give feedback, and vote for new topics.

Faculty can create, update, and monitor SDPs through a user-friendly dashboard.

All program data and feedback are stored securely in a centralized database.

The system analyzes feedback to identify trending technologies and student interests.

It ensures transparency, efficiency, and continuous improvement in SDP planning.

The project bridges the gap between academic learning and industry requirements.

Overall, it enhances student engagement and promotes skill-based education.

## **Learning Objectives:**

- To identify and apply the latest technological trends that enhance students' technical skills.
- To design and develop a web-based system for planning and managing Student Development Programs (SDPs).
- To integrate student feedback for improving the quality and relevance of SDPs.
- To enhance coordination between students and faculty through a centralized digital platform.
- To apply programming and database knowledge in developing a real-time application.
- To promote continuous learning and align academic activities with industry needs.

## **Learning Outcomes:**

- Gained practical experience in web design, development, and database integration.
- Improved ability to analyze student feedback and identify trending technologies.
- Applied theoretical knowledge to develop a functional, real-time web application.

## **CHAPTER 2: OVERVIEW OF THE COMMUNITY**

**Community Overview:** Gudlavalleru, Krishna District, Andhra Pradesh

### **Historical Profile:**

Gudlavalleru is a small village in Krishna District, Andhra Pradesh, with a rich history dating back to the Kakatiya dynasty. The village was once a major center for agriculture and trade.

### **Community Diversity:**

The community comprises predominantly Telugu-speaking people, with a mix of various castes and religions, including Hindus, Muslims, and Christians.

### **Traditions, Ethics, and Values:**

1. Strong agricultural heritage
2. Respect for elders and traditional customs
3. Importance of education and social service
4. Vibrant festivals like Sankranti, Ugadi, and Vinayaka Chavithi
5. Emphasis on community bonding and collective welfare

### **Socio-Economic Conditions:**

1. Primarily agrarian economy with rice, cotton, and vegetable cultivation
2. Moderate literacy rate (approx. 70%)
3. Limited job opportunities, leading to migration to urban areas
4. Access to basic amenities like healthcare, education, and transportation
5. Growing awareness of digital technologies and online services

### **Demographics:**

1. Population: Approximately 5,000-7,000
2. Age distribution: 25% children (0-14), 55% adults (15-59), 20% elderly (60+)
3. Sex ratio: The sex ratio is 1,013 females per 1,000 males
4. Caste distribution: OC (50%), BC (30%), SC (15%), ST (5%)

**Education:**

1. Literacy rate: 70% (male: 75%, female: 65%)
2. Schools: Government schools (AANM and VVRSSR Gudlavalleru High School) and private institutions
3. Higher education: Nearby towns like Vijayawada and Machilipatnam offer college options

**Healthcare:**

1. Primary Health Center (PHC) in the village
2. Access to nearby hospitals in Vijayawada and Machilipatnam
3. Limited specialized medical services

**Infrastructure:**

1. Well-connected roads to nearby towns
2. Regular bus services
3. Basic amenities like electricity, water supply, and sanitation

## **CHAPTER 3: COMMUNITY SERVICE PART**

Community Service Part: Empowering Education in Gudlavalleru

### **Project Overview:**

As part of our community service project, we aimed to bridge the educational divide in Gudlavalleru, Krishna District, Andhra Pradesh. We conducted a comprehensive survey to identify the community's needs and developed a tailored intervention.

### **Activities Undertaken:**

- **Survey and Data Collection:** Conducted surveys among students to identify interests, skill gaps, and preferred SDP topics.
- **Website Design and Development:** Developed a web-based platform to plan, manage, and evaluate Student Development Programs efficiently.
- **Module Implementation:** Created separate login modules for students and faculty with topic voting, feedback, and enrollment features.
- **Database and Backend Integration:** Integrated a secure Supabase database for data storage, authentication, and smooth backend connectivity.
- **Testing and Evaluation:** Tested the website on multiple devices to ensure functionality, responsiveness, and user-friendly performance.

### **Impact:**

- Increased student participation and engagement in technical skill development programs.
- Improved coordination between students and faculty through a centralized online platform.
- Enabled data-driven planning of SDPs based on real-time feedback and analytics..
- Bridged the gap between academic learning and industry requirements for better student growth.

### **Reflection:**

Through this project, I acquired:

### **Values:**

- 1.Empathy: Understanding users' learning needs and challenges.
- 2.Collaboration: Recognizing the importance of sharing knowledge and skills.
- 3.Responsibility: secure, reliable, and user-friendly platform features.

**Life Skills:**

- 1.Communication: Interacting effectively through chat and feedback systems.
- 2.Problem-Solving: Handling technical issues, real-time messaging, and user experience challenges.
- 3.Time Management: Organizing tasks and managing development milestones efficiently

**Technical Skills:**

- 1.Web Development: Gained hands-on experience in designing and developing responsive web pages using HTML, CSS, and JavaScript.
- 2.Database Management: Learned to create and manage databases using Supabase and SQL for secure data storage and retrieval.
- 3.Real-Time Communication: Enabled real-time communication between students and faculty through instant feedback and topic suggestion features on the platform.
- 4.Authentication & Security: Implemented secure user authentication using Supabase with encrypted login and role-based access control.

This experience has instilled in me a sense of purpose and responsibility towards community development. I am committed to continuing this work and inspiring others to join the effort.

## **ACTIVITY LOG FOR THE FIRST WEEK**

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
19-05-2025 Monday			
20-05-2025 Tuesday			
21-05-2025 Wednesday			
22-05-2025 Thursday			
23-05-2025 Friday			
24-05-2025 Saturday			

## **WEEKLY REPORT**

**WEEK-1(From DT.19-05-2025 to DT.24-05-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## **ACTIVITY LOG FOR THE SECOND WEEK**

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
<b>26-05-2025</b> <b>Monday</b>			
<b>27-05-2025</b> <b>Tuesday</b>			
<b>28-05-2025</b> <b>Wednesday</b>			
<b>29-05-2025</b> <b>Thursday</b>			
<b>30-05-2025</b> <b>Friday</b>			
<b>31-05-2025</b> <b>Saturday</b>			

## **WEEKLY REPORT**

**WEEK–2(From Dt.26-05-2025 to Dt 31-05-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## **ACTIVITY LOG FOR THE THIRD WEEK**

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
<b>02-06-2025</b> Monday			
<b>03-06-2025</b> Tuesday			
<b>04-06-2025</b> Wednesday			
<b>05-06-2025</b> Thursday			
<b>06-06-2025</b> Friday			
<b>07-06-2025</b> Saturday			

## **WEEKLY REPORT**

**WEEK-3(From Dt.02-06-2025 to Dt. 07-06-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## ACTIVITY LOG FOR THE FOURTH WEEK

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
09-06-2025 Monday			
10-06-2025 Tuesday			
11-06-2025 Wednesday			
12-06-2025 Thursday			
13-06-2025 Friday			
14-06-2025 Saturday			

## **WEEKLY REPORT**

**WEEK–4(From DT.09-06-2025 to DT.14-06-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## **ACTIVITY LOG FOR THE FIFTH WEEK**

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
16-06-2025 Monday			
17-06-2025 Tuesday			
18-06-2025 Wednesday			
19-06-2025 Thursday			
20-06-2025 Friday			
21-06-2025 Saturday			

## **WEEKLY REPORT**

**WEEK-5(From DT.16-06-2025 to DT.21-06-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## ACTIVITY LOG FOR THE SIXTH WEEK

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
23-06-2025 Monday			
24-06-2025 Tuesday			
25-06-2025 Wednesday			
26-06-2025 Thursday			
27-06-2025 Friday			
28-06-2025 Saturday			

## **WEEKLY REPORT**

**WEEK–6(FromDt.23-06-2025 toDt.28-06-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## **ACTIVITY LOG FOR THE SEVENTH WEEK**

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
14-07-2025 Monday			
15-07-2025 Tuesday			
16-07-2025 Wednesday			
17-07-2025 Thursday			
18-07-2025 Friday			
19-07-2025 Saturday			

## **WEEKLY REPORT**

**WEEK-7(From DT.14-07-2025 to DT.19-07-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## ACTIVITY LOG FOR THE EIGHT WEEK

<b>DAY &amp; DATE</b>	<b>BRIEF DESCRIPTION OF THE DAILY ACTIVITY</b>	<b>LEARNING OUTCOME</b>	<b>Person In-charge Signature</b>
<b>21-07-2025</b> <b>Monday</b>			
<b>22-07-2025</b> <b>Tuesday</b>			
<b>23-07-2025</b> <b>Wednesday</b>			
<b>24-07-2025</b> <b>Thursday</b>			
<b>25-07-2025</b> <b>Friday</b>			
<b>26-07-2025</b> <b>Saturday</b>			

## **WEEKLY REPORT**

**WEEK–8(From DT.21-07-2025 to DT.26-07-2025)**

**Objective of the Activity Done:**

**Detailed Report:**

## **CHAPTER5: OUTCOMES DESCRIPTION**

**Details of the Socio-Economic Survey of the Village/Habitation. Attach the questionnaire prepared for the survey.**

1. What is the main purpose of the web-based system?
2. What type of data does the system store?
3. Which department is this system intended for?
4. How confident are you that this system will continue to improve future Skill Development Programs?
5. What is one of the key benefits of using this system?
6. How easy is the platform to navigate?
7. How would you rate the clarity of information displayed on the platform?
8. How well does the system integrate with other academic tools or platforms you use (e.g., LMS, email, or student portal)?
9. How user-friendly do you find the interface of the system?
10. Would you like to see additional features such as AI-based skill recommendations or personalized learning paths?
11. Has the system improved your awareness of available SDPs?
12. Do you think the system helps align programs with industry trends?
13. How effectively does the platform support continuous learning?
14. Has this platform increased your participation in SDPs?
15. Do you believe this system helps improve academic outcomes?
16. How transparent do you find the SDP evaluation process through this system?
17. How useful is student feedback in improving future SDPs?
18. How often do you provide feedback or suggestions on SDPs?
19. Do you feel your feedback is considered when planning new SDPs?
20. Does the platform encourage you to be more involved in department activities?
21. How satisfied are you overall with the system's performance?

22. What feature do you find most useful in the system?
23. What improvements would you suggest for the system?
24. Would you recommend this platform to other departments or institutions
25. How likely are you to continue using this platform for future SDPs?

## **Describe the problems you have identified in the community:**

### **Skill Gaps and Learning Challenges**

1. Lack of awareness about the latest technologies and required industry skills among students..
2. Absence of a feedback-based system for planning effective Student Development Programs (SDPs).
3. Limited access to digital learning resources and technical guidance..
4. Communication gaps between students and faculty in organizing and managing skill development activities..

### **Networking and Collaboration Constraints**

1. Limited student-faculty collaboration in planning programs.
2. Poor communication for sharing ideas and feedback.
3. No unified platform for teamwork and knowledge sharing.
4. Limited collaboration between students and faculty in skill development activities.

### **Resource and Accessibility Limitations**

1. Inadequate access to digital learning tools and online resources.
2. Limited availability of technical infrastructure like computers and internet.
3. Lack of updated study materials and software for practical learning.
4. Unequal access to learning opportunities among students.

### **Technological and Communication Challenges**

1. Limited knowledge of advanced web and database technologies.
2. Difficulty in maintaining proper communication between students and faculty.
3. Poor internet connectivity affecting online learning and collaboration.
4. Lack of technical support for resolving software or platform issues.

### **Time and Scheduling Constraints**

1. Difficulty in scheduling SDPs effectively.
2. Overlapping academic and program timings.
3. Limited time for planning and execution.

**Short term and long-term action plan for possible solutions for the problems identified and that could be recommended to the concerned educational authorities for implementation.**

**1. Awareness Campaign:**

Conducted a campaign to promote technical skill development awareness. Demonstrated the use of the Skill Development Planner platform. Improved student participation and interest in new technologies.

**2. Platform Launch:**

To introduce an interactive system for managing, planning, and improving Student Development Programs efficiently.

**3. Partnerships and Collaboration:**

Built collaboration between students, faculty, and the department to plan SDPs. Encouraged teamwork and improved communication for better program outcomes..

**4. Content and Skill Accessibility:**

Developed a web platform to provide easy access to learning resources and SDPs. Ensured students can explore skill-based content anytime to enhance technical knowledge..

**Long-Term Action Plan:**

**1. Skill Development Programs:**

Organized skill development programs to enhance students' technical and practical knowledge . Focused on trending technologies, hands-on learning, and industry-relevant skills for career growth..

**2. Platform Maintenance and Enhancement:**

Regularly updated the platform to improve performance, security, and user experience. Added new features and resources based on student feedback and technological advancements.

**3. Performance Tracking and Analytics:**

Implemented tools to monitor student participation and feedback in SDPs. Analyzed data to evaluate program effectiveness and plan future improvements.

## **Description of the Community awareness programme/conducted**

**w.r.t the problems and their outcomes.**

### **Description of the Community Awareness Program**

The community awareness program aimed to educate students on technical skill Development and digital learning. It demonstrated the use of the Student Skill Development Planner for accessing programs and giving feedback. The program increased awareness, participation, and interest in continuous learning.

#### **Key Problems Addressed:**

- 1.Limited Access to Skill Sharing Opportunities: The community awareness program aimed to educate students on technical skill development and digital learning.
- 2.Lack of Networking and Collaboration Platforms: Lack of networking and collaboration platforms restricted student interaction and knowledge exchange.
- 3.Need for real time communication: Need for real-time communication to enable instant feedback, updates, and collaboration between students and faculty.

#### **Program Outcomes:**

- 1.Increased Skill Accessibility: Increased skill accessibility by providing students with easy online access to learning resources and development programs.
2. Enhanced Networking and Collaboration: Enhanced networking and collaboration by connecting students and faculty through a unified digital platform.
3. Empowered Learning Community: Empowered learning community by encouraging active participation, knowledge sharing, and continuous skill improvement among students.

Through these efforts, the program has successfully promoted a culture of collaborative learning and empowered users to both teach and learn, making skill development more accessible and engaging

## **Reported of the community service project work done in the related subject w.r.t the habitation/village .**

### **Key Problems in Skill Development Opportunities**

Students face several challenges in accessing effective skill development opportunities. There is limited awareness of the latest technologies and industry-relevant skills, which affects their career readiness. Access to quality learning resources and practical training remains inadequate, making it difficult for students to enhance their technical abilities. The absence of structured platforms for feedback and collaboration further limits active participation. Additionally, irregular conduct of Student Development Programs reduces continuous learning and overall skill growth.

### **Analysis of Skill Exchange Barriers**

Students face barriers in sharing knowledge due to limited communication and collaboration platforms. Unequal access to digital tools restricts participation in group learning. Lack of mentorship and networking opportunities hinders continuous skill exchange.

### **Proposed Solution: Student Technical Skill Development Planner Based On Student Feedback Using Latest Trends and Techniques**

To address these challenges, the project proposes a web-based Student Technical Skill Development Planner Based On Student Feedback Using Latest Trends and Techniques that connects with faculty and students. The platform enables users to:

1. Create a web platform to manage and share skill development programs.
2. Add real-time feedback, collaboration, and topic suggestion features.
3. Ensure equal access to digital learning resources for all students.

By centralizing skill exchange in a user-friendly, interactive platform, individuals can seamlessly teach, learn, and collaborate.

## **Code for Website:**

### **Admin.html:**

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Admin Dashboard - SkillQuest</title>
<link href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700&display=swap" rel="stylesheet">
<style>
* {
margin: 0;
padding: 0;
box-sizing: border-box;
}
body {
font-family: 'Inter', sans-serif;
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
position: relative; color: white;
line-height: 1.6; min-height: 100vh; font-size: 18px;
animation: explosiveEntry 1.2s ease-out;
}
@keyframes explosiveEntry { 0% {
transform: scale(0) rotate(180deg); opacity: 0;
filter: blur(20px);
} 50% {
transform: scale(1.2) rotate(0deg); opacity:
filter: blur(5px);
}
100% {
transform: scale(1) rotate(0deg); opacity: 1;
filter: blur(0px);
}
}
body::before { content: ""; position: absolute; top: 0;
left: 0;
right: 0;
bottom: 0;
background: rgba(0, 0, 0, 0.5);
z-index: -1;
}
* {
color: #1a1a1a !important; font-size: 18px !important;
}
.container {
```

```
max-width: 1200px; margin: 0 auto; padding: 20px;
}
.header { display: flex;
justify-content: space-between; align-items: center;
padding: 20px 0;
border-bottom: 2px solid #e2e8f0; margin-bottom: 30px;
}
.welcome {
font-size: 1.8rem; font-weight: 600; color: #1e293b;
}
.nav-buttons { display: flex; gap: 15px;
}
.btn {
padding: 12px 24px; border: none; border-radius: 8px;
background: #6b7280; color: white;
cursor: pointer; transition: all 0.3s ease; text-decoration: none; font-weight: 500;
}
.btn:hover {
background: #4b5563; transform: translateY(-1px);
}
.btn-primary { background: #6b7280; color: white;
}
.dashboard { display: grid;
grid-template-columns: repeat(auto-fit, minmax(350px, 1fr)); gap: 40px;
margin: 50px 0;
padding: 20px 0;
}
.content-grid { display: grid;
grid-template-columns: 2fr 1fr; gap: 30px;
margin-bottom: 30px;
}
.stat-card {
background: rgba(255, 255, 255, 0.2); backdrop-filter: blur(15px);
border: 2px solid rgba(255, 255, 255, 0.4); border-radius: 25px;
padding: 60px 40px; text-align: center;
box-shadow: 0 15px 50px rgba(0,0,0,0.5); min-height: 220px;
display: flex;
flex-direction: column; justify-content: center; transform: scale(1);
transition: transform 0.3s ease;
}
.stat-card:hover { transform: scale(1.05);
}
.stat-number { font-size: 5rem;
font-weight: 800; color: white;
margin-bottom: 15px;
text-shadow: 3px 3px 6px rgba(0,0,0,0.7); line-height: 1;
```

```
}

.stat-label { color: white;
font-weight: 700; font-size: 1.3rem;
text-shadow: 2px 2px 4px rgba(0,0,0,0.7);
text-transform: uppercase; letter-spacing: 1px;
}

.section {
background: rgba(255, 255, 255, 0.15); backdrop-filter: blur(10px);
border: 1px solid rgba(255, 255, 255, 0.3); border-radius: 12px;
padding: 25px;
box-shadow: 0 4px 20px rgba(0,0,0,0.3); animation: rainDrop 1.5s ease-out;
}

.section:nth-child(1) { animation-delay: 0.1s; }
.section:nth-child(2) { animation-delay: 0.3s; }
.section:nth-child(3) { animation-delay: 0.5s; }
.section:nth-child(4) { animation-delay: 0.7s; }
.section:nth-child(5) { animation-delay: 0.9s; }

@keyframes rainDrop { 0% {
transform: translateY(-150vh) scale(0.3); opacity: 0;
} 70% {
transform: translateY(0) scale(1.1); opacity: 1;
} 85% {
transform: translateY(-10px) scale(0.95);
} 100% {
transform: translateY(0) scale(1); opacity: 1;
}
}

.section h2 {
margin-bottom: 20px; font-size: 1.3rem;
color: white;
font-weight: 600;
}

.section-header { display: flex;
justify-content: space-between; align-items: center;
margin-bottom: 20px;
}

.section h2 {
font-size: 1.3rem; margin: 0;
}

.programs-grid { display: grid;
grid-template-columns: repeat(auto-fit, minmax(300px, 1fr)); gap: 20px;
}

.program-card {
background: rgba(255, 255, 255, 0.1); backdrop-filter: blur(8px);
border: 1px solid rgba(255, 255, 255, 0.2); border-radius: 8px;
padding: 20px; margin-bottom: 15px;
transition: all 0.3s ease;
```

```
}

.program-card:hover {
background: rgba(255,255,255,0.2); transform: translateY(-2px);
}

.program-title { font-weight: 600;
margin-bottom: 8px; color: white;
}

.program-meta { font-size: 0.9rem; opacity: 0.8;
margin-bottom: 15px;
}

.program-actions { display: flex; gap: 10px;
flex-wrap: wrap;
}

.btn-small {
padding: 8px 16px; font-size: 0.9rem; border-radius: 20px; border: none;
cursor: pointer;
transition: all 0.3s ease;
}

.btn-edit { background: #4ecdc4; color: white; }

.btn-view { background: #45b7d1; color: white; }

.btn-delete { background: #ff6b6b; color: white; }

.two-column { display: grid;
grid-template-columns: 1fr 1fr; gap: 30px;
}

.vote-item, .feedback-item { display: flex;
justify-content: space-between; align-items: center;
padding: 15px;
background: rgba(255,255,255,0.1); backdrop-filter: blur(8px);
border: 1px solid rgba(255, 255, 255, 0.2);
border-radius: 10px; margin-bottom: 10px;
}

.form-grid { display: grid;
grid-template-columns: 1fr 1fr 1fr; gap: 15px;
margin-bottom: 15px;
}

.form-input, .form-select { padding: 12px;
border: 1px solid rgba(255, 255, 255, 0.2); border-radius: 6px;
background: rgba(255, 255, 255, 0.1); backdrop-filter: blur(8px);
color: black !important; font-family: inherit;
}

.form-input::placeholder {
color: rgba(0, 0, 0, 0.7) !important;
}

.form-input:focus, .form-select:focus { outline: none;
border-color: #3b82f6;
box-shadow: 0 0 0 3px rgba(59, 130, 246, 0.1);
}
```

```

.modal { display: none;
position: fixed; top: 0;
left: 0;
width: 100%;
height: 100%;
background: rgba(0,0,0,0.5); z-index: 1000;
}
.modal-content { position: absolute; top: 50%;
left: 50%;
transform: translate(-50%, -50%);
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
padding: 30px; border-radius: 20px; width: 90%;
max-width: 600px; color: white;
}
.modal-header { display: flex;
justify-content: space-between; align-items: center;
margin-bottom: 20px;
}
.close {
background: none; border: none; color: white;
font-size: 1.5rem; cursor: pointer;
}
@media (max-width: 768px) {
.two-column {
grid-template-columns: 1fr;
}
.form-grid {
grid-template-columns: 1fr;
}
.programs-grid {
grid-template-columns: 1fr;
}
}
</style>
</head>
<body>
<div class="container">
<header class="header">
<h1 class="welcome" id="greeting">Admin Dashboard</h1>
<div class="nav-buttons">
<button id="btn-analytics" class="btn" style="padding: 8px 16px; font-size: 0.9rem;">Analytics<img alt="Settings icon" style="vertical-align: middle;"/> SettingsLogout

```

```
<div class="section">
<div class="section-header">
<h2>'-' IH_µ_ µ- Programs Management</h2>
<div>
<select id="filter-status" class="form-select" style="margin-right: 15px;">
<option value="all">All Programs</option>
<option value="upcoming">Upcoming</option>
<option value="ongoing">Ongoing</option>
<option value="completed">Completed</option>
</select>
<button id="btn-add" class="btn btn-primary">Add New Program</button>
</div>
</div>
<div id="list-programs"></div>
</div>
<div class="section">
<div class="section-header">
<h2>••.^•ç Upcoming Programs</h2>
<button id="btn-toggle-voting" class="btn">Close Requests</button>
</div>
<div id="list-votes"></div>
<form id="form-topic" style="margin-top: 15px;">
<div class="form-grid">
<input id="topic-title" placeholder="Program title" class="form-input" style="grid-column: 1 / 3;">
<button type="submit" class="btn">Add Program</button>
</div>
</form>
</div>
</div>
<div class="section">
<h2>★ Recent Feedback</h2>
<div id="list-feedback"></div>
</div>
<div class="section">
<div class="section-header">
<h2>.•9,,` Student Suggestions</h2>
<span id="suggest-count" style="opacity: 0.8;"></span>
</div>
<div id="list-suggest"></div>
</div>
<section class="dashboard" style="margin-top: 40px;">
<div class="stat-card">
<div class="stat-number" id="stat-programs">0</div>
<div class="stat-label">Total Programs</div>
</div>
<div class="stat-card">
```

```

<div class="stat-number" id="stat-registrations">0</div>
<div class="stat-label">Total Registrations</div>
</div>
<div class="stat-card">
<div class="stat-number" id="stat-feedback">0</div>
<div class="stat-label">Feedback Entries</div>
</div>
</section>
</div>
<!-- Program Modal -->
<div id="program-modal" class="modal">
<div class="modal-content">
<div class="modal-header">
<h3>Program Details</h3>
<button class="close" onclick="closeModal()">&times;</button>
</div>
<form id="form-program">
<input type="hidden" id="program-id">
<div class="form-grid">
<input id="title" placeholder="Program Title" class="form-input" required>
<input id="category" placeholder="Category" class="form-input" required>
<select id="status" class="form-select">
<option value="upcoming">Upcoming</option>
<option value="ongoing">Ongoing</option>
<option value="completed">Completed</option>
</select>
</div>
<textarea id="description" placeholder="Description" class="form-input" rows="3" style="width: 100%; margin-bottom: 15px;"></textarea>
<div class="form-grid">
<input id="capacity" type="number" placeholder="Capacity" class="form-input" value="30">
<button type="submit" class="btn btn-primary">Save Program</button>
<button type="button" class="btn" onclick="closeModal()">Cancel</button>
</div>
</form>
</div>
</div>
<!-- Analytics Modal -->
<div id="analytics-modal" class="modal">
<div class="modal-content" style="max-width: 900px; max-height: 80vh; overflow-y: auto;">
<div class="modal-header">
<h3>Analytics Dashboard</h3>
<button class="close" onclick="closeAnalyticsModal()">&times;</button>
</div>
<div id="analytics-content">

```

```

<div style="display: grid; grid-template-columns: repeat(auto-fit, minmax(200px, 1fr)); gap: 20px; margin-bottom: 30px;">
  <div class="stat-card" style="background: linear-gradient(135deg, #667eea 0%, #764ba2 100%)">
    <div class="stat-number" id="analytics-total-programs">0</div>
    <div class="stat-label">Total Programs</div>
  </div>
  <div class="stat-card" style="background: linear-gradient(135deg, #f093fb 0%, #f5576c 100%)">
    <div class="stat-number" id="analytics-total-participants">0</div>
    <div class="stat-label">Total Participants</div>
  </div>
  <div class="stat-card" style="background: linear-gradient(135deg, #4facfe 0%, #00f2fe 100%)">
    <div class="stat-number" id="analytics-avg-rating">0</div>
    <div class="stat-label">Average Rating</div>
  </div>
</div>
<div style="background: rgba(255,255,255,0.1); backdrop-filter: blur(10px); border-radius: 15px; padding: 25px; margin-bottom: 20px;">
  <h4 style="color: white; margin-bottom: 20px; text-align: center;">□#/Program Participation Chart</h4>
  <div id="participation-chart" style="height: 200px; position: relative;"></div>
</div>
<div style="background: rgba(255,255,255,0.1); backdrop-filter: blur(10px); border-radius: 15px; padding: 25px;">
  <h4 style="color: white; margin-bottom: 20px; text-align: center;">★ Rating Distribution</h4>
  <div id="rating-chart" style="height: 150px; position: relative;"></div>
</div>
</div>
</div>
</div>
<script src="https://unpkg.com/@supabase/supabase-js@2"></script>
<script src=".js/store.js"></script>
<script>
if (!Store.requireRole('admin')) { window.location.href = 'auth-faculty.html'; }
const elPrograms = document.getElementById('list-programs'); const elVotes = document.getElementById('list-votes');
const elFeedback = document.getElementById('list-feedback'); const btnAdd = document.getElementById('btn-add');
const btnLogout = document.getElementById('btn-logout'); const filterStatus = document.getElementById('filter-status');
async function refreshStats() { try {
  if (Store.useSupabase()) {
    const programs = await Store.listProgramsSupabase(); const feedback = await

```

```

Store.listFeedbackSupabase();
const totalRegistrations = programs.reduce((sum, p) => sum +
(p.participants?.length || 0), 0);
document.getElementById('stat-programs').textContent = programs.length;
document.getElementById('stat-registrations').textContent =
totalRegistrations;
document.getElementById('stat-feedback').textContent = feedback.length;
} else {
const s = Store.stats();
document.getElementById('stat-programs').textContent = s.programs;
document.getElementById('stat-registrations').textContent = s.registrations;
document.getElementById('stat-feedback').textContent = s.feedback;
}
} catch (error) {
console.error('Error refreshing stats:', error); const s = Store.stats();
document.getElementById('stat-programs').textContent = s.programs;
document.getElementById('stat-registrations').textContent = s.registrations;
document.getElementById('stat-feedback').textContent = s.feedback;
}
}

async function renderPrograms() { const status = filterStatus.value; let items;
try {
if (Store.useSupabase()) {
items = await Store.listProgramsSupabase(status === 'all' ? undefined : status);
} else {
items = Store.listPrograms(status === 'all' ? undefined :
status);
}
} catch (error) {
console.error('Error loading programs:', error); items = [];
}
elPrograms.innerHTML = '';
if (!items || items.length === 0) {
elPrograms.innerHTML = '<div style="text-align: center; opacity: 0.7; padding: 40px;">No programs found. Create your first program!</div>';
return;
}
items.forEach(p => {
const participantCount = p.participants ? p.participants.length :
0;
const card = document.createElement('div'); card.className = 'program-card';
card.innerHTML =
<div class="program-title">${p.title}</div>
<div class="program-meta">${p.category} • ${p.status} •
${participantCount}/${p.capacity} enrolled</div>
<div class="program-actions">
<button data-id="${p.id}" class="btn-small btn-edit">Edit</button>

```

```

<button data-id="${p.id}" class="btn-small btn-view">View Registrations</button>
<button data-id="${p.id}" class="btn-small btn-download" style="background: #8b5cf6; color: white;">Download List</button>
<button data-id="${p.id}" class="btn-small btn-delete">Delete</button>
</div>
`; elPrograms.appendChild(card);
});
}
async function renderVotes() { try {
const items = Store.useSupabase()
? await Store.listVotesSupabase()
: Store.listVotes(); elVotes.innerHTML = "";
if (!items || items.length === 0) {
elVotes.innerHTML = '<div style="text-align: center; opacity: 0.7; padding: 20px;">No program requests yet. Add one below!</div>';
return;
}
items.forEach(v => {
const row = document.createElement('div'); row.className = 'vote-item';
row.innerHTML = `<div>
<div style="font-weight: 500;">${v.topic_title}</div>
<div style="font-size: 0.8rem; opacity: 0.8;">${v.category} • ${v.votes} requests</div>
</div>
<button data-id="${v.id}" class="btn-small btn-delete" onclick="deleteVote('${v.id}')">Delete</button>
`; elVotes.appendChild(row);
});
} catch (error) {
console.error('Error loading votes:', error); elVotes.innerHTML = '<div style="text-align: center; opacity: 0.7; padding: 20px;">Error loading votes.</div>';
}
}
async function renderFeedback() { try {
const allFeedback = Store.useSupabase()
? await Store.listFeedbackSupabase()
: Store.listFeedback();
const programs = Store.useSupabase()
? await Store.listProgramsSupabase()
: Store.listPrograms(); elFeedback.innerHTML = "";
if (!allFeedback || allFeedback.length === 0) { elFeedback.innerHTML = '<div style="text-align: center; opacity: 0.7; padding: 20px;">No feedback yet.</div>'; return;
}
}

```

```

// Group feedback by program const feedbackByProgram = {};
allFeedback.forEach(f => {
const programId = f.program_id;
if (!feedbackByProgram[programId]) { feedbackByProgram[programId] = [];
}
feedbackByProgram[programId].push(f);
});
// Display feedback grouped by program
Object.keys(feedbackByProgram).forEach(programId => {
const program = programs.find(p => p.id === programId); const programTitle =
program ? program.title : 'Unknown
Program';
const feedbacks = feedbackByProgram[programId];
const programSection = document.createElement('div');
programSection.style.marginBottom = '20px'; programSection.innerHTML =
`

#### ${programTitle} (${feedbacks.length} feedback)


`;
feedbacks.slice(0, 3).forEach(f => {
const card = document.createElement('div'); card.className = 'feedback-item';
card.style.marginBottom = '8px'; card.innerHTML =
<div>
<div style="font-weight: 500;">${f.student_name || f.student_roll}</div>
<div style="font-size: 0.9rem;">${'★'.repeat(f.rating)} ${f.comment}</div>
</div>
`; programSection.appendChild(card);
});
elFeedback.appendChild(programSection);
});
} catch (error) {
console.error('Error loading feedback:', error); elFeedback.innerHTML = '<div
style="text-align: center; opacity:
0.7; padding: 20px;">Error loading feedback.</div>';
}
}
|| ";
// Modal functions
function openModal(program = null) { document.getElementById('program-
id').value = program?.id || ""; document.getElementById('title').value =
program?.title || ""; document.getElementById('category').value =
program?.category
document.getElementById('description').value =
program?.description || "";
document.getElementById('status').value = program?.status || 'upcoming';
document.getElementById('capacity').value = program?.capacity
|| 30;
}

```

```

document.getElementById('program-modal').style.display =
'block';
}
function closeModal() {
document.getElementById('program-modal').style.display = 'none';
}
function closeAnalyticsModal() { document.getElementById('analytics-
modal').style.display =
'none';
}
async function showAnalytics() { try {
let programs, feedback;
if (Store.useSupabase()) {
programs = await Store.listProgramsSupabase(); feedback = await
Store.listFeedbackSupabase();
} else {
programs = Store.listPrograms(); feedback = Store.listFeedback();
}
const totalPrograms = programs.length;
const totalParticipants = programs.reduce((sum, p) => sum +
(p.participants?.length || 0), 0);
const avgRating = feedback.length > 0 ? (feedback.reduce((sum, f) => sum +
f.rating, 0) / feedback.length).toFixed(1) : '0';
document.getElementById('analytics-total-      programs').textContent =
totalPrograms;
document.getElementById('analytics-total-      participants').textContent =
totalParticipants;
document.getElementById('analytics-avg-rating').textContent = avgRating;
// Create participation chart createParticipationChart(programs);
// Create rating chart createRatingChart(feedback);
document.getElementById('analytics-modal').style.display =
'block';
} catch (error) {
console.error('Error loading analytics:', error); alert('Error loading analytics
data');
}
}
function createParticipationChart(programs) {
const chartContainer = document.getElementById('participation- chart');
chartContainer.innerHTML = '';
const maxParticipants = Math.max(...programs.map(p =>
p.participants?.length || 0), 1);
programs.slice(0, 6).forEach((program, index) => {
const participantCount = program.participants?.length || 0; const percentage =
(participantCount / maxParticipants) * 100;
const bar = document.createElement('div'); bar.style.cssText = `display: flex;

```

```

align-items: center; margin-bottom: 15px; color: white;
`;
bar.innerHTML = `
<div style="width: 120px; font-size: 0.9rem; margin-right: 15px;">${program.title.substring(0, 15)}...</div>
<div style="flex: 1; background: rgba(255,255,255,0.2); border-radius: 10px; height: 20px; position: relative; overflow: hidden;">
<div style="background: linear-gradient(90deg, #4facfe 0%, #00f2fe 100%); height: 100%; width: ${percentage}%; border-radius: 10px; transition: width 0.5s ease;"></div>
</div>
<div style="width: 50px; text-align: right; font-size: 0.9rem; margin-left: 10px;">${participantCount}</div>
`;
chartContainer.appendChild(bar);
});
}

function createRatingChart(feedback) {
const chartContainer = document.getElementById('rating-chart');
chartContainer.innerHTML = '';
const ratingCounts = [0, 0, 0, 0, 0]; feedback.forEach(f => {
if (f.rating >= 1 && f.rating <= 5) { ratingCounts[f.rating - 1]++; }
});
const maxCount = Math.max(...ratingCounts, 1);
for (let i = 0; i < 5; i++) {
const percentage = (ratingCounts[i] / maxCount) * 100; const stars = '★'.repeat(i + 1);
const bar = document.createElement('div'); bar.style.cssText = `
display: flex;
align-items: center; margin-bottom: 10px; color: white;
`;
bar.innerHTML = `
<div style="width: 80px; font-size: 0.9rem; margin-right: 15px;">${stars}</div>
<div style="flex: 1; background: rgba(255,255,255,0.2); border-radius: 8px; height: 15px; position: relative; overflow: hidden;">
<div style="background: linear-gradient(90deg, #f093fb 0%, #f5576c 100%); height: 100%; width: ${percentage}%; border-radius: 8px; transition: width 0.5s ease;"></div>
</div>
<div style="width: 40px; text-align: right; font-size: 0.9rem; margin-left: 10px;">${ratingCounts[i]}</div>
`;
chartContainer.appendChild(bar);
}
}

```

```

// Event listeners
btnAdd.addEventListener('click', () => openModal());
btnLogout.addEventListener('click', () => {
  Store.logout(); window.location.href = 'index.html';
});
document.getElementById('btn-analytics').addEventListener('click', async () => {
  await showAnalytics();
});
document.getElementById('btn-settings').addEventListener('click', () => {
  const current = prompt('Enter current access code:');
  const cfg = Store.getConfig();
  if (current !== (cfg.adminCode || cfg.facultyCode)) { alert('Invalid current code'); }
  return;
})
const newCode = prompt('Enter new access code (min 6 characters):');
if (newCode && newCode.length >= 6) { Store.setAdminCode(newCode);
  alert('Access code updated successfully!'); }
});
filterStatus.addEventListener('change', renderPrograms);
document.getElementById('form-program').addEventListener('submit', async (e) => {
  e.preventDefault();
  const programId = document.getElementById('program-id').value.trim();
  const payload = {
    title: document.getElementById('title').value.trim(),
    category: document.getElementById('category').value.trim(),
    description: document.getElementById('description').value.trim(),
    status: document.getElementById('status').value,
    capacity: parseInt(document.getElementById('capacity').value || '30', 10)
  };
  if (programId) payload.id = programId; if (!payload.title) {
    alert('Program title is required'); return;
  }
  try {
    console.log('Saving program:', payload);
    if (Store.useSupabase()) { console.log('Using Supabase');
      const result = await Store.saveProgramSupabase(payload);
      console.log('Supabase result:', result);
    } else {
      console.log('Using localStorage'); Store.saveProgram(payload);
    }
    closeModal();
    await refreshStats(); await renderPrograms();
  }
});

```

```

alert('Program      saved      successfully!');      console.log('Program      saved
successfully');
} catch (error) {
console.error('Error saving program:', error); alert('Failed to save program: ' +
error.message);
}
});
elPrograms.addEventListener('click', async (e) => { const t = e.target;
const id = t.getAttribute('data-id');
if (t.classList.contains('btn-edit')) { let p;
if (Store.useSupabase()) {
const programs = await Store.listProgramsSupabase(); p = programs.find(prog
=> prog.id === id);
} else {
p = Store.getProgram(id);
}
openModal(p);
}
if (t.classList.contains('btn-delete')) { if (confirm('Delete this program?')) {
if (Store.useSupabase()) {
await Store.deleteProgramSupabase(id);
} else {
Store.deleteProgram(id);
}
await renderPrograms(); await refreshStats();
}
}
}
if (t.classList.contains('btn-view')) { let p;
if (Store.useSupabase()) {
const programs = await Store.listProgramsSupabase(); p = programs.find(prog
=> prog.id === id);
} else {
p = Store.getProgram(id);
}
const participants = p.participants || [];
alert(`Roll Numbers for "${p.title}":\n\n${participants.join('\n')}` +
`'No registrations yet'`);
}
if (t.classList.contains('btn-download')) { let p;
if (Store.useSupabase()) {
const programs = await Store.listProgramsSupabase(); p = programs.find(prog
=> prog.id === id);
} else {
p = Store.getProgram(id);
}
const participants = p.participants || [];
const csv = 'Roll Number\n' + participants.map(s =>

```

```

`"${s}"`).join("\n");
const blob = new Blob([csv], { type: 'text/csv' });
const url = URL.createObjectURL(blob);
const a = document.createElement('a');
a.href = url;
a.download = `${p.title.replace(/\^a-zA-Z]/gi, '_')}_registrations.csv`;
a.click();
URL.revokeObjectURL(url);
}
});
document.getElementById('form-topic').addEventListener('submit', async (e) => {
e.preventDefault();
const title = document.getElementById('topic-title').value.trim();
if (!title) {
alert('Program title is required');
return;
}
try {
console.log('Adding vote topic:', title);
await Store.addVoteTopic({ title, category: 'general' });
document.getElementById('topic-title').value = '';
await renderVotes();
console.log('Vote topic added successfully');
} catch (error) {
console.error('Error adding vote topic:', error);
alert('Failed to add program request: ' + error.message);
}
});
const btnToggleVoting = document.getElementById('btn-toggle-voting');
btnToggleVoting.addEventListener('click', () => {
Store.setVotingOpen(!Store.isVotingOpen());
btnToggleVoting.textContent =
Store.isVotingOpen() ? 'Close Requests' : 'Open Requests';
});
// Vote deletion function
async function deleteVote(id) {
if (confirm('Delete this program request?')) {
await Store.deleteVote(id);
await renderVotes();
}
}
// Initialize
async function init() {
await refreshStats();
await renderPrograms();
await renderVotes();
await renderFeedback();
btnToggleVoting.textContent =
Store.isVotingOpen() ? 'Close Requests' : 'Open Requests';
}
async function renderSuggestions() {
try {
const items = await Store.useSupabase();
await Store.listSuggestionsSupabase();
const wrap = document.getElementById('list-suggest');
const count = document.getElementById('suggest-count');

```

```

wrap.innerHTML = '';
count.textContent = `${items.length} suggestions`;
if (!items || items.length === 0) {
  wrap.innerHTML = '<div style="text-align: center; opacity: 0.7; padding: 20px;">No suggestions yet.</div>';
  return;
}
items.slice(0, 10).forEach(s => {
  const card = document.createElement('div'); card.className = 'feedback-item';
  card.innerHTML = `
<div>
<div style="font-weight: 500;">${s.student_roll || s.roll}</div>
<div style="font-size: 0.9rem;">${s.text}</div>
</div>
`;
  wrap.appendChild(card);
});
} catch (error) {
  console.error('Error loading suggestions:', error);
}
}

init(); renderSuggestions();
</script>
</body>
</html>

```

## **Help.html:**

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Help - Student Development Programs</title>
<link href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700&display=swap" rel="stylesheet">
<style>
* {
margin: 0;
padding: 0;
box-sizing: border-box;
}
body {
font-family: 'Inter', sans-serif;
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
color: white; line-height: 1.6;
min-height: 100vh; padding: 20px;
}

```

```
}

.container {
  max-width: 800px; margin: 0 auto;
  background: rgba(255, 255, 255, 0.1); backdrop-filter: blur(15px);
  border: 1px solid rgba(255, 255, 255, 0.2); border-radius: 20px;
  padding: 40px;
}
.header {
  text-align: center; margin-bottom: 40px;
}
.title {
  font-size: 2.5rem; font-weight: 700; margin-bottom: 10px;
}
.subtitle {
  font-size: 1.1rem; opacity: 0.9;
}
.section {
  margin-bottom: 30px;
}
.section-title {
  font-size: 1.5rem; font-weight: 600; margin-bottom: 15px; color: #fbbf24;
}
.content {
  font-size: 1rem; margin-bottom: 15px;
}
.feature-list {
  list-style: none; padding-left: 0;
}
.feature-list li {
  margin-bottom: 8px; padding-left: 20px; position: relative;
}
.feature-list li::before {
  content: "•"; color: #fbbf24; font-weight: bold; position: absolute; left: 0;
}
.back-btn {
  display: inline-block; padding: 16px 32px;
  background: linear-gradient(135deg, #10b981 0%, #059669
  100%);
  color: white;
  text-decoration: none; border-radius: 50px; font-weight: 600; margin-top: 30px;
  transition: all 0.3s ease;
  box-shadow: 0 4px 12px rgba(16, 185, 129, 0.3); font-size: 1.1rem;
}
.back-btn:hover {
  transform: translateY(-3px);
  box-shadow: 0 8px 25px rgba(16, 185, 129, 0.4);
}
```

```
</style>
</head>
<body>
<div class="container">
<div class="header">
<div class="title">‘ _H_1μ’ – μ Help Guide</div>
<div class="subtitle">Student Development Programs</div>
</div>
<div class="section">
<div class="section-title">* ◆For Students</div>
<div class="content">
<ul class="feature-list">
<li>Click 'Student Portal' to register or login with your roll number</li>
<li>Browse available programs and enroll in ones you're interested in</li>
<li>Give feedback and ratings after completing programs</li>
<li>Vote on upcoming program topics</li>
<li>Submit suggestions for new programs</li>
</ul>
</div>
</div>
<div class="section">
<div class="section-title">•~-V•T^ -•• “-’ For Faculty</div>
<div class="content">
<ul class="feature-list">
<li>Click 'Faculty Dashboard' and use access code:  
IT2025</li>
<li>Create, edit, and manage student development programs</li>
<li>View student registrations and export participant lists</li>
<li>Monitor feedback and program analytics</li>
<li>Manage voting topics and review student suggestions</li>
<li>Change system settings and access codes</li>
</ul>
</div>
</div>
<div class="section">
<div class="section-title"> Technical Features</div>
<div class="content">
<ul class="feature-list">
<li>Works with Supabase database or offline with localStorage</li>
<li>Real-time data synchronization across browser tabs</li>
<li>Secure authentication and role-based access control</li>
<li>Responsive design works on all devices</li>
<li>Data is automatically saved and backed up</li>
</ul>
</div>
</div>
```

```

<div class="section">
<div class="section-title"> -`' Why Use This System?</div>
<div class="content">
<ul class="feature-list">
<li><strong>Streamlined Management:</strong> Centralized platform for all student development activities</li>
<li><strong>Easy Enrollment:</strong> Students can browse and enroll in programs with just a few clicks</li>
<li><strong>Real-time Feedback:</strong> Instant feedback collection helps improve program quality</li>
<li><strong>Data-Driven Decisions:</strong> Analytics help faculty make informed decisions about programs</li>
<li><strong>Student Voice:</strong> Voting and suggestion features ensure student needs are heard</li>
<li><strong>Efficient Communication:</strong> Reduces paperwork and manual processes</li>
<li><strong>Scalable Solution:</strong> Works for small classes or large institutions</li>
<li><strong>Always Available:</strong> 24/7 access from any device with internet</li>
</ul>
</div>
</div>
<div class="section">
<div class="section-title">3. Troubleshooting</div>
<div class="content">
<ul class="feature-list">
<li>If login fails, check your roll number and password</li>
<li>Clear browser cache if you experience issues</li>
<li>Use 'Clear All Data' button to reset the system</li>
<li>Contact system administrator for technical support</li>
</ul>
</div>
</div>
<div style="text-align: center;">
<a href="index.html" class="back-btn"> ↺ Back to Home</a>
</div>
</div>
</body>
</html>

```

### **Complete-database-setup.sql:**

```

-- COMPLETE DATABASE SETUP FOR SKILLQUEST
-- Run this entire script in Supabase SQL Editor
-- ENUM for roles (create only if it doesn't exist) do $$ begin
create type public.app_role as enum ('faculty','student'); exception
when duplicate_object then null; end $$;
-- Users profile (linked later when we wire Auth) create table if not exists

```

```

public.profiles (
    id uuid primary key default gen_random_uuid(), name text,
    roll text unique,
    role public.app_role default 'student', password text, -- Added password
    column created_at timestamptz default now()
);
-- Programs (SDPs)
create table if not exists public.programs (
    id uuid primary key default gen_random_uuid(),
    title text not null, description text, category text,
    status text check (status in ('upcoming','ongoing','completed')) default
    'upcoming',
    capacity int default 30,
    created_at timestamptz default now()
);
-- Enrollments (student roll per program) create table if not exists
public.enrollments (
    id uuid primary key default gen_random_uuid(),
    program_id uuid references public.programs(id) on delete cascade not null,
    student_roll text not null,
    enrolled_at timestamptz default now(), unique(program_id, student_roll)
);
-- Feedback
create table if not exists public.feedback (
    id uuid primary key default gen_random_uuid(),
    program_id uuid references public.programs(id) on delete cascade not null,
    student_roll text not null, student_name text,
    rating int check (rating between 1 and 5) not null, comment text,
    created_at timestamptz default now()
);
-- Topic votes (one vote per student per topic) create table if not exists
public.votes (
    id uuid primary key default gen_random_uuid(), topic_title text not null,
    category text, votes int default 0,
    created_at timestamptz default now()
);
create table if not exists public.vote_receipts (
    id uuid primary key default gen_random_uuid(),
    vote_id uuid references public.votes(id) on delete cascade not null,
    student_roll text not null,
    created_at timestamptz default now(), unique(vote_id, student_roll)
);
-- Suggestions (student ideas)
create table if not exists public.suggestions (
    id uuid primary key default gen_random_uuid(), student_roll text not null,
    student_name text, text text not null,
    created_at timestamptz default now()
)

```

```

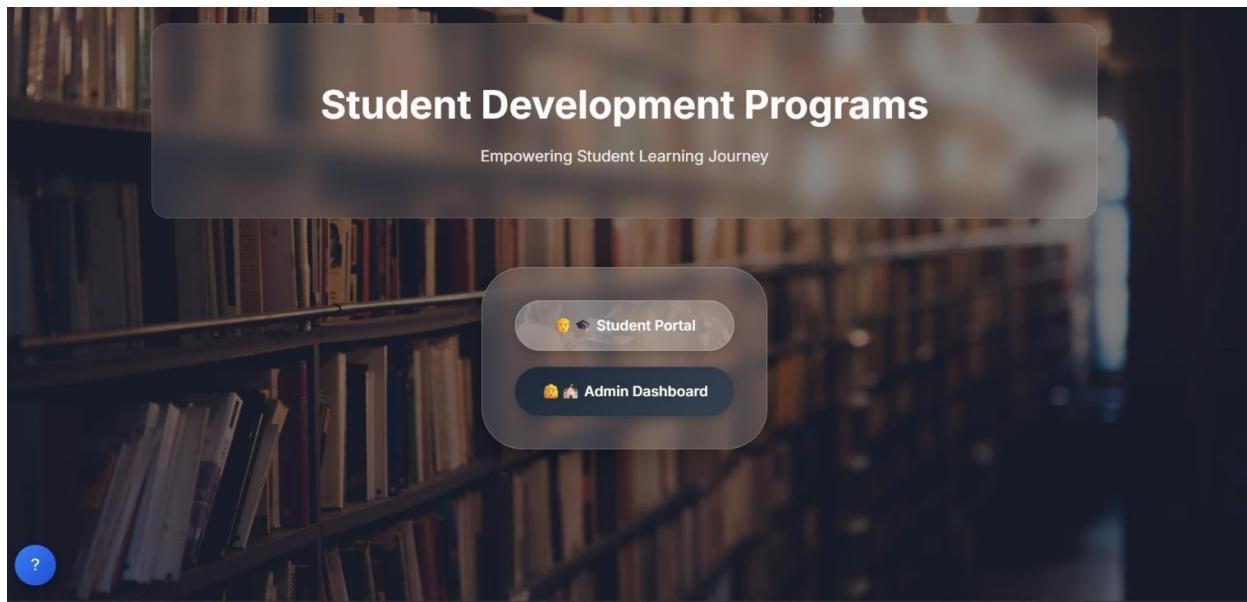
);
-- App settings (for universal faculty code) create table if not exists
public.settings ( key text primary key,
value jsonb not null,
updated_at timestamptz default now()
);
insert into public.settings(key, value)
values ('faculty_code', jsonb_build_object('code','IT2025'))
on conflict (key) do update set value = excluded.value, updated_at = now();
-- Add password column if it doesn't exist
ALTER TABLE profiles ADD COLUMN IF NOT EXISTS password TEXT;
-- Enable RLS
alter table public.profiles enable row level security; alter table public.programs
enable row level security; alter table public.enrollments enable row level
security; alter table public.feedback enable row level security; alter table
public.votes enable row level security;
alter table public.vote_receipts enable row level security; alter table
public.suggestions enable row level security; alter table public.settings enable
row level security;
-- Permissive policies for development do $$$
begin
-- Drop existing policies if they exist
drop policy if exists profiles_read on public.profiles; drop policy if exists
profiles_write on public.profiles; drop policy if exists profiles_update on
public.profiles; drop policy if exists programs_read on public.programs; drop
policy if exists programs_write on public.programs;
drop policy if exists programs_update on public.programs; drop policy if exists
programs_delete on public.programs; drop policy if exists enrollments_read on
public.enrollments; drop policy if exists enrollments_write on
public.enrollments; drop policy if exists enrollments_delete on
public.enrollments; drop policy if exists feedback_read on public.feedback;
drop policy if exists feedback_write on public.feedback; drop policy if exists
votes_read on public.votes;
drop policy if exists votes_write on public.votes; drop policy if exists
votes_update on public.votes; drop policy if exists votes_delete on
public.votes;
drop policy if exists voterec_read on public.vote_receipts; drop policy if exists
voterec_write on public.vote_receipts; drop policy if exists suggestions_read on
public.suggestions; drop policy if exists suggestions_write on
public.suggestions; drop policy if exists settings_read on public.settings;
drop policy if exists settings_update on public.settings;
-- Create new policies
create policy profiles_read on public.profiles for select using (true); create
policy profiles_write on public.profiles for insert with check (true);
create policy profiles_update on public.profiles for update using (true);
create policy programs_read on public.programs for select using (true); create
policy programs_write on public.programs for insert with check (true);

```

```
create policy programs_update on public.programs for update using (true);
create policy programs_delete on public.programs for delete using (true);
create policy enrollments_read on public.enrollments for select using (true);
create policy enrollments_write on public.enrollments for insert with check
(true);
create policy enrollments_delete on public.enrollments for delete using (true);
create policy feedback_read on public.feedback for select using (true); create
policy feedback_write on public.feedback for insert with check (true);
create policy votes_read on public.votes for select using (true); create policy
votes_write on public.votes for insert with check (true); create policy
votes_update on public.votes for update using (true); create policy
votes_delete on public.votes for delete using (true);
create policy voterec_read on public.vote_receipts for select using (true);
create policy voterec_write on public.vote_receipts for insert with check
(true);
create policy suggestions_read on public.suggestions for select using (true);
create policy suggestions_write on public.suggestions for insert with check
(true);
create policy settings_read on public.settings for select using (true); create
policy settings_update on public.settings for update using (true); end $$;
-- Verification queries
SELECT 'Tables created successfully' as status;
SELECT table_name FROM information_schema.tables WHERE
table_schema = 'public' AND table_name IN ('profiles', 'programs',
'enrollments', 'feedback', 'votes', 'vote_receipts', 'suggestions', 'settings');
SELECT column_name, data_type FROM information_schema.columns
WHERE table_name = 'profiles' AND table_schema = 'public';
```

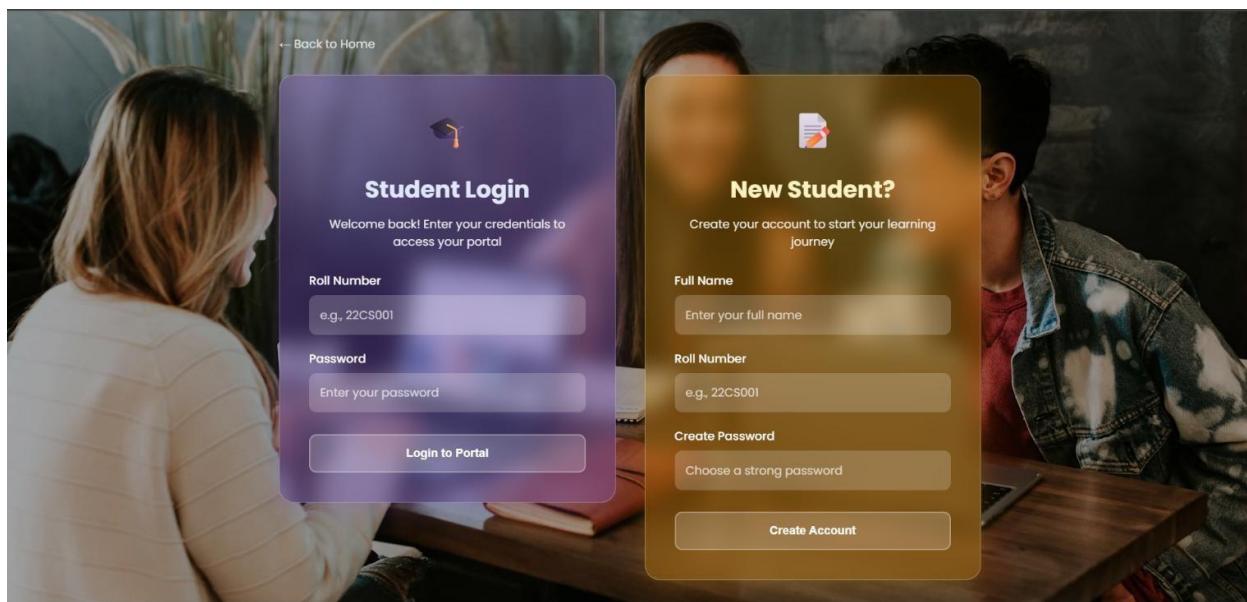
## User Interface of Website:

**Figure:** Home page.



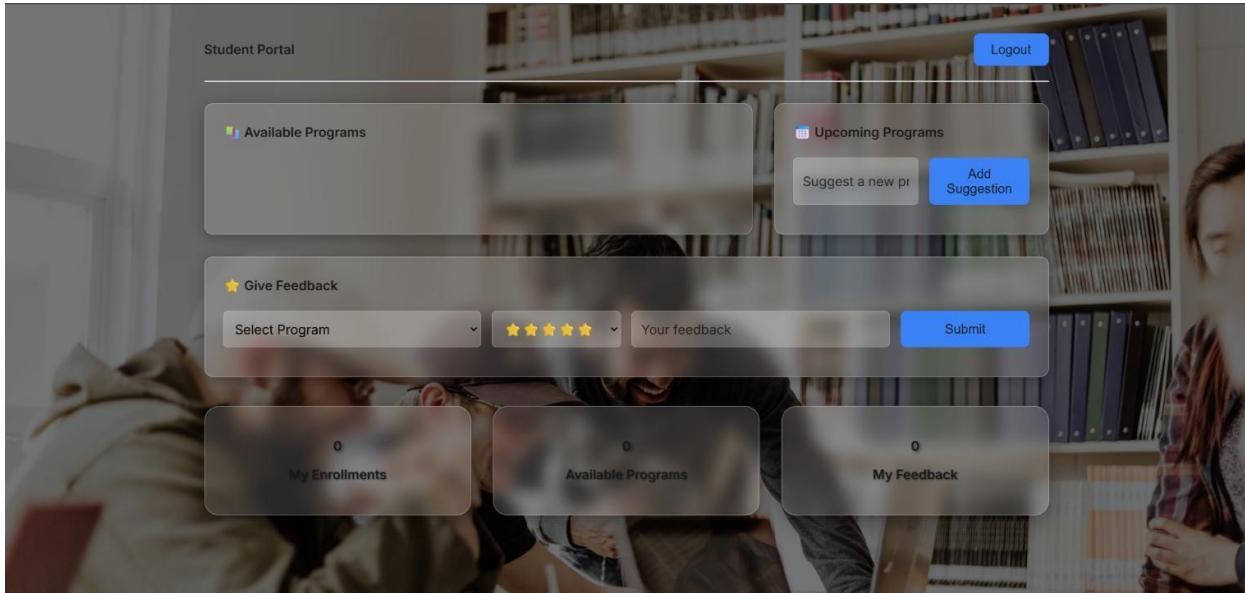
**Description:** In home page there is two buttons student and faculty you choose.

**Figure:** Student portal login and registration interface.



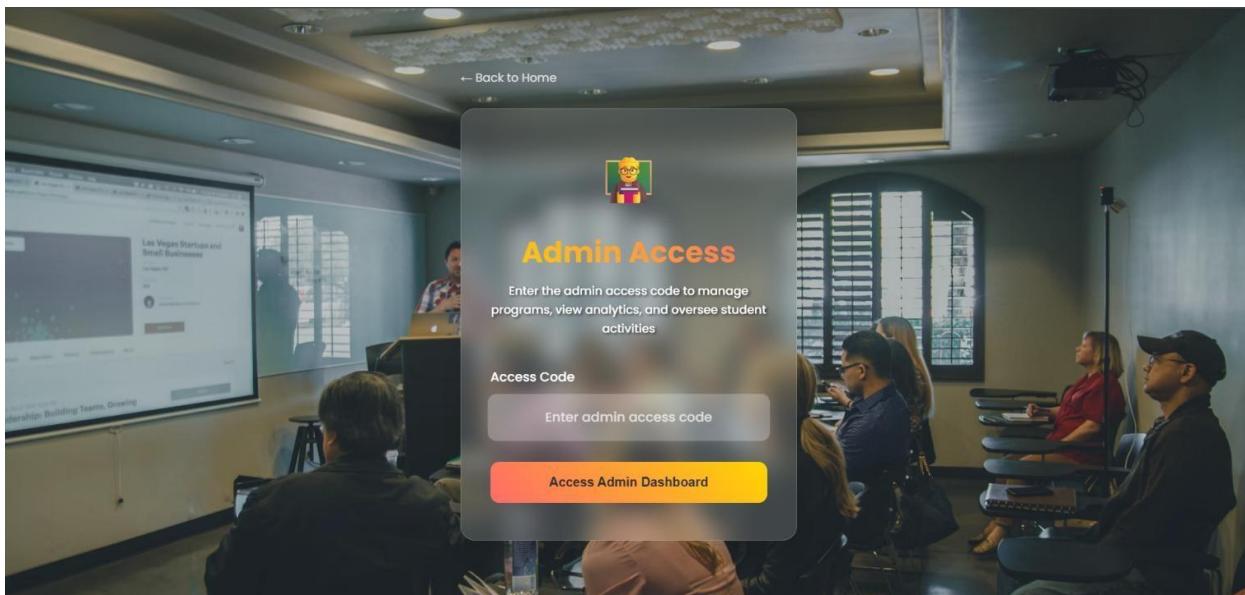
**Description:** By using this portal the student will register

**Figure:** Student portal dashboard interface



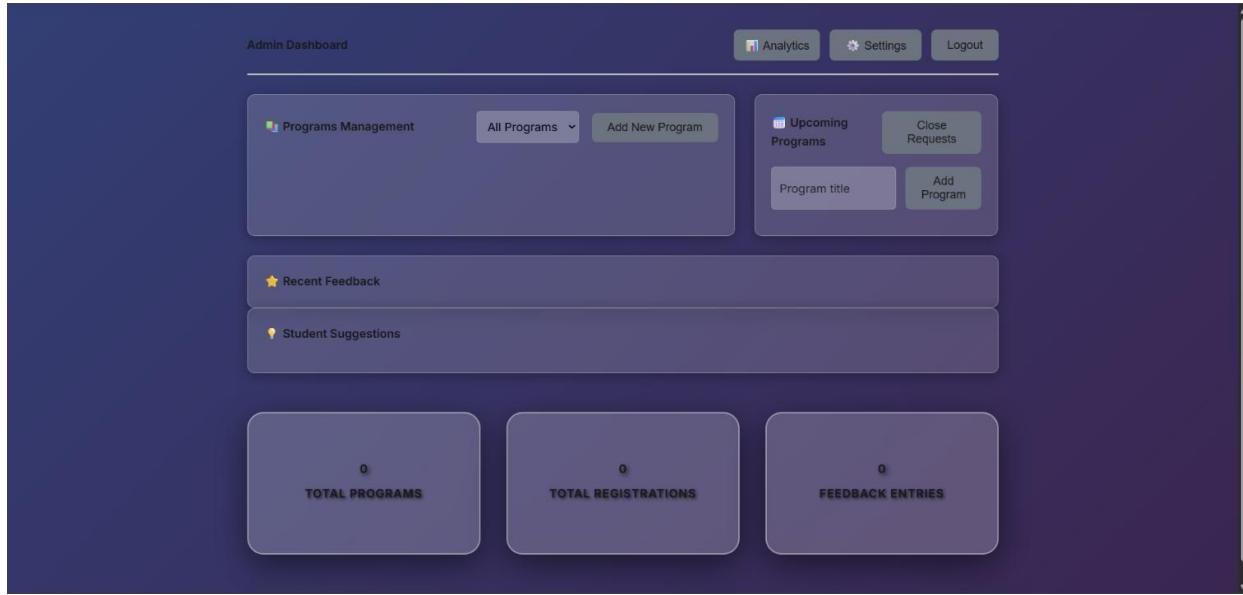
**Description:** here the student see upcoming programs giving feedback and managing enrollments

**Figure:** Users' Dashboard.



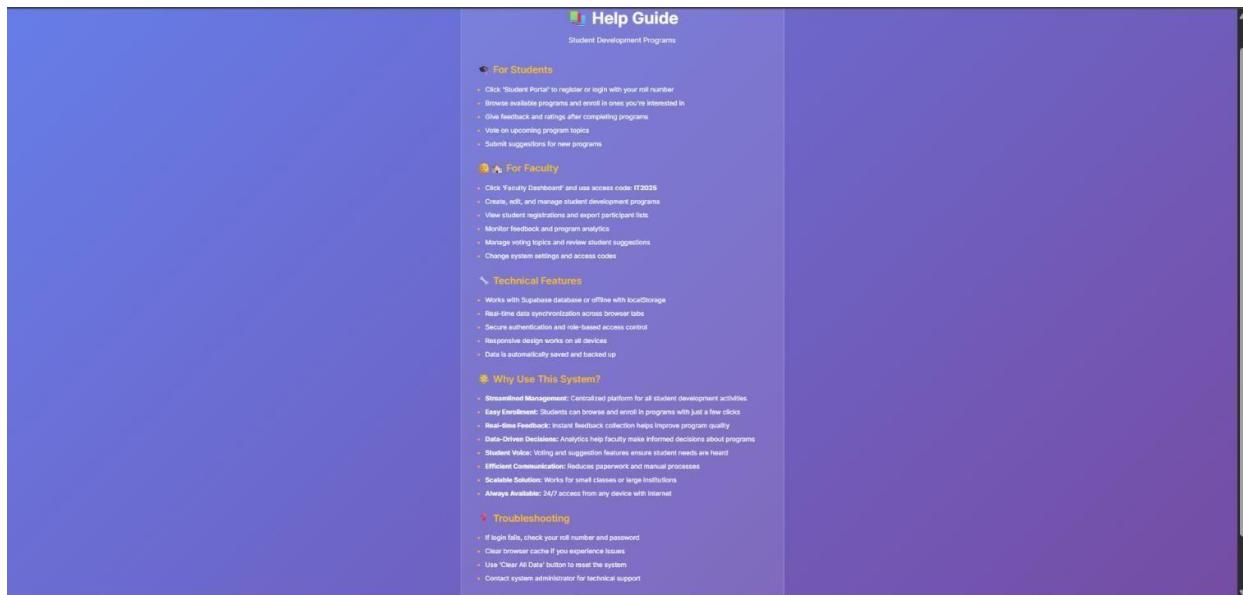
**Description:** The admin access will have a code to continue.

**Figure:** The admin access will have a code to continue



**Description:** Here the admin access add new programs and managing upcoming programs viewing feedbacks and suggestions.

**Figure:** : Help guide-Student development program system.



**Description:** It is an help guide the guide is split into the sections with the key instructions and features of different groups.

## **CHAPTER6: RECOMMENDATIONS AND CONCLUSIONS OF THE COMMUNITY SERVICE PROJECT.**

### **Conclusion**

The Student Technical Skill Development Planner successfully addresses the challenges in organizing and managing student development programs. It provides a digital platform that enhances communication, feedback collection, and program planning between students and faculty. By integrating real-time data, feedback analysis, and modern technologies, the system ensures continuous learning and aligns academic activities with industry needs. Overall, the project promotes efficiency, transparency, and active student participation, contributing to the holistic development of technical and professional skills.

### **Community Impact**

The project created a positive impact by promoting digital awareness and technical skill development among students. It improved communication between students and faculty, encouraging active participation in learning programs. The platform enhanced access to resources, supported continuous learning, and helped bridge the gap between academic education and industry requirements within the community.

### **Future Directions**

The project can be further enhanced by integrating AI-based skill recommendations to personalize learning paths for students.

Adding mobile app support will increase accessibility and ease of use.

Future updates may also include real-time chat, certificate generation, and cloud-based analytics to improve engagement and performance tracking. Collaboration with industry experts can be expanded to provide live workshops and internship opportunities for students.

## Student Self-Evaluation for the Community Service Project

Student Name: Maganti Sai Veni

Registration No: 23481A12D9

Period of CSP: from 19-05-2025 to 30-06-2025 and from 14-07-2025 to 26-07-2025

Date of Evaluation:

Name of the Person in-charge: Maganti Sai Veni

Address with mobile number: Department of Information Technology, SRGEC, Gudlavalleru-521356,  
9963580835.

**Please rate your performance in the following areas:**

**Rating Scale:**      **1 is lowest and 5 is highest rank**

<b>1) Oral communication</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>2) Written communication</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>3) Proactiveness</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>4) Interaction ability with community</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>5) Positive Attitude</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>6) Self-confidence</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>7) Ability to learn</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>8) Work Plan and organization</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>9) Professionalism</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>10) Creativity</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>11) Quality of work done</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>12) Time Management</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>13) Understanding the Community</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>14) Achievement of Desired Outcomes</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>15) OVERALL PERFORMANCE</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>

**Date:**

**Signature of the Student**

## **Student Evaluation by the Person in-charge in the Community/Habitation**

Student Name: Maganti Sai Veni

Registration No: 23481A12D9

Period of CSP: from 19-05-2025 to 30-06-2025 and from 14-07-2025 to 26-07-2025

Date of Evaluation:

Name of the Person in-charge: Dr. M. V. L. N. Raja Rao

Address with mobile number: Department of Information Technology, SRGEC, Gudlavalleru-521356,

**Please rate the student's performance in the following areas:**

**Please note that your evaluation shall be done independent of the student's self-evaluation**

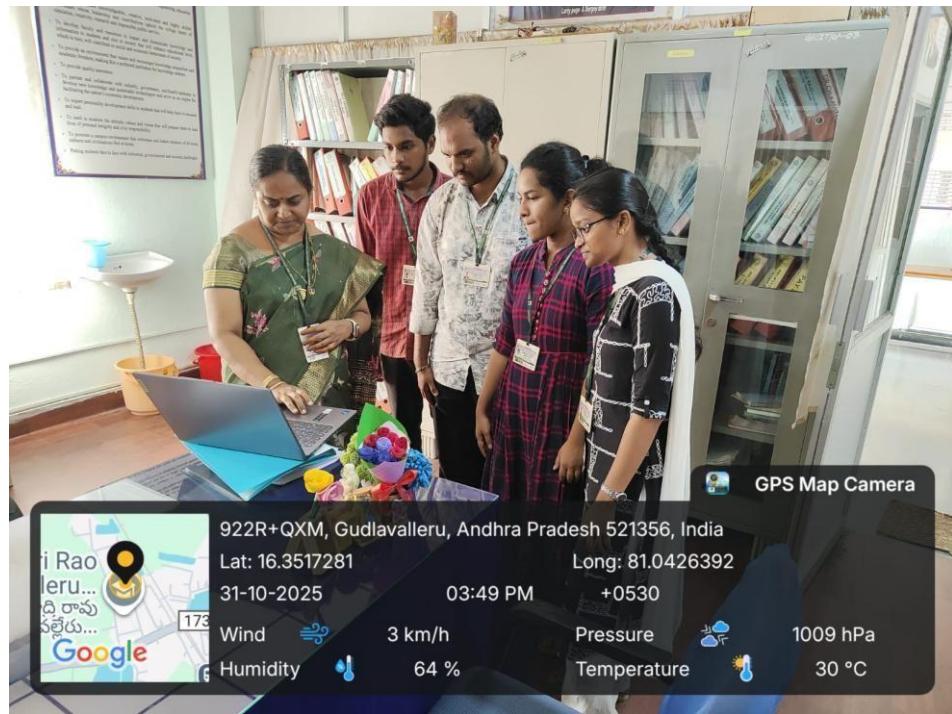
**Rating Scale: 1 is lowest and 5 is highest rank**

<b>1) Oral communication</b>	1	2	3	4	5
<b>2) Written communication</b>	1	2	3	4	5
<b>3) Proactiveness</b>	1	2	3	4	5
<b>4) Interaction ability with community</b>	1	2	3	4	5
<b>5) Positive Attitude</b>	1	2	3	4	5
<b>6) Self-confidence</b>	1	2	3	4	5
<b>7) Ability to learn</b>	1	2	3	4	5
<b>8) Work Plan and organization</b>	1	2	3	4	5
<b>9) Professionalism</b>	1	2	3	4	5
<b>10) Creativity</b>	1	2	3	4	5
<b>11) Quality of work done</b>	1	2	3	4	5
<b>12) Time Management</b>	1	2	3	4	5
<b>13) Understanding the Community</b>	1	2	3	4	5
<b>14) Achievement of Desired Outcomes</b>	1	2	3	4	5
<b>15) OVERALL PERFORMANCE</b>	1	2	3	4	5

Date:

Signature of the Supervisor

# Community Service Project Survey Photos



**Description:** Our team visited department of information technology branch HOD mam.



**Description:** Here our time explaning the website how to use it .

# **COMMUNITY SERVICE PROJECT WORK MAPPING WITH PROGRAMMME OUTCOMES**

## **PROGRAMME OUTCOMES(POs)**

Engineering Graduates will be able to:

**PO1:Engineering Knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

**PO2: Problem Analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

**PO3: Design/Development of Solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

**PO4: Conduct Investigations of Complex Problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

**PO5: Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

**PO6: The Engineer and The World:** Analyze and evaluate societal and environmental aspects where solving complex engineering problems has its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

**PO7: Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

**PO8: Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

**PO9: Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language and learning differences

**PO10: Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

**PO11: Life-Long Learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

### **Program Specific Outcomes (PSOs)**

Students will be able to

**PSO1 :** Organize, maintain and protect IT Infrastructural resources.

**PSO2 :** Design and Develop web, mobile, and smart apps based software solutions to the real world example

## PROJECT PROFORMANCE

Classification of Project	Application	Product	Research	Review

Note :Tick Appropriate category.

Community Service Project Outcomes	
Course Outcome (CO1)	Identify real community problems through field survey and stakeholder interaction.
Course Outcome (CO2)	Apply engineering knowledge to propose and design a solution addressing community needs
Course Outcome (CO3)	Develop and implement the proposed system/Prototype using appropriate tools, techniques, and teamwork
Course Outcome (CO4)	Evaluate the solution effectiveness through testing, validation, and user feedback.
Course Outcome (CO5)	Communicate project outcomes professionally with consideration for ethics, safety, society, and sustainability.

## Mapping Table

	CS3505: COMMUNITY SERVICE PROJECT												
Course outcomes	Program Outcomes and Program Specific Outcome												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1: Identify real community problems through field survey and stakeholder interaction.	2	3	2	1	1	2	2	3	3	3	2	1	1
CO2: Apply engineering knowledge to propose and design a solution addressing community needs.	3	3	3	2	2	3	2	2	2	2	2	3	3
CO3 Develop and implement the proposed system/Prototype using appropriate tools, techniques, and teamwork.	3	2	3	2	3	2	2	3	3	2	3	3	3
CO4: Evaluate the solution effectiveness through testing, validation, and user feedback	2	3	2	3	3	2	2	2	2	2	3	3	2
CO5: Communicate project outcomes professionally with consideration for ethics, safety, society, and sustainability.	1	2	2	2	2	2	3	3	3	3	3	2	2

**Note: Map each project outcomes with POs and PSOs with either 1 or 2 or 3 based on level of mappings as follows:**

- 1.Slightly (Low) mapped
- 2.Moderately (Medium) mapped
- 3.Substantially (High) mapped

<b>PROGRAMME OUTCOMES</b>	<b>MAPPING HIGH-3/MEDIUM-2/LOW-1</b>	<b>JUSTIFICATION</b>
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

**11**

## **PROGRAM SPECIFIC OUTCOMES**

<b>PSOs</b>	<b>1</b>	<b>2</b>
<b>PROJECT</b>		

<b>PROGRAM SPECIFIC OUTCOMES</b>	<b>MAPPING HIGH-3/MEDIUM- 2/LOW-1</b>	<b>JUSTIFICATION</b>
1		
2		