A Project Report on

MERN Based PLACEMENT MANAGEMENT PORTAL

A Dissertation submitted to JNTU Hyderabad in partial fulfilment of the academic requirements for the award of the degree.

Bachelor of Technology

In

Computer Science and Engineering

Submitted by

N. Srikanth Reddy
(20H51A0570)
N. Sathwik
(20H51A05E9)
N. Krishna Koushik
(20H51A05L3)

Under the esteemed guidance of

Ms. E. Krishnaveni
(Assistant Professor)

Department of Computer Science and Engineering

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

(UGC Autonomous)

*Approved by AICTE *Affiliated to JNTUH *NAAC Accredited with A+ Grade

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

2020-2024

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Major Project report entitled **"MERN Based Placement Management Portal"** being submitted by N. Srikanth Reddy (20H51A0570), N. Sathwik (20H51A05E9), N. Krishna Koushik (20H51A05L3) in partial fulfilment for the award of **Bachelor of Technology in Computer Science and Engineering** is a record of bonafide work carried out his/her under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University of Institute for the award of any Degree.

Ms. E. Krishnaveni **Assistant Professor** **Dept. of CSE**	**Dr. Siva Skandha Sanagala** **Associate Professor and HOD** **Dept. of CSE**	**EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

With great pleasure we want to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project work a grand success.

We are grateful to **Ms. E. Krishnaveni, Assistant Professor**, Department of Computer Science and Engineering for her valuable technical suggestions and guidance during the execution of this project work.

We would like to thank **Dr. Siva Skandha Sanagala,** Head of the Department of Computer Science and Engineering, CMR College of Engineering and Technology, who is the major driving forces to complete my project work successfully.

We are very grateful to **Dr. Ghanta Devadasu**, Dean-Academics, CMR College of Engineering and Technology, for his constant support and motivation in carrying out the project work successfully.

We are highly indebted to **Major Dr. V A Narayana,** Principal, CMR College of Engineering and Technology, for giving permission to carry out this project in a successful and fruitful way.

We would like to thank the **Teaching & Non- teaching** staff of Department of Computer Science and Engineering for their co-operation

We express our sincere thanks to **Shri. Ch. Gopal Reddy**, Secretary& Correspondent, CMR Group of Institutions, and **Shri Ch Abhinav Reddy**, CEO, CMR Group of Institutions for their continuous care and support.

Finally, we extend thanks to our parents who stood behind us at different stages of this Project. We sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project work.

N. Srikanth Reddy	20H51A0570
N. Sathwik	20H51A05E9
N. Krishna Koushik	20H51A05L3

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	LIST OF FIGURES	iii
	ABSTRACT	iv
1	**INTRODUCTION**	1-3
	1.1. Problem Statement	2
	1.2. Research Objective	2
	1.3. Project Scope and Limitations	2-3
2	**BACKGROUND WORK**	4-10
	2.1. Manual Process	5-6
	2.1.1. Introduction	5
	2.1.2. Merits, Demerits and Challenges	5-6
	2.1.3. Implementation	6
	2.2. Placement Management System for Campus Recruitment	6-8
	2.2.1. Introduction	6
	2.2.2. Merits, Demerits and Challenges	7
	2.2.3. Implementation	8
	2.3. Placement Management System	9-10
	2.3.1. Introduction	9
	2.3.2. Merits, Demerits and Challenges	9
	2.3.3. Implementation	10
3	**PROPOSED SYSTEM**	11-26
	3.1. Objective of Proposed Model	12
	3.2. Algorithms Used for Proposed Model	12-13
	3.3. Proposed Method	13-14
	3.4. Designing	15-17
	3.4.1. Architectural Diagram	15
	3.4.2. Class Diagram	16

	3.4.3. Use Case Diagram	17
	3.5. Stepwise Implementation and Code	18-26
4	**RESULTS AND DISCUSSION**	27-36
	4.1. Results and Discussion	28-36
5	**CONCLUSION**	37-38
	5.1. Conclusion and Future Enhancement	38
	REFERENCES	39-41
	GITHUB LINK	42
	PUBLISHED PAPER	43-52
	PUBLICATION CERTIFICATES	53-56

List of Figures

FIGURE NO.	TITLE	PAGE NO.
2.2.3.1	Architectural Diagram	8
2.3.2.1	Data Flow Diagram	10
3.4.1.1	Architectural Diagram	15
3.4.2.1	Class Diagram	16
3.4.3.1	Use Case Diagram	17
4.1.1	Student Login Page	28
4.1.2	Student Home Page	29
4.1.3	Student Skills Page	29
4.1.4	Student Achievements Page	30
4.1.5	Student Certifications Page	30
4.1.6	Student Projects Page	31
4.1.7	Resume Upload Page	31
4.1.8	Job Postings	32
4.1.9	Problem Solving	32
4.1.10	Submissions Page	33
4.1.11	Leaderboard	33
4.1.12	Admin Login Page	34
4.1.13	Department Catalog	34
4.1.14	Student Interests	35
4.1.15	Add Company	35
4.1.16	Company Requirements vs Student Profile	36

ABSTRACT

The MERN Based Placement Management Portal is a web-based Placement Management System developed using the MERN Stack, designed to streamline and simplify the placement process for educational institutions. This indispensable tool enables the placement cell, students, and administrators by providing a centralized platform for efficient communication and data management.

With the MERN Based Placement Management Portal, students can easily access and maintain their interest's details, eliminating the need for manual paperwork and saving valuable time. Through the application, students can select companies based on their individual interests, skills, and qualifications. Along with these the student can submit their certifications. And the interesting fact about our Portal is it consists of DSA Tracker where the students can solve problems, and check their progress.

The system leverages the power of the MERN Stack (MongoDB, Express.js, React, and Node.js) to ensure a robust and scalable solution. MongoDB serves as the database for storing student information, preferences, certifications and problems, while Express.js and Node.js handle the backend logic. React, on the other hand, takes care of the frontend, providing an intuitive and user-friendly interface.

CHAPTER 1
INTRODUCTION

CHAPTER 1
INTRODUCTION

1.1. Problem Statement

The problem we're addressing is the inefficiency of manual placement processes within educational institutions, where paperwork consumes time and causes missed opportunities. Our solution is a MERN-based centralized portal, a game-changer in education placement. It streamlines communication, eliminates manual tasks, and introduces a DSA (Data Structures and Algorithms) tracker. It empowers students and administrators with an intuitive platform, transforming the education placement experience.

1.2. Research Objective

- The objective of our research is to streamline the process of managing placements, job opportunities, students performances, placements and students data involved within educational institution.

1.3. Project Scope and Limitations

Project Scope:

1. Student Management:
   The Portal offers a comprehensive platform for students to efficiently manage their placement-related data, including interests, company preferences, qualifications, and certifications. This feature simplifies the placement process by eliminating the need for manual paperwork.
2. Job Listings:
   The system provides an extensive database of job listings, allowing students to browse and select companies that align with their interests and qualifications. This centralized job listing feature streamlines the job search process and enhances students' opportunities.
3. Application Management:

The portal facilitates the application process, enabling students to apply to their preferred companies seamlessly. This application management system ensures a systematic approach to company applications and offers a centralized view of application status.

4. DSA Tracker:

   A notable feature of our portal is the DSA Tracker, which empowers students to enhance their data structures and algorithms skills. It offers a platform for problem-solving and progress tracking, contributing to their overall professional development.

Limitations:

1. Educational Institution Specific:

   The MERN Based Placement Management Portal is tailored for educational institutions, limiting its application to the context of academic placement processes.

2. Dependence on Internet Connectivity:

   The portal relies on internet connectivity, which may pose challenges in regions with unreliable or limited internet access.

3. Maintenance and Updates:

   Regular maintenance and updates are necessary to keep the system efficient and secure, which requires dedicated resources and attention to ensure its continued functionality and relevance.

CHAPTER 2
BACKGROUND WORK

CHAPTER 2
BACKGROUND WORK

2.1 Manual Process:

2.1.1. Introduction:

The existing solution for managing the placement process in educational institutions primarily relies on manual work and is characterized by a labor-intensive and time-consuming approach. Placement officers traditionally send Google Forms or similar survey tools to students, requesting them to indicate their interest in attending specific campus drives or placement events. This method aims to gather student preferences and data to facilitate the placement process.

2.1.2. Merits, Demerits and Challenges:

Merits:

- Data Collection: It allows for data collection on student preferences for specific companies and events.
- Low-Cost: Using existing, free survey tools like Google Forms can be cost-effective for educational institutions.

Demerits:

- Manual Work: The process heavily relies on manual work, involving the distribution and collection of surveys, which can be time-consuming for both students and placement officers.
- Error-Prone: Manual data entry and interpretation of student responses can lead to errors and inconsistencies in managing the placement process.
- Scalability: This approach may struggle to scale with a large number of students and companies.

Challenges:

- Time and Effort: The manual distribution and collection of forms demand a significant investment of time and effort from both students and placement officers.
- Data Integrity: The manual nature of this method can introduce errors, affecting the accuracy and integrity of the collected data.
- Limited Features: It lacks features like a comprehensive database, certifications submission, and DSA tracking, which are essential for modern placement processes.

2.1.3. Implementation of Manual Process:

To implement this existing solution, placement officers create Google Forms or similar surveys tailored to each campus drive or event. The collected data is then processed and analyzed manually by the placement cell to determine which students should be shortlisted for specific opportunities. However, this approach faces challenges such as data integrity issues, time constraints, and limited insights into student performance. As a result, there is a clear need for a more streamlined and automated solution like the MERN Based Placement Management Portal to improve efficiency and reduce manual work in the placement process.

2.2. Placement Management System for Campus Recruitment:

2.2.1. Introduction:

In the context of educational institutions and their placement processes, an existing Placement Management System serves as a valuable tool for placement cells, students, and administrators. However, this system lacks some critical features, such as a DSA Tracker and certificate upload option.

2.2.2. Merits, Demerits and Challenges:

Merits:

- The existing Placement Management System offers a centralized platform for communication and data management, benefiting both placement cell personnel and students.
- It allows students to access and maintain their placement-related information, eliminating the need for cumbersome manual paperwork.
- The system enables students to express their preferences and qualifications, facilitating the selection of companies that align with their interests.

Demerits:

- The system lacks essential features like a DSA Tracker, which is crucial for monitoring and enhancing students' data structures and algorithms skills. This absence limits the system's capacity to support students in problem-solving and skill development.
- Additionally, the system does not provide an option for students to upload their certifications, potentially leading to inaccuracies in the representation of their skills and qualifications.

Challenges:

- The absence of a DSA Tracker hinders placement officers' ability to track students' daily performance and evaluate their problem-solving skills, making it challenging to gauge their true capabilities.

2.2.3. Implementation of Placement Management System for Campus Recruitment:

Fig 2.2.3.1 Architectural Diagram

The proposed system by Ajeena Sunny, Aneena Felix, Angelin Saji, Christina Sevastian and Praseetha V.M. [2] is a web-based placement management platform designed to streamline the process of collecting and managing student placement data. Developed using the Laravel framework with a Model-View-Template pattern, it offers different account types for users such as Principal, HOD, placement officers, coordinators, and students. Each student can upload their information, which is then sorted based on eligibility criteria set by companies. The system utilizes MySQL for database management and sends out email notifications with links for eligible candidates to express interest in specific placement opportunities. A key feature is its machine learning application, which predicts a student's placement probabilities in different companies based on departmental placement history, skill set, academic performance, and other relevant data.

2.3. Placement Management System:

2.3.1. Introduction:

In the realm of placement management systems, there exists an alternative solution that provides only partial support for managing student requests to attend campus drives. This system falls short of offering placement officers the crucial functionality of accepting or rejecting student requests, resulting in a less versatile approach to managing campus placements within educational institutions.

2.3.2. Merits, Demerits and Challenges:

Merits:

- This system allows students to express their interest in attending campus drives and submit their applications.
- Basic communication between students and the placement cell is facilitated, offering a straightforward method for students to apply for various opportunities.

Demerits:

- The system lacks the essential capability for placement officers to accept or reject student requests. This omission limits the placement officers' ability to efficiently manage and control the student application process.
- There is no structured system for confirming or denying student participation in campus drives, leading to potential organizational issues and delays.

Challenges:

- The absence of an accept/reject mechanism hinders placement officers' ability to effectively screen and manage student participation, which can result in a less organized and streamlined placement process.
- This deficiency may lead to issues in terms of tracking student participation and making informed decisions about student applications.

2.3.3. Implementation of Placement Management System:

The Placement Management System given by Farheen Taqi Rizvi, Naushin A. Khan, Saurabh Upadhyay, Sonali Suryawanshi, and Shiburaj Pappu [1] aims to streamline the organization of student and company records by focusing on specific categories. It empowers placement officers to efficiently manage student data, sorting it based on percentage criteria required by companies while allowing students to review their information, thus addressing existing system limitations. Admin privileges prioritize comprehensive data review, sorting, and deletion. Key benefits include centralized notice dissemination, integration with the exam cell for academic data validation, simplified information retrieval, and automated functionalities like percentage and CGPA calculation to enhance accuracy and efficiency for students.

Fig 2.3.2.1 Data Flow Diagram

CHAPTER 3
PROPOSED SYSTEM

CHAPTER 3
PROPOSED SYSTEM

3.1. Objective of Proposed Model

- The objective of our proposed model is to streamline the process of managing placements, job opportunities, placements and students data within an educational institution.
- To maintain and manage students complete details including academic, skills, achievements, certifications, projects and resume.
- To provide a coding platform for students to improve their problem solving skills in DSA.

3.2. Algorithms Used for Proposed Model

- **Timsort Algorithm:**

  Step 1: Divide the array into the number of blocks known as run.

  Step 2: Consider the size of run, either 32 or 64.

  Step 3: Sort the individual elements of every run one by one using insertion sort.

  Step 4: Merge the sorted runs one by one using the merge function of merge sort.

  Step 5: Double the size of merged sub-arrays after every iteration.
- **Linear Search Algorithm:**

  Step 1: Every element is considered as a potential match for the key and checked for the same.

  Step 2: If any element is found equal to the key, the search is successful and the index of that element is returned.

  Step 3: If no element is found equal to the key, the search yields “No match found”.
- **Binary Search Algorithm:**

  Step 1: Divide the search space into two halves by finding the middle index “mid”.

  Step 2: Compare the middle element of the search space with the key.

  Step 2.1: If the key is found at middle element, the process is terminated.

  Step 2.2: If the key is not found at middle element, choose which half will be used as the next search space.

Step 2.3: If the key is smaller than the middle element, then the left side is used for next search.

Step 2.4: If the key is larger than the middle element, then the right side is used for next search.

Step 3: This process is continued until the key is found or the total search space is exhausted.

3.3. Proposed Method

Our proposed method consists of two modules as Student module and Admin module.

STUDENT MODULE:

Every student is assigned a unique ID by the college, which they use to log in along with their password. The system automatically records the academic details of the students. Here's a breakdown of the features:

- **Profile Building:** Students have the opportunity to enhance their profiles by adding their skills and experiences, thereby improving their chances in the job market.
- **Certifications:** This section allows students to upload certifications to validate their skill sets, enhancing their credibility to potential employers.
- **Resume:** Students can upload their resumes for the placement officer's reference, ensuring that their qualifications and experiences are easily accessible during the job application process.
- **Projects:** Students can showcase their projects by uploading descriptions and links, highlighting their practical skills and achievements to potential recruiters.
- **Dream Company Selection:** Students can explore job postings from their dream companies, access detailed company information, and apply directly to the positions they're interested in.
- **Check Approval Status:** This feature enables students to track the status of their applications to companies, ensuring transparency and keeping them informed throughout the application process.

- DSA Problem Solving: This section provides a platform for students to practice solving Data Structures and Algorithms (DSA) problems, enhancing their coding skills and preparing them for technical interviews.

ADMIN MODULE:

The Placement Officer, acting as the admin of the portal, accesses the system using specific login credentials. Here are the functionalities available to the admin:

- **Job Posting:** The admin can post job openings on behalf of companies, specifying job descriptions, eligibility criteria, and other relevant details to attract suitable candidates.
- **Access Students Details:** The admin has access to detailed information about the students, facilitating effective management and coordination throughout the placement process.
- **Access Students Documents:** Placement officers can view documents uploaded by students, including certificates and resumes, aiding in the evaluation of candidates' qualifications and suitability for specific roles.
- **View Students Company Interests:** This feature allows the admin to see which companies students are interested in, enabling strategic planning and coordination to match students with suitable job opportunities.
- **View Leaderboard:** The admin can view the leaderboard to track student performance, identifying high achievers and potentially recognizing outstanding contributions to the placement process.

3.4. Designing

3.4.1. Architectural Diagram

Fig 3.4.1.1 Architectural Diagram

3.4.2. Class Diagram

Fig 3.4.2.1 Class Diagram

3.4.3. Use Case Diagram

Fig 3.4.3.1 Use Case Diagram

3.5. Stepwise Implementation and Code

Implementation:

Step 1: Firstly, open the project folder in vs code.

Step 2: Open the terminal and split the terminal into two parts.

Step 2.1: In the first terminal, move to back-end folder by running the command “cd back-end’.

Step 2.2: In the second terminal, move to front-end folder by running the command “cd front-end’.

Step 3: Now do the following 2 steps.

Step 3.1: Run the command “nodemon server.js” in the back-end directory.

Step 3.2: After running the backend server, run the command “npm start” in the front-end directory

Step 4: Now the application will run on the localhost server “http://localhost:3000/”

PROJECT CODE

Server.Js

```
import express from "express";
import mongoose from "mongoose";
import cors from "cors";
import loginModel from "./models/loginModel.js";
import skillModel from './models/skillModel.js';
import achieveModel from './models/achieveModel.js';
import companyModel from "./models/companyModel.js";
import path from "path";
import bodyParser from 'body-parser';
import multer from 'multer';
import { config } from "dotenv";
```

```
import stdinterestModel from "./models/stdinterestModel.js";
import certificationModel from "./models/certificationModel.js";
import resumeModel from "./models/resumeModel.js";
import codeSubmissionModel from "./models/codeSubmission.js";
import solCountModel from "./models/solCountModel.js";
import os from "os";
import { spawn } from 'child_process';
import fs from 'fs';
import validator from 'validator';
import stdProjectModel from "./models/stdProjectModel.js";config();
const app = express();
app.use(cors());
const upload = multer({
  storage: multer.memoryStorage(),
  limits: {
      fileSize: 5 * 1024 * 1024,},});
app.use(bodyParser.json());
const port = process.env.PORT || 5010;
app.post('/', async (req, res) => {
    const { uid, password } = req.body;try {
        const check = await loginModel.findOne({ uid, password }); if (check) {
            res.json("exist");} else {
            res.json("notexist");}} catch (e) {res.json("notexist"); });
app.get('/AddSkills/get-skills/:uid', async (req, res) => {
  const { uid } = req.params;
  try {
    let student = await skillModel.findOne({ uid });
    if (!student) {
      student = new skillModel({ uid, skills: [] });
      await student.save();}
```

```
    res.json(student.skills);
  } catch (e) {
    console.log(e);res.status(500).json('Error getting skills')}});
app.post('/AddSkills/add-skill/:uid', async (req, res) => {
  const { uid, skill } = req.body;
  try {
    let student = await skillModel.findOne({ uid });
    if (!student) {
      student = new skillModel({ uid, skills: [] });
      await student.save();}
    if (student.skills.includes(skill)) {
      res.json('This skill already exists!');
    } else {
      student.skills.push(skill);
      await student.save();
      res.json('Skill added successfully');}
  } catch (e) {
    res.json('Error adding skill');}});
app.get('/Achievements/get-achievements/:uid', async (req, res) => {
  const { uid } = req.params;
  try {
    let student = await achieveModel.findOne({ uid });
    if (!student) {
      student = new achieveModel({ uid, skills: [] });
      await student.save();}
    res.json(student.achievements);
  } catch (e) {
      student = new achieveModel({ uid, achievements: [] });
      await student.save();}
    if (student.achievements.includes(achievement)) {
```

```
      res.json('This Achievement already exists!');
    } else {
      student.achievements.push(achievement);
      await student.save();
      res.json('Achievement added successfully');
    }
  } catch (e) {
    res.json('Error adding Achievement');
  }
});
app.post('/AddCompanies/add-company', upload.any(), async (req, res) => {
  const { companyName, jobTitle, reqSkills, jobCriteria, cmpPackage } = req.body;
  const file = req.files[0];
  console.log(req.files);
  const company = new companyModel({{
      data: file.buffer,
      contentType: file.mimetype}})try
    const savedCompany = await company.save();
    res.json({ message: 'Company added successfully', company: savedCompany });
  } catch (err) {
    res.status(400).json({ message: 'Failed to add company', error: err.message }) });
 ({ companyName: companyToDelete });
    await stdinterestModel.deleteOne({companyName: companyToDelete});
    res.json({ message: 'Company deleted successfully' });
  } catch (err) {
    res.status(400).json({ message: 'Failed to delete company', error: err.message }) );
app.post('/Apply/std', async (req, res) => {
  const { companyName, uid } = req.body;
  try {
    let stdInterest = await stdinterestModel.findOne({ companyName });
```

```
    if (!stdInterest) {
      stdInterest = new stdinterestModel({ companyName, students: [] });
      await stdInterest.save()
    if (stdInterest.students.find(student => student.studentId === uid)) {
      res.json('You have already Applied!');
    } else {
      stdInterest.students.push({ studentId: uid, status: 'pending' });
      await stdInterest.save();
      res.json('Applied Successfully');}
  } catch (e) {
    res.json('Error in Applying'});
app.post('/api/updateStatus/:company', async (req, res) => {
  const companyName = req.params.company;
  const { studentId, status } = req.body;
  try {
    let stdInterest = await stdinterestModel.findOne({ companyName: companyName });
    if (!stdInterest) {
      return res.json({ message: companyName });
    }
    let student = stdInterest.students.find(student => student.studentId === studentId);
    if(!student){
      stdInterest.students.push({ studentId: studentId, status: status });
      await stdInterest.save();
      res.json({message:`${status} Sucessfully`});}
    else{
    student.status = status;
    await stdInterest.save();
    res.json({ message: 'Status updated successfully' });}
  } catch (error) {
    console.error(error);
```

```
  res.status(500).json({ message: 'Server error' });}});
app.get('/UploadResume/get-resume/:uid', async (req, res) => {
 try {
    const doc = await resumeModel.findOne({ uid: req.params.uid });
    if (!doc) {
       res.send('No resume found');
    } else {
       res.contentType(doc.resume.contentType);
       res.send(doc.resume.data);
    }
 } catch (err) {
    res.status(500).send(err);
})
app.post('/UploadResume/upload/:uid', upload.single('resume'), async (req, res) => {
 try {
    const doc = await resumeModel.findOneAndUpdate(
       { uid: req.params.uid },
       { resume: { data: req.file.buffer, contentType: req.file.mimetype } },
       { upsert: true, new: true }
    );
    res.status(200).send('Resume uploaded successfully');
 } catch (err) {
    res.status(500).send(err);
 }});
app.post('/api/submit', async (req, res) => {
 const { uid, problemId, problemName, code, language, testCases } = req.body; // Include language
 let status = 'Accepted';
 for (let i = 0; i < testCases.length; i++) {
   const { input, output: expectedOutput } = testCases[i];
   try {
```

```
      const output = await compileAndRun(code, language, input);
      if (output.includes('Error')) {
        status = "Compile/Runtime Error";
        break;  }
      if(!checkOutput(output, expectedOutput)) {
        status = "Wrong Answer";
        break;}
    } catch (error) {
      status = 'Rejected';
      break;}
  const submission = await codeSubmissionModel.findOne({ uid, problemId });
  if (submission) {
    submission.code.push(code);
    submission.status.push(status);
    await submission.save();
  } else {
    const newSubmission = new codeSubmissionModel({ uid, problemId, problemName, code:
[code], status: [status] });
    await newSubmission.save();}
  if (status === 'Accepted') {
    let student = await solCountModel.findOne({ uid });
    if (student) {
      if (!student.solvedProblems.includes(problemId)) {
        student.solvedProblems.push(problemId);
        student.solvedCount += 1;
        await student.save();}} else {
      const newStudent = new solCountModel({ uid, solvedCount: status === 'Accepted' ? 1 : 0,
solvedProblems: [problemId] });
      await newStudent.save();} }
  res.json({ message: `${status}` })}});
```

```
function compileAndRun(code, language, input) {
  let codeKeywords;
  if (language === 'java') {
    codeKeywords = javaKeywords;
    if (!codeKeywords.some(keyword => code.includes(keyword))) {
      return Promise.resolve('Invalid Code')   }
  } else if (language === 'python') {
    codeKeywords = [];
  } else if (language === 'cpp') {
    ]
    if (language === 'python') {
      process = spawn('python', ['-c', code]);; });
      compile.on('close', (code) => {
        if (code !== 0) {
          cleanup();
          resolve(errorOutput);
          return;}
        process = spawn('java', ['Main']);
        timer = setTimeout(handleTimeout, timeLimit);
        runProcess(process, input, resolve, reject);
      }); return;
    } else if (language === 'cpp') {
      fs.writeFileSync('main.cpp', code);
      const compile = spawn('g++', ['main.cpp', '-o', 'main']);
      compile.stderr.on('data', (data) => {
        errorOutput += data;});
      compile.on('close', (code) => {
        if (code !== 0) {cleanup();
          resolve(errorOutput);return;}
        process = spawn('./main');
```

```
      timer = setTimeout(handleTimeout, timeLimit);
      runProcess(process, input, resolve, reject);});
    return;} else { cleanup();
    resolve('Unsupported language');
    return;}
  timer = setTimeout(handleTimeout, timeLimit);
  runProcess(process, input, resolve, reject); });}
```

CHAPTER 4
RESULTS AND DISCUSSION

CHAPTER 4
RESULTS AND DISCUSSION

4.1. Results & Discussion

- Our proposed system provides an user friendly interface to students and placement coordinators. It ensures effective data management, for students in managing their complete data including academic, skills, certifications, projects, company preferences and resume, and for placement coordinators in managing the job listings and students preferences.
- It provides comprehensive coding platform for students to solve and improve their problems solving skills in DSA and check progress.
- The placement coordinator can track the students progress in DSA through the leaderboard.

Fig 4.1.1 Student Login Page

Fig 4.1.2 Student Home Page

Fig 4.1.3 Student Skills Page

Fig 4.1.4 Student Achievements Page

Fig 4.1.5 Student Certifications Page

Fig 4.1.6 Student Projects Page

Fig 4.1.7 Resume Upload Page

Fig 4.1.8 Job Postings

Fig 4.1.9 Problem Solving

Fig 4.1.10 Submissions Page

Fig 4.1.11 Leaderboard

Fig 4.1.12 Admin Login Page

Fig 4.1.13 Department Catalog

Fig 4.1.14 Student Interests

Fig 4.1.15 Add Company

Fig 4.1.16 Company Requirements vs Student Profile

CHAPTER 5
CONCLUSION

CHAPTER 5
CONCLUSION

5.1. Conclusion and Future Enhancement

Overall our proposed method is a comprehensive solution that provides students with an easy-to-use platform to efficiently manage their data, search for jobs, and seamlessly apply to their desired companies. Incorporating a DSA tracker accelerates student's professional development by providing a dedicated space to improve their Data Structures and Algorithms (DSA) skills. Provides an easy way for college placement cells to access students data and manage placements data. Our project helps the placement coordinator to track the progress of the students in problem solving in DSA and also helps in analyzing the students profile.

In future our project will be enhanced by the following:

- **Company Logins:** In future company logins will be added, which helps the placement coordinators and companies to easily communicate and schedule the drives.
- **Student Experiences:** Student experiences with the drives can be gathered and make visible to the current students such that they can have an overview of how to prepare for the drives.
- **AI Enhancement:** With the AI enhancement the placement coordinators can get the students profiles that matches with the companies profile automatically, which makes the selecting process simple.

REFERENCES

REFERENCES

[1] Farheen Taqi Rizvi, Naushin A. Khan, Saurabh Upadhyay, Sonali Suryawanshi, Shiburaj Pappu. "Placement Management System." Journal of Emerging Technologies and Innovative Research (JETIR) 8. 4 (2021).

[2] Ajeena Sunny, Aneena Felix, Angelin Saji, Christina Sebastian, Praseetha V.M. "Placement Management System for Campus Recruitment." International Journal of Innovative Science and Research Technology 5. 5 (2020): 1705-1710.

[3] Santhosh Kumar H," Online Training and Placement Management system" International Journal of Engineering Research Technology (IJERT), ICACT-2016 Conference Proceedings.

[4] Mythili M, Aishwarya R, Shenbagam P, Sandhiya C," E Placement Management", International Journal of Pure and Applied Mathematics (IJPAM), Volume 119 No. 10 2018,1823-1834.

[5] Rajnesh Tripathi, Raghvendra Singh, Jaweria Usmani, "Campus Recruitment and Placement System", International Conference on Recent Innovations in Science and Engineering (Icrise -18), April, 2018.

[6] Anjali v, Jeyalakshmi PR, Anbubala R,Sri Mathura devi, Ranjini.V, "Web Based Placement Management System", International Journal of Computer Science and Information Technologies(IJCSIT), Vol. 7 (2) , 2016, 760-763.

[7] T. Gosavi, S. Gaikwad, R. Nazirkar and A. Salke, "ANDROID BASED TRAINING AND PLACEMENT AUTOMATION", National Conference-Ekalavya-2k15, ISSN 2348-6953.76SSGM Journal of Science and Engineering, Vol. 1, Issue 1, June 2023

[8] S. R. Bharamagoudar, R. B. Geeta, and S. G. Totad, "Web based student information management system," International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, issue 6, pp. 2342-2348, 2013.

[9] Cmak Zeelan Basha, Sofia Tasneem, Preethi Miriyala "Enhanced Technique for Placement Monitoring using ServiceNow Portal" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-1, November 2019.

[10] Navaneeth Kumar B, Vamsi Kandula, Praneeth Ambiti, K Hema, Kishore Buddha "Student Analysis System for Training and Placement" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-6, March 2020.

[11] Spoorthi M.S, Kavana v, Koushik M.N, Veena M "A review on placement management system" International Journal of Creative Research Thoughts (IJCRT) ISSN: 2320-2882, volume-9 Issue-7, July 2021.

GITHUB LINK

https://github.com/GodVilan/MERN_Based_Placement_Management_Portal

PUBLISHED PAPER

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

IJRAR.ORG E-ISSN: 2348-1269, P-ISSN: 2349-5138

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

MERN Based Placement Management Portal

[1]Nandireddy Srikanth Reddy, [2]Narla Krishna Koushik, [3]Nalumasu Sathwik, [4]Erukonda Krishnaveni

[1,2,3]UG Student, Department of CSE, CMR College of Engineering & Technology, Hyderabad, Telangana, India

[4]Assistant Professor, Department of CSE, CMR College of Engineering & Technology, Hyderabad, Telangana, India

Abstract: The MERN-based placement Management Portal is a web-based placement management system developed using the MERN stack technologies, designed to streamline and simplify the placement process for educational institutions. This tool enables placement officers, students and administrators to provide a centralized platform for effective communication and data management regarding the placements. With the MERN-based placement management portal, students can easily access and store detailed information about their skills and academic details, eliminating the need for manual paperwork and saving important time. Through the application, students can choose companies based on personal interests, skills and qualifications. Along with that, students can submit their certificates. And the important thing about our portal is that it includes a DSA Tracker where students can solve problems and check their progress. The system uses the MERN stack (MongoDB, Express.js, React.js and Node.js) to ensure a robust outcome. MongoDB serves as the database to store student information, details, certificates while Express.js and Node.js handle the backend. React.js, takes care of the front end, providing a user-friendly interface.

Index Terms **– MERN stack, DSA Tracker.**

I.INTRODUCTION

The MERN-based placement management portal is a web-based system that uses MERN technologies to simplify the placement process in educational institutions by eliminating the manual work. This portal provides a platform for seamless communication and data management between placement officers and students. Students benefit from easy access and storage of their information about their skills and academic details, eliminating the need for manual work. The portal provides an approach, allowing students to choose companies based on their personal interests, skills and qualifications that are posted by the placement officer. The feature of DSA Tracker allows students to actively improve their skills in data structures and algorithms, thereby promoting continuous development. The technologies of the MERN stack helps in the outcome, MongoDB helps in storing the student's data, Express.js and Node.js handling in the backend, and React.js framework creating a user-friendly and attractive interface for the users.

Our MERN-based placement management portal is proven to be innovative solution for educational institutions, technology to eliminate manual processes. Providing a seamless platform for communication and data management not only simplifies administrative tasks but also allows students to easily access academic and competency-related information. This software helps students to select companies that are posted by the placement officers based on their preferences provides an easy and meaningful placement experience. The feature of the DSA tracker in the portal helps in continuous skill development student's careers.
The MERN stack's architecture delivers reliable results with MongoDB protecting student data, Express.js and Node.js orchestrating the backend, and React.js creating an intuitive user interface.

II.LITERATURE SURVEY

The first paper of this paper's is "Placement support system" [1]. Placement support focuses on the automation of the placement cell. Authorizing the resumes, communicating about the varied job openings to the scholar community, managing the company relationship for inviting them for the placements, creating the location metrics, monitoring the progress of the choice process and communicating with different users. This system is often used as an application by the college to manage the student information concerning placements. Also helps companies coming for campus recruitment to ascertain student details. Before coming for the campus, companies can get information about eligible students alongside interested students.

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

The second paper is "Online Training and Placement System" [2]. This system gives a very efficient way of placement for students. In this system, the student does their registration in a very simple manner and the placement officer can easily get the information of students. The system can thus easily access the eligible students. In this system, information regarding the campus is sent to the student automatically. In overall architecture, data is stored and then as per rules and condition data is obtained and processing is applied on it such as making the report and sending mail to the student. The developed System can guarantee to keep the records safe and private which is stored in the database. It converts unstructured data into structured data and sorted format. All these are contributing to the control of the system. The TPO is the main admin of the system. Other important users of the training and placement system are student, company, and forum for which the system designed. In the existing Placement system, maximum work goes manually and is an error-prone system, takes time for any changes in the system. This big problem is the searching; sorting and updating of the student data and no any notification method available for giving information to the student except the notice board. The proposed system gets automated in the online registration all the user, activation of the user and deactivation of the user, personalization to the user, resources to be provided online, communication between the users, and gives online feedback. The admin can see the user information and will validate it, generate the student list based on company criteria; company details can be provided to the user, searching and sorting can be done, and reports to be generated. Alumni data to be maintained. Overall, all the process of the training and placement department is automated

III. METHODOLOGY

1.Technologies Used

"MERN-based placement management portal" leverages the MERN stack and includes the following technologies:
1. MongoDB: A MongoDB function can be used to read, insert, update, or delete data from your MongoDB data database. When you connect your database to Internal, the basic functions corresponding to each collection in the database are automatically created for you.
2. Express.Js: Express is a Node.js web application framework that provides extensive functionality for building web and mobile applications. It is used to create single-page, multi-page and hybrid web applications. This is a layer built on top of Node.js that helps manage servers and routes.
3. React.Js: React essentially allows developers to use individual parts of their application on both the client and server side, which ultimately increases the speed of the development process development.
Simply put, different developers can write individual parts and any changes made will not affect the application logic.
4. Node.Js: Functions in Node.js is used directly in the code. Using Node.js allows you to get the desired action from your application in a single process, i.e. you don't need to create a new thread for each new request.

STUDENT MODULE:

Every student has a unique ID given by the college and the students can login through their ID along with their password. Academic details of the students are entered by default. It contains the following:

- Profile Building: The students can add their skills and improve their candidature.
- Certifications: Provide a section for students to upload their skillset certifications.
- Resume: Upload your resume for placement officer's reference.
- Projects: Students can upload the description and link of their projects.
- Dream Company Selection: Provides a platform for students to explore job postings, view company details and apply for interested companies.
- Check Approval Status: The students can check the approval status of the dream company application.
- DSA Problem Solving: Implements a coding practice section where students can solve Data Structures and Algorithms (DSA) Problems.

ADMIN MODULE:

The Placement Officer is the admin of the portal. The admin can login through his login credentials. It contains the following:

- Job Posting: Enable the admin to post job openings on behalf of companies. Include fields for job descriptions, eligibility criteria, and other relevant information.
- Access Students Details: The admin can access student's details.
- Access Students Documents: Placement officer can access the documents that are uploaded by the students such as certificates, resume.
- View Students Company Interests: Provides a feature for the admin to view the companies that the students are interested in. This can help in coordinating and facilitating the placement process.
- View Leaderboard: Placement officer can check the leaderboard and observe the student's performance.

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

4.Implementation Block Diagram

Fig 1. Flow Diagram

IV.RESULTS AND DISCUSSION

1.Figures

Login page of the MERN based placement management portal where we can shift between student login and admin login as per the requirement.

Fig 1. Student Login Page

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Then the launch page contains all the basic academic details, skills and other details of the specific details.

Fig 2. Home Page

In the navbar , we can shift between different pages of the website such as Skills , Achievements , Certifications , Projects , Job postings and Leaderboard.

Fig 3. Skills Page

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Fig 4. Achievements

Fig 5. Certifications

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Fig 6. Projects

Fig 7. Resume Upload

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Fig 8. Job Postings

Fig 9. Problems Page

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Fig 10. Submissions Page

Fig 11. Leaderboard

Fig 12. Admin Login Page

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Fig 13. Admin Home Page

Fig 14. Student Interests

Fig 15. Add Job postings

© 2024 IJRAR February 2024, Volume 11, Issue 1 www.ijrar.org (E-ISSN 2348-1269, P- ISSN 2349-5138)

Fig 16. Comparing Student skills and Company's required skills

V.CONCLUSION

In summary, "MERN-based Placement Management Portal" is a comprehensive solution that provides students with an easy-to use the platform to efficiently manage their data, search for jobs, and seamlessly apply to their desired companies.
Incorporating a DSA tracker accelerates student's professional development by providing a dedicated space to improve their Data Structures and Algorithms (DSA) skills.
However, it is important to be aware of certain limitations, this portal is specifically designed for educational institutions and its scope is limited to the university placement process. Additionally, reliance on internet connectivity can pose challenges in areas where access is unreliable or limited.
Furthermore, regular maintenance and updates are essential to maintain efficiency and security requires dedicated resources and attention. Despite these limitations, the portal remains a valuable tool for streamlining the employment process, reducing manual workload, and generally improving students career prospects. Institutions using this platform must be aware of their unique circumstances and ensure ongoing support to maximize benefits.

REFERENCES

[1]. Santhosh Kumar H," Online Training and Placement Management system" International Journal of Engineering Research Technology (IJERT), ICACT-2016 Conference Proceedings.
[2]. Mythili M, Aishwarya R, Shenbagam P, Sandhiya C," E Placement Management", International Journal of Pure and Applied Mathematics (IJPAM), Volume 119 No. 10 2018,1823-1834.
[3]. Rajnesh Tripathi, Raghvendra Singh, Jaweria Usmani, "Campus Recruitment and Placement System", International Conference on Recent Innovations in Science and Engineering (Icrise -18), April, 2018.
[4]. Ajeena Sunny1 Aneena Felix1 Angelin Saji1 Christina Sebastian1 Praseetha V.M2 1UG Student, Dept. of CSE, SJCET, Palai, Kerala, India
[5]. Anjali v, Jeyalakshmi PR, Anbubala R,Sri Mathura devi, Ranjini.V, "Web Based Placement Management System", International Journal of Computer Science and Information Technologies(IJCSIT), Vol. 7 (2) , 2016, 760-763.
[6]. T. Gosavi, S. Gaikwad, R. Nazirkar and A. Salke, "ANDROID BASED TRAINING AND PLACEMENT AUTOMATION", National Conference-Ekalavya-2k15, ISSN 2348-6953.76SSGM Journal of Science and Engineering, Vol. 1, Issue 1, June 2023
[7]. S. R. Bharamagoudar, R. B. Geeta, and S. G. Totad, "Web based student information management system," International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, issue 6, pp. 2342-2348, 2013.
[8]. Cmak Zeelan Basha, Sofia Tasneem, Preethi Miriyala "Enhanced Technique for Placement Monitoring using ServiceNow Portal" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-1, November 2019.
[9]. Navaneeth Kumar B, Vamsi Kandula, Praneeth Ambiti, K Hema, Kishore Buddha "Student Analysis System for Training and Placement" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-6, March 2020.
[10]. Spoorthi M.S, Kavana v, Koushik M.N, Veena M "A review on placement management system" International Journal of Creative Research Thoughts (IJCRT) ISSN: 2320-2882, volume-9 Issue-7, July 2021.

PUBLICATION CERTIFICATES

Certificate of Publication

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

E-ISSN: 2348-1269, P-ISSN: 2349-5138

The Board of

International Journal of Research and Analytical Reviews (IJRAR)

Is hereby awarding this certificate to

Erukonda Krishnaveni

In recognition of the publication of the paper entitled

MERN Based Placement Management Portal

Published In IJRAR (www.ijrar.org) UGC Approved (Journal No : 43602) & 7.17 Impact Factor

Volume 11 Issue 1 February 2024, Date of Publication: 26-February-2024

PAPER ID : IJRAR24A2362

Registration ID : 283502

R.B.Joshi

EDITOR IN CHIEF

UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR

An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal

Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014

Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org

IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138

Certificate of Publication

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

E-ISSN: 2348-1269, P-ISSN: 2349-5138

The Board of

International Journal of Research and Analytical Reviews (IJRAR)

Is hereby awarding this certificate to

Nandireddy Srikanth Reddy

In recognition of the publication of the paper entitled

MERN Based Placement Management Portal

Published In IJRAR (www.ijrar.org) UGC Approved - Journal No : 45602 & 7.17 Impact Factor

Volume 11 Issue 1 February 2024, Date of Publication: 26-February-2024

PAPER ID : IJRAR24A2362

Registration ID : 283502

R.B.Joshi

EDITOR IN CHIEF

UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR

An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal

Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014

Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org

IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138

Certificate of Publication

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

E-ISSN: 2348-1269, P-ISSN: 2349-5138

The Board of

International Journal of Research and Analytical Reviews (IJRAR)

Is hereby awarding this certificate to

Narla Krishna Koushik

In recognition of the publication of the paper entitled

MERN Based Placement Management Portal

Published In IJRAR (www.ijrar.org) UGC Approved (Journal No : 43602) & 7.17 Impact Factor

Volume II Issue 1 February 2024, Date of Publication: 26-February-2024

PAPER ID : IJRAR24A2362

Registration ID : 283502

R.B.Joshi

EDITOR IN CHIEF

UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR

An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal

Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014

Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org

IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138

Certificate of Publication

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

E-ISSN: 2348-1269, P-ISSN: 2349-5138

The Board of

International Journal of Research and Analytical Reviews (IJRAR)

Is hereby awarding this certificate to

Nalumasu Sathwik

In recognition of the publication of the paper entitled

MERN Based Placement Management Portal

Published in IJRAR (www.ijrar.org) UGC Approved (Journal No : 43602) & 7.17 Impact Factor

Volume II Issue I February 2024, Date of Publication: 26-February-2024

PAPER ID : IJRAR24A2362

Registration ID : 283502

R.B.Joshi

EDITOR IN CHIEF

UGC and ISSN Approved - Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.17 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS | IJRAR

An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal

Website: www.ijrar.org | Email: editor@ijrar.org | ESTD: 2014

Manage By: IJPUBLICATION Website: www.ijrar.org | Email ID: editor@ijrar.org

IJRAR | E-ISSN: 2348-1269, P-ISSN: 2349-5138