COLLEGE CODE:[8223]

COLLEGE NAME: [vandayar engineering college]

DEPARTMENT:[Computer Science And Engineering]

STUDENT NM-ID:[47CA18F7582CBFE85C64213B41E7C205]

ROLL NO:[822323104001]

DATE: [26.09.2025]

Completed the project named as

Phase-3

TECHNOLOGY PROJECT NAME:STUDENT GRADING SYSTEM

SUBMITTED BY,

NAME: [Akalya.S]

MOBILE NO:[6380809485]

1.**Project Setup**

Repository Creation

A GitHub repository is created to maintain the project code.

Proper naming conventions are followed (example: student-grading-system).

License and README files are initialized to give clarity about the project.

Folder Structure

Frontend: React components, styles, assets.

Backend: Node.js/Express APIs, middleware, business logic.

Database: Schemas, migrations, seed files.

Config: Environment variables and secret keys.

Tools & Dependencies

Node.js, Express.js – Backend API.

React.js – Frontend development.

MongoDB/MySQL – Database.

Postman – API testing.

GitHub – Version control.

Example:

```
/project
  /backend
  /frontend
  /database
  README.md
```

2.Core Feature Implementation

Authentication Module

Login/Signup with roles (Student, Teacher, Admin).

JWT (JSON Web Token) for secure authentication.

Password hashing using bcrypt for security.

Marks Entry

Teachers upload marks through dashboard.

Bulk upload option (Excel/CSV file).

Validation: Ensures no duplicate entries.

Grade Calculation

Predefined rules (e.g., A: 90–100, B: 75–89).

GPA/CGPA calculator.

Automated percentage-to-grade conversion.

Student Dashboard

Shows marks for each subject.

Overall GPA/CGPA chart (pie or bar graph).

Downloadable PDF report card.

Admin Panel

Add/update/remove users.

Define subjects, grading rules, and academic year.

Monitor overall system performance.

3.DataStorage(LocalState/Data base)

Local State Management

Frontend uses React Context API or Redux.

Example: Student grades stored temporarily in state before saving to database.

Database Schema

User Table: Stores student/teacher/admin info.

Marks Table: Stores subject marks.

Grades Table: Stores calculated results.

Sample Schema (MySQL):

Users (id, name, email, role, password)

Marks (id, student_id, subject_id, marks)

Subjects (id, name, max_marks)

Grades (id, student_id, grade, GPA)

Security Measures

Input validation before saving.

Encryption for sensitive data.

Restricted access based on user roles.

4.Testing Core Features

Unit Testing

Each module tested individually.

Example: Check if login API returns correct response.

Integration Testing

Ensures different modules work together.

Example: Teacher uploads marks → Student views grade.

Manual Testing

Test cases executed manually by team.

Example: Invalid login credentials, missing fields in form.

Bug Fixing

Errors logged and fixed before deployment.

Continuous testing ensures system stability.

Version Control (GitHub)

Branching Strategy

Main Branch – Stable code only.

Feature Branches – New features developed separately.

Bugfix Branches – For fixing issues.

Commit Messages

Follow proper format:

Feat: add login API

Fix: resolve marks entry bug

Docs: update README

Collaboration

Multiple developers push code into GitHub.

Pull Requests reviewed before merging.

GitHub Issues used for task tracking.

5.Continuous Integration

GitHub Actions configured for automated testing.

Every push triggers tests to ensure nothing breaks.

Example Workflow (End-to-End)

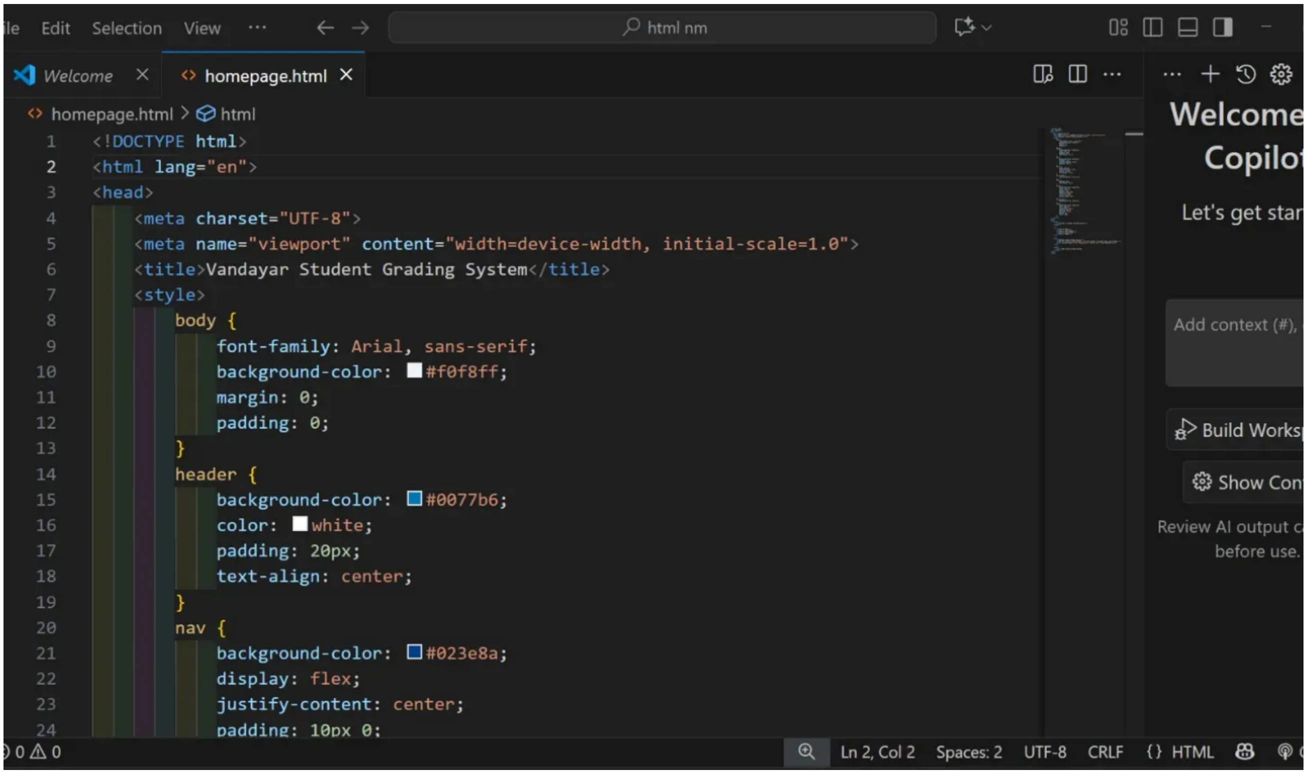
1. Teacher logs in and enters marks for students.

2. System stores marks in the database.

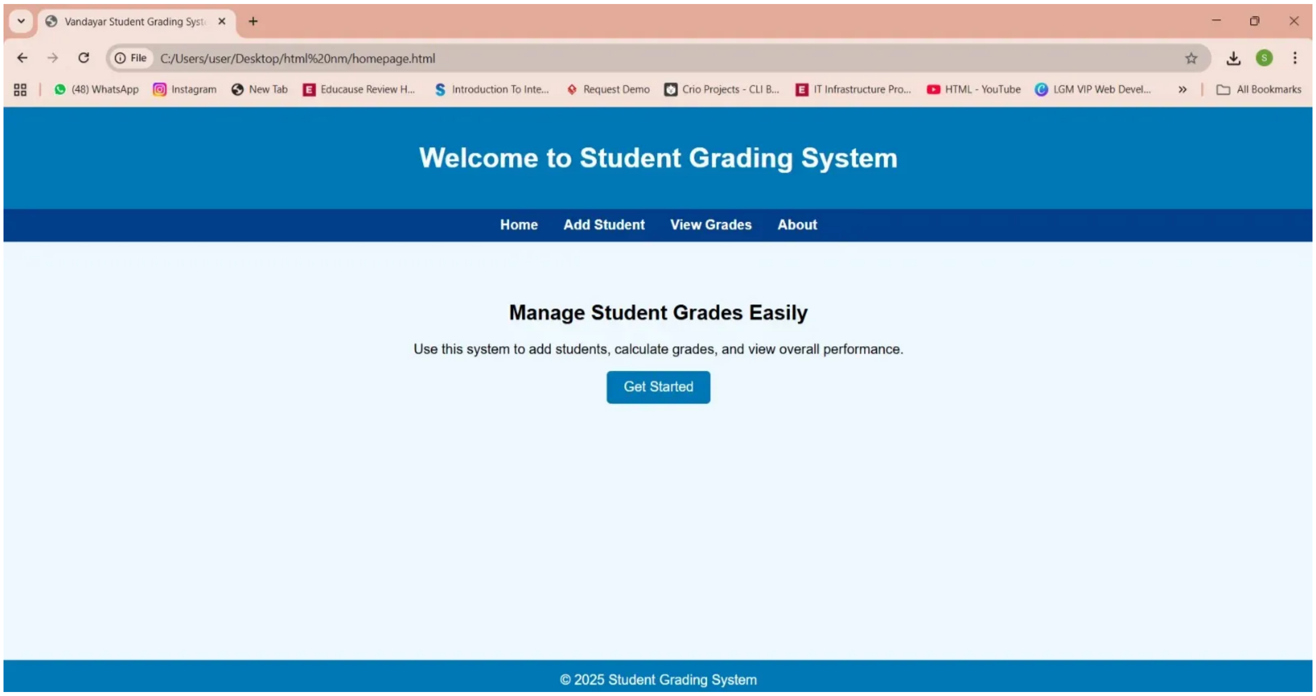
3. Grade calculation service automatically computes grades.

4. Students log in to their dashboard and view updated grades.

5. Admin monitors performance and generates reports.



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Vandayar Student Grading System</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f0f8ff;
            margin: 0;
            padding: 0;
        }
        header {
            background-color: #0077b6;
            color: white;
            padding: 20px;
            text-align: center;
        }
        nav {
            background-color: #023e8a;
            display: flex;
            justify-content: center;
            padding: 10px 0;
```



Welcome to Student Grading System

Manage Student Grades Easily

Use this system to add students, calculate grades, and view overall performance.

© 2025 Student Grading System