Project Plan

1. Project Overview

Project Title:
Teacher's Gradebook and Report Card System

Problem Summary:
Teachers often face challenges in managing student grades efficiently, calculating averages, and generating report cards. Manual processes are time-consuming and prone to errors. This project aims to develop a digital gradebook to streamline these tasks, reducing workload and improving accuracy.

Intended Audience/Users:
Primary and secondary school teachers.
Educational institutions looking for a simple and efficient grading solution.

Main Features/Components:
Record and calculate grades for students.
Generate report cards based on student performance.
Support for calculating weighted averages for assignments, exams, and projects.

2. Project Objectives

- Develop a user-friendly digital gradebook with a clear interface using JavaFX.
- Implement a robust database system with MariaDB to securely store student and grade data.
- Enable automated calculation of averages and weighted scores.
- Provide an option for teachers to generate customizable report cards.

3. Scope and Deliverables

Scope (Inclusions):
- Grade recording functionality.
- Automated calculation of averages and weighted scores.
- Report card generation.
- User-friendly interface with basic CRUD operations.

Scope (Exclusions):
- Integration with external educational platforms.
- Real-time collaboration or cloud-based features.

Key Deliverables:
- Functional Gradebook Application: A working JavaFX application integrated with MariaDB.

- Database Design: A well-structured MariaDB database schema.
- Report Card Template: A functional feature to generate printable report cards.
- Technical Documentation: Detailed documentation for system use and maintenance.

4. Project Timeline

Sprint	Start Date	End Date	Key Objectives and Milestones
Sprint 1	2025-01-15	2025-01-29	- Define project requirements and features (Product Backlog). - Complete project architecture design (database schema and UI mockups). - Set up the development environment and create the basic code framework.
Sprint 2	2025-01-29	2025-02-12	- Implement grade recording functionality (data input and storage). - Develop the basic grade display interface. - Integrate the database and perform initial testing.
Sprint 3	2025-02-12	2025-03-05	- Refine the grade calculation module (averages and weighted calculations). - Add the report card generation feature. - Conduct unit testing and bug fixing.
Sprint 4	2025-03-05	2025-03-14	- Optimize the user interface and user experience. - Conduct comprehensive testing (integration and user acceptance testing). - Write the user manual and technical documentation. - Finalize the project for submission.

5. Resource Allocation

Team Members and Roles:

- **Jia Ke** (Database and Frontend Specialist, Scrum Master for Sprint 1 and 4):

Responsibilities:

Facilitate the Scrum process during Sprint 1 and 4, ensuring smooth planning and task coordination.

Lead the establishment of the project framework, including setting up the database schema (MariaDB) and designing the initial UI (JavaFX).

Contribute to frontend development and ensure seamless integration with the backend.

Oversee project finalization, including report card optimization and submission readiness in Sprint 4.

- **Wang Qingyun** (Backend Developer and Documentation Specialist, Scrum Master for Sprint 2):

Responsibilities:
Facilitate the Scrum process during Sprint 2, ensuring effective sprint planning and backlog management.
Develop backend logic, focusing on core features such as grade calculation and weighted averages.
Integrate backend functionality with the database and UI.
Focus on project documentation, including technical documentation and user manuals, throughout all sprints.

- **Yang Yang** (Programmer and Debugging Specialist, Scrum Master for Sprint 3):

Responsibilities:
Facilitate the Scrum process during Sprint 3, coordinating testing and debugging efforts.
Contribute to both frontend and backend development during all sprints, refining and completing features as needed.
Lead testing efforts (unit testing, integration testing) in Sprint 3, identifying and resolving bugs.
Assist in final testing and feedback gathering during the project closure phase.

Software, Hardware, and Tools:

- Frontend Development: JavaFX.
- Backend Development: Java.
- Database Management: MariaDB.
- IDE: IntelliJ IDEA.
- Testing Tools: JUnit for unit testing.

External Resources or Support:
Online resources and tutorials for JavaFX and MariaDB.

6. Risk Management

Risk Description	Likelihood	Impact	Mitigation Strategy
UI/UX Design Not User-Friendly	Low	Medium	Seek feedback from potential users (teachers) during development and iterate based on their input. Use UI design guidelines.
Feature Scope Creep	Medium	High	Stick to the original requirements document. Any new features must be reviewed and approved by the team.
Integration Problems Between Frontend and Backend	Medium	High	Use clear API contracts between frontend and backend. Test integration early in the development phase.
Critical Bugs or Errors Near Deadline	High	High	Schedule a buffer period in Sprint 4 to address last-minute issues. Prioritize critical bugs during debugging.
Version Control Conflicts	Medium	Medium	Use clear Git workflows (e.g., branching strategy). Regularly merge changes to avoid conflicts.

7. Testing and Quality Assurance

Testing Types:
Unit testing for individual components.
Integration testing for database and application interaction.
User acceptance testing with mock data.
Success Criteria:
All features function as intended without critical bugs.
User-friendly and intuitive interface.
Database operates efficiently under various scenarios.
Tools/Frameworks:
JUnit for testing Java code.
MariaDB tools for database testing.

8. Documentation and Reporting

Planned Documentation:
User Manual: Instructions for using the system.
Technical Documentation: Details of system architecture and database schema.
Reporting Progress:
Weekly updates with screenshots or progress notes.
Final report summarizing achievements and lessons learned.