### **Project Overview**

This document outlines the scope, stakeholders, and plan for the **School Management System**, a capstone **project for Group 3 in** course CS492. The goal is to develop a functional software product that streamlines school administration by providing a centralized platform for managing student and teacher data, academic performance, and classroom logistics. The project will follow an Agile methodology, with development organized into two distinct sprints.

**Project Scope**

The system will provide core functionalities for three main user roles: Administrators, Teachers, and Students.

* **In-Scope Features:**
  + **User Management:** The ability for administrators to create, edit, and delete student and teacher profiles.
  + **Academic Administration:** The assignment of teachers to classrooms and the allocation of students to those classes, including rules for capacity.
  + **Performance Management:** The ability for teachers to enter student grades and for students to view their own academic results.
  + **Role-Based Access:** Secure user authentication and dedicated dashboards for administrators, teachers, and students, ensuring users only see relevant information and tools.
* **Out-of-Scope Features (Outside of the MVP):**
  + Parent/Guardian portal
  + School billing and finance modules
  + Library management system
  + Event scheduling and calendar system
  + Attendance tracking

### **Project Team & Stakeholders**

The project team is composed of the following members, fulfilling key roles within the Scrum framework:

* **Product Owner:** Miguel
* **Scrum Master:** Brian Keener
* **Development Team:** Nadia, Miguel, Robert, Brian

### **Product Backlog**

The following table represents the full product backlog, containing all user stories prioritized for development across the project's Sprints.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *E1* | *Manage Student Records* | *Add/Edit/Delete student profiles* | *Admin can manage student records* | *High* | *1* |
| *E2* | *Academic Performance Management* | *Enter and view student grades* | *Teachers can submit grades and students can view them* | *High* | *1* |
| *E3* | *Manage Teacher Profiles* | *Add/Edit teacher details* | *Admin can update teacher info* | *Medium* | *1* |
| *E4* | *Assign Teachers to Classrooms* | *Map teachers to subjects/classrooms* | *Admin can assign teachers to specific classes* | *High* | *1* |
| *E5* | *Allocate Classrooms* | *Assign class IDs and manage capacity* | *Admin can assign students/teachers to rooms* | *Medium* | *1* |
| *E6* | *Authentication and Roles* | *Secure login and role-based dashboards* | *Users log in and access correct dashboard* | *High* | *2* |
| *E7* | *Teacher Dashboard* | *View assigned classes, enter grades* | *Teachers can see class list and submit results* | *High* | *2* |
| *E8* | *Student Dashboard* | *View personal info and grades* | *Students can access profile and grades* | *High* | *2* |
| *E9* | *Class Capacity Rule* | *Limit classes to 20 students* | *System prevents adding more than 20 students* | *Medium* | *2* |
| *E10* | *View Student Performance* | *Teachers view student academic history* | *Teachers see correct and updated grades* | *Medium* | *2* |
| *E11* | *Admin Dashboard* | *View all teachers/students with full creation, read, update and deletion access* | *Admins designed user interface with full access* | *High* | *2* |

### **Sprint Plan Overview**

Development will be executed over two sprints, each lasting two weeks. The work is divided as follows based on the product backlog:

* **Sprint 1 (Weeks 1-2): Foundational Features**
  + Focus on establishing core data models and administrative capabilities. This includes managing student and teacher records, grade entry, and classroom assignments.
  + **Stories:** E1, E2, E3, E4, E5.
* **Sprint 1 Goal:** Build core data management for students, teachers, and class logistics.
* **Definition of Done**: Code is committed, reviewed, and tested; meets acceptance criteria; includes unit/integration tests; is documented; and is deployed to a test environment if applicable.
* **Sprint 2 (Weeks 3-4): Dashboards & User Experience**
  + Focus on building the user-facing components, including secure login and role-specific dashboards. This sprint will also implement key business rules like class capacity.
  + **Stories:** E6, E7, E8, E9, E10, E11.