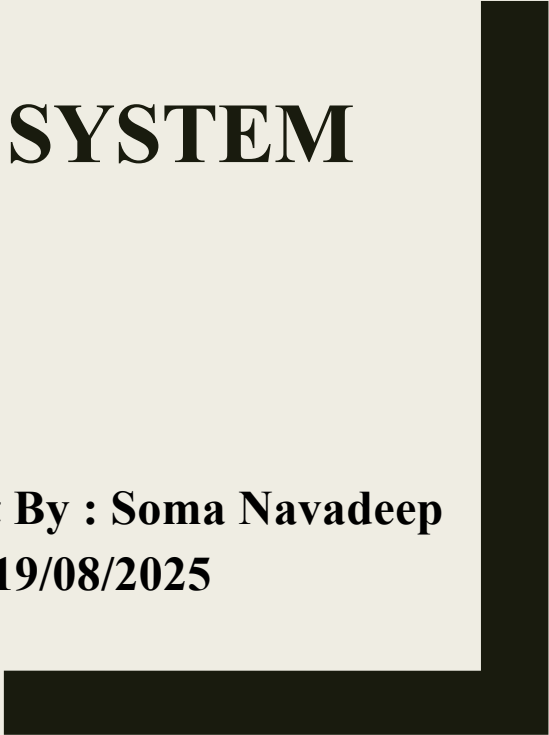




SCHOOL MANAGEMENT SYSTEM

Project By : Soma Navadeep

Date : 19/08/2025



Situation :

- Schools are currently managing records (students, fees, attendance, exams) manually or using outdated systems.
- Communication between teachers, parents, and students is not smooth.
- Lots of paperwork and time is wasted in routine school operations.
- Different departments (admin, accounts, academics) are not connected properly.
- No proper data storage or backup system in case records are lost.

Problem :

- Manual processes cause errors (wrong attendance, missing fee records).
- Parents don't get updates on time about student performance.
- Teachers spend more time on admin work than on teaching.
- Delay in generating reports like exam results, fee dues, or attendance sheets.
- Data security issues – records can be misplaced or misused.

Opportunity :

- Build a School Management System to digitalize and automate school processes.
- Save time and improve accuracy in operations like attendance, fees, and report cards.
- Allow future features like online learning, digital exams, and mobile apps.
- Better data security, backup, and faster report generation.
- Helps schools scale easily as number of students/staff grows.
- Allow future features like online learning, digital exams, and mobile apps.

Purpose Statement :

The purpose of this project is to develop a **School Management System** that helps schools manage their daily activities in an easy and organized way. It will reduce paperwork, save time, improve communication between teachers, parents, and students, and make processes like attendance, exams, and fee management more accurate and faster.

Key issues identified

- **Manual Workload** – Too much paperwork for attendance, marks, and fees, leading to errors.
- **Poor Communication** – Parents, teachers, and students are not always updated on time.
- **Limited Accessibility** – Information is only available in school offices, not online.
- **Data Inaccuracy** – Records of students, exams, and fees often have mistakes.

Project Objectives:

1. Centralize Student Information

Maintain all student data (personal details, academics, attendance, fees, etc.) in one system instead of scattered registers or files.

2. Automate Administrative Tasks

Reduce manual efforts for teachers and staff by automating tasks like attendance marking, timetable generation, and report cards.

3. Ensure Data Security & Privacy

Protect sensitive student and staff information with role-based access and secure authentication.

4. Enhance Parent-Teacher Communication

Provide parents with instant access to student performance, attendance, and school notices via mobile/web app.

5. Enable Anywhere Access

Provide cloud-based access so students, teachers, and parents can log in anytime, anywhere.

6. Support Online Learning

Integrate features like online classes, assignments, and exam submissions to support hybrid/remote education.

7. Follow school rules and guidelines

Make sure the system meets education policies and data rules.

8. Faster Deployment & Continuous Improvement

Build and release features step by step using Agile, so feedback can be added fast.

Success Criteria:

1. Easy Access to Records

All student, teacher, and school records should be available in one place and easy to find anytime.

2. Faster System Performance

The system should work smoothly with less downtime, faster loading, and quick responses.

3. Better Communication

Parents, teachers, and students can easily share information and updates without delays.

4. Accurate Reports

The system should generate correct and useful reports for exams, fees, and attendance.

5. User-Friendly Design

Teachers, students, and parents should be able to use the system without much training.

Agile Methods/Approach:

Agile is a way of doing projects where work is done in small steps (sprints). After each sprint, we deliver a working part of the system and take feedback. This makes the project flexible and easy to improve.

1. Set Vision & Goals

Goal: Build a system to manage admissions, attendance, exams, fees, communication, and reports.

Success = reduce paperwork, improve communication, easy access to student/teacher data.

2. The Team

Product Owner (School Principal/Management) – decides priorities.

Scrum Master (BA/PM) – ensures agile process is followed.

Development Team – developers, testers, designers.

3. Create the Product Backlog

“As a teacher, I want to mark attendance online so I don’t have to use paper registers.”

“As a parent, I want to check my child’s exam results online for quick access.”

4. Prioritize the Backlog

Must-have features first: Student Admission, Attendance, Exams, Fee Payment.

Later features: Notifications, Reports, Mobile App.

Team gives story points (1–5) for each feature depending on complexity.

5. Sprint setup & Sprint Planning (Start of each sprint)

Select top stories (e.g., Login, Student Admission, Attendance).

6. Build & Test (During sprint)

Developers code → Testers verify features → BA checks acceptance criteria.

Example: Attendance system works for one class.

7. Daily Stand-ups and Sprint reviews (demo)

Show new features to school stakeholders.

Example : Demonstrate online attendance system to Principal.

8. Sprint Retrospective

Team discusses improvements

9. Go live

After MVP is ready (Admissions + Attendance + Exams + Fees), deploy for real use.

10. Handle Change Requests

Example: Principal requests a new “SMS alert for holidays.”

Added to backlog, prioritized for a future sprint.

Resources:

1. People (Team Members)

- Project Manager / Scrum Master – 1
- Business Analyst (BA) – 1
- Developers – 3 - 4
- Testers / QA Engineers – 2
- UI/UX Designer – 1
- Technical Support Staff – 1
- Stakeholder - School Representatives (Teachers/Admins for feedback) – 2

2. Timeline

Requirement Gathering & Planning – 2 weeks

Design & Prototyping – 2 weeks

Development (in sprints) – 3 to 4 months

Testing & UAT – 1 month

Deployment & Training – 2 weeks

Total Estimated Duration – 5 to 6 months

3. Budget

Estimated budget components

Development costs : Salaries

Infrastructure costs : Services, backup systems

Licensing costs : Software licenses

Maintenance and support : 1-2 months of post development support

Rough Budget 30 lakhs INR

4. Other Resources

Hardware - Dedicated servers for hosting the application and database

Backup and disaster recovery systems

Software – Development tools

Network - High speed internet connections

Risk and dependencies

1. Changing Requirements

School may keep asking for new features, which can delay the project.

2. Technical Issues

Server or software problems may affect performance.

3. Low User Adoption

Teachers or staff may resist using the new system.

4. Time Delays

If sprints are not completed on time, project delivery may get late.

5. Budget Overrun

If extra features or changes are added, costs may increase.

Dependencies

1. Training & Support

System success depends on proper training given to staff.

2. Management Approval

Project progress depends on quick decisions from school management.

3. School Staff Availability

Need teachers/admins for feedback in every sprint.

4. Third-party Services

Depend on payment gateways, SMS, or email services for integration.

5. Technology Infrastructure

Reliable internet and devices are needed for smooth working.



Thank you