**A person holding books and smiling

AI-generated content may be incorrect.**

**A white card with black text

AI-generated content may be incorrect.**

**ABSTRACT**

The Student Tracking System provides a robust solution for managing and monitoring student performance in educational institutions. It enables teachers and administrators to track academic progress, while also allowing parents to follow their children's performance in real-time. The system features student data management, attendance tracking, grade monitoring, task and exam management, analytical reporting, and notifications. Through automation and centralized data, the platform improves efficiency, reduces errors, and ensures a seamless academic monitoring process.

**ACKNOWLEDGMENTS**

I would like to express my deepest gratitude to my supervisor, [Supervisor Name], for their invaluable guidance and continuous support throughout the development of this project. Their insights have been instrumental in shaping the direction and quality of this work. I also extend my sincere thanks to my family and friends for their encouragement and support.

**TABLE OF CONTENTS**

1. **INTRODUCTION**
   * 1.1 Background
   * 1.2 Motivation
   * 1.3 Problem Statement and Suggested Solution
   * 1.4 Aims and Objectives
   * 1.5 Scope of the Project
   * 1.6 Target Audience
   * 1.7 Constraints and Limitations
2. **LITERATURE REVIEW**
   * 2.1 Overview
   * 2.2 Similar Systems Analysis
   * 2.3 Summary
3. **REQUIREMENTS AND ANALYSIS**
   * 3.1 Stakeholder Analysis
   * 3.2 User Stories & Use Cases
   * 3.3 Functional Requirements
   * 3.4 Non-Functional Requirements
   * 3.5 Software & Hardware Requirements
4. **SYSTEM ANALYSIS & DESIGN**
   * 4.1 Problem Statement & Objectives
   * 4.2 Use Case Diagram & Descriptions
   * 4.3 Functional & Non-Functional Requirements
   * 4.4 Software Architecture
   * 4.5 Database Design & Data Modeling
   * 4.6 Data Flow & System Behavior
   * 4.7 UI/UX Design & Prototyping
   * 4.8 System Deployment & Integration
   * 4.9 Additional Deliverables
5. **IMPLEMENTATION & TESTING**
   * 5.1 Source Code Management
   * 5.2 Testing & Validation
   * 5.3 Security Considerations
6. **RESULTS & DISCUSSION**
   * 6.1 Performance Evaluation
   * 6.2 Challenges & Solutions
   * 6.3 Future Enhancements
7. **CONCLUSION & RECOMMENDATIONS**
8. **REFERENCES**

CHAPTER 1:

INTRODUCTION

**1.1 BACKGROUND**

Educational institutions often face challenges in tracking student performance efficiently. Traditional methods rely on manual data entry, which can be time-consuming and prone to errors. The lack of a centralized system results in delays in monitoring student progress, affecting both educators and parents. To address these issues, the Student Tracking System provides a modern, automated approach to academic monitoring.

**1.2 MOTIVATION**

The need for a reliable and efficient student tracking system arises from the increasing demand for digital solutions in education. Many institutions struggle with fragmented systems that do not provide real-time insights. The goal of this project is to enhance transparency, streamline communication between stakeholders, and improve academic performance tracking.

**1.3 PROBLEM STATEMENT & SUGGESTED SOLUTION**

**Problem Statement**

Manual tracking methods in schools often result in inefficiencies, miscommunication, and delays in identifying struggling students. Additionally, the absence of automated reports makes it difficult for parents to stay updated on their children's academic progress.

**Suggested Solution**

The Student Tracking System is designed to automate student monitoring processes, providing real-time updates on attendance, grades, and assignments. The platform ensures efficient data management and facilitates seamless communication between teachers, students, and parents.

**1.4 AIMS & OBJECTIVES**

* Develop a centralized platform for academic performance tracking.
* Automate attendance and grade management.
* Provide real-time notifications to students and parents.
* Generate analytical reports for teachers and administrators.
* Ensure data security and access control.

**1.5 SCOPE OF THE PROJECT**

The system covers the following features:

* Student registration and profile management.
* Attendance tracking with automated reports.
* Grade entry, report generation, and performance comparison.
* Task and exam scheduling with notifications.
* Real-time alerts and messaging between stakeholders.

**1.6 TARGET AUDIENCE**

* **School Administrators**: Monitor overall performance and manage users.
* **Teachers**: Input grades, track attendance, and assign tasks.
* **Students**: View academic records and assignments.
* **Parents**: Receive updates and track student progress.

**1.7 CONSTRAINTS & LIMITATIONS**

* System availability may be affected by server downtime.
* Users require internet access to use the platform.
* Data privacy must be ensured to protect student information.

CHAPTER 2:

LITERATURE REVIEW

The Literature Review chapter provides an overview of existing research, systems, and studies related to student tracking systems. This chapter helps in understanding the current challenges and how the proposed system will improve upon them.

**2.1 OVERVIEW**

This section reviews existing **academic tracking systems** and their **limitations**. The goal is to highlight the need for a more **automated, efficient, and user-friendly solution**.

**Key Points Covered:**

* How traditional student tracking works.
* Challenges in manual or semi-automated tracking.
* The importance of improving efficiency through technology.

**2.2 SIMILAR SYSTEMS ANALYSIS**

This section compares existing student tracking solutions with the proposed system. The comparison focuses on:

1. Attendance Management – How different systems handle student attendance.
2. Grade Tracking – The level of analytics and reporting provided.
3. Notifications – How systems communicate updates to students, teachers, and parents.

**The goal is to identify gaps in existing systems and show how the proposed system addresses these challenges.**

| **Feature** | **PowerSchool (SIS - Existing System 1)** | **Google Classroom (LMS - Existing System 2)** | **Student Tracking System (Proposed)** |
| --- | --- | --- | --- |
| **Attendance Management** | **Manual Entry (Teachers record attendance manually)** | **Semi-Automated (Teachers mark attendance online)** | **Digital Entry (Teachers input attendance via system, reducing paperwork)** |
| **Grade Tracking** | **Limited (Only stores grades, no insights)** | **Basic Analytics (Simple grading and tracking)** | **Standard Reports (Displays student performance and trends)** |
| **Notifications** | **No (No built-in notifications)** | **Limited (Assignment reminders only)** | **Direct Alerts (Sends notifications about attendance and grades)** |

**2.3 CHALLENGES IN EXISTING SYSTEMS**

This section discusses limitations found in other student tracking solutions, such as:

* **Manual Data Entry → Time-consuming and prone to errors.**
* **Limited Communication → Lack of instant notifications for students and parents.**
* **Basic Reporting → No advanced insights into student performance.**

**2.4 SUMMARY**

Existing systems lack automation, real-time reporting, and seamless communication features. The proposed system aims to fill these gaps.

CHAPTER 3:

REQUIREMENTS & ANALYSIS

**3.1 STAKEHOLDER ANALYSIS**

Identifies key users and their needs.

**3.2 USER STORIES & USE CASES**

Provides interaction scenarios for different user roles.

**3.3 FUNCTIONAL REQUIREMENTS**

* Student data management
* Attendance tracking
* Grade management
* Task scheduling
* Real-time notifications

**3.4 NON-FUNCTIONAL REQUIREMENTS**

* Secure authentication and authorization
* High system availability
* Responsive UI for all devices

**3.5 SOFTWARE & HARDWARE REQUIREMENTS**

* **Backend**: ASP.NET Core
* **Frontend**: html, CSS, Java Script, Bootstrap
* **Database**: SQL Server
* **Hardware**: Cloud-based or on-premises server

**3.6.0 USE CASE DIAGRAM**

Shows the interactions between users (Student, Teacher, Parent, Admin) and the system, such as receiving notifications, viewing reports, and managing users

A diagram of a person's workflow

AI-generated content may be incorrect.

USE CASE

|  |
| --- |
|  |

|  |
| --- |
| Actors:   * **Student**: Views grades, attendance, and notifications. * **Parent**: Monitors student progress and receives notifications. * **Teacher**: Marks attendance, enters grades, and tracks student performance. * **Admin**: Manages users, moderates system access, and generates reports.   Use Cases:   * **Register/Login**: Allows users (students, teachers, parents, and admins) to access the system. * **View Grades**: Students and parents can check grades. * **Enter Grades**: Teachers record student grades. * **Track Attendance**: Teachers mark attendance, and students/parents can view it. * **Receive Notifications**: Users receive alerts about important updates. * **Manage Users**: Admins add, edit, or remove users |

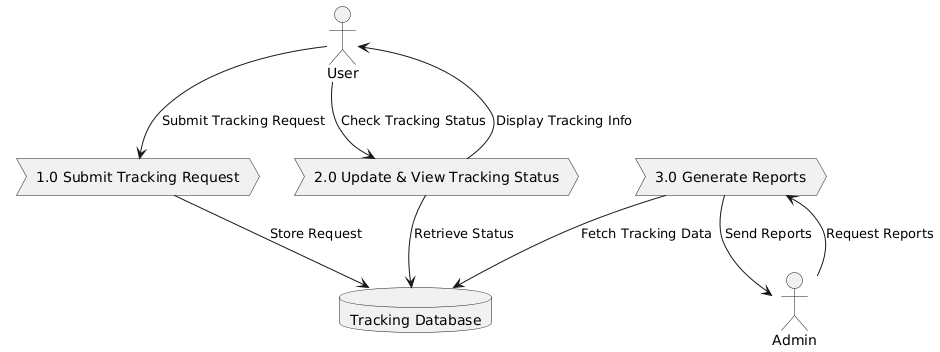
**3.6.1 UML (Unified Modeling Language) Model Diagrams**

**3.6.2 ERD DIAGRAMS**

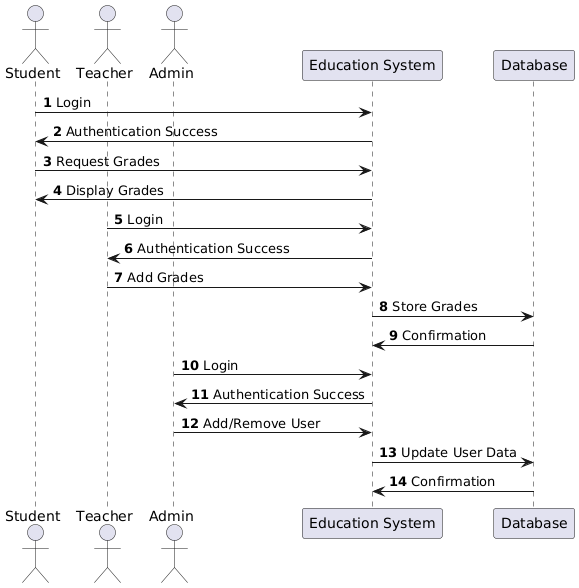
A diagram of a company

AI-generated content may be incorrect.

**3.6.3 DATAFLOW**

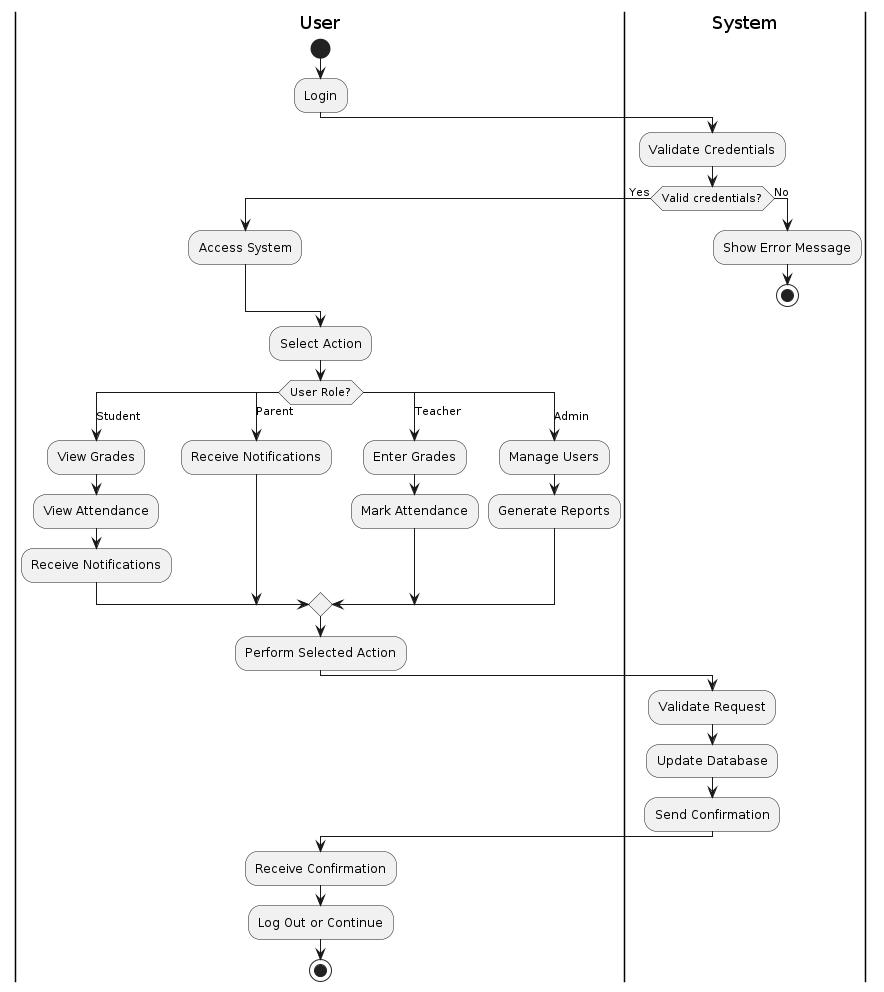


**3.6.4 SEQUENCE DIAGRAM**



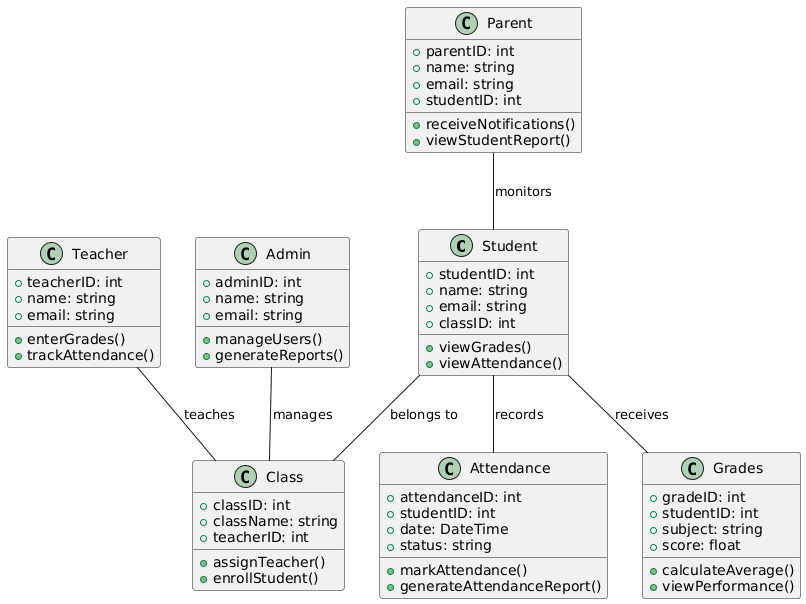
Shows the flow of messages between objects in a specific scenario, such as how a student or parent retrieves grade information from the system

**3.6.5 ACTIVITY DIAGRAM**



Class

**3.6.6 CLASS DIAGRAM**



Displays the core entities of the system (such as Student, Teacher, Class, Grades) and their relationships, including inheritance and associations.