# Documentation for Student Management System

## 1. Overview

The \*\*Student Management System\*\* is an F# application developed to simplify the management of student records. It employs an SQLite database to store essential information such as student IDs, names, and grades. The application features a user-friendly graphical user interface (GUI) for seamless interaction.

## 2. Key Features

1. \*\*Database Management\*\*:  
 - Uses SQLite database (`students.db`) for persistent storage of student records.  
  
2. \*\*Student Records\*\*:  
 - Allows users to \*\*add\*\*, \*\*edit\*\*, and \*\*delete\*\* student records.  
  
3. \*\*Grade Management\*\*:  
 - Calculates individual student averages.  
 - Provides class-wide statistics, including:  
 - Pass/fail rates.  
 - Highest and lowest grades.  
  
4. \*\*User Roles\*\*:  
 - \*\*Admin Role\*\*: Full control over database operations, including adding and modifying student records.  
 - \*\*Viewer Role\*\*: Read-only access to view student records and statistical reports.

## 3. How It Works

1. \*\*Database Initialization\*\*:  
 - Upon launch, the application checks for the existence of the `students.db` file.  
 - If absent, the application creates the database and initializes a `Students` table.  
  
2. \*\*Adding Students\*\*:  
 - Admin users input the name and grades of students via the GUI.  
 - The data is stored in the `students.db` database.  
  
3. \*\*Viewing Records\*\*:  
 - All student records are displayed in a readable format.  
 - Includes detailed statistics on class performance.  
  
4. \*\*Grade Calculations\*\*:  
 - Automatically calculates:  
 - Average grades for individual students.  
 - Class-wide metrics, including:  
 - Pass/fail statistics.  
 - Highest and lowest grades.

## 4. Application Workflow

1. \*\*Startup\*\*:  
 - Initialize database.  
 - Load existing records into the GUI.  
  
2. \*\*Admin Operations\*\*:  
 - Add, edit, or delete student records.  
 - View detailed statistical reports.  
  
3. \*\*Viewer Operations\*\*:  
 - Read-only access to records and reports.  
  
4. \*\*Data Updates\*\*:  
 - Modifications are reflected instantly in the database.  
 - Statistics recalculated dynamically.

## 5. Benefits

- \*\*Efficiency\*\*: Simplifies student data management for educators and institutions.  
- \*\*Accuracy\*\*: Ensures data is up-to-date with instant updates and calculations.  
- \*\*Security\*\*: Role-based access control protects sensitive data.  
- \*\*Insights\*\*: Provides valuable performance insights through automated statistics.

## 6. Future Enhancements

- Integration with cloud databases for scalability.  
- Support for exporting reports in formats like CSV or PDF.  
- Enhanced statistical analysis tools (e.g., performance trends over time).  
- Multi-language support for broader accessibility.