Advanced Student Management System

Functionalities:

1. Authentication and Authorization:
    - Implement login functionality for administrators and teachers with different roles and access levels.
    - Administrators can perform all CRUD operations on students and manage users.
    - Teachers can view student details and update attendance.

2. Student Profile Management:
    - Allow students to create their profiles with details like name, roll number, address, contact information, etc.
    - Enable students to update their profiles.

3. Course Management:
    - Implement a course management system where administrators can add, delete, or update course details.
    - Assign students to courses and vice versa.
    - Allow teachers to view courses they are assigned to.

4. Attendance Tracking:
    - Enable teachers to mark attendance for students in each course.
    - Generate attendance reports for administrators and teachers.

5. Grading System:
    - Implement a grading system where teachers can assign grades to students for assignments, quizzes, and exams.
    - Calculate overall grades based on weighted averages.
    - Allow students to view their grades.

6. Communication Module (bonus)
    - Implement a messaging system where administrators, teachers, and students can communicate.
    - Allow file attachments and group messaging for courses.

7. Data Export and Backup:
    - Provide functionality to export student data, course details, attendance records, etc., to CSV or Excel files.
    - Implement automated backup functionality to prevent data loss.(bonus)

Advanced

- User Interface: Develop a user-friendly GUI using JavaFX or Swing for better user interaction.
- Reports Generation: Implement a reporting engine to generate various reports like student profiles, attendance summaries, grade reports, etc.
- Integration with External Systems: Integrate with external systems like email services for notifications, payment gateways for fee transactions, etc.
- Use database management systems like MySQL or PostgreSQL for persistent storage.
- Utilize encryption techniques to secure sensitive data like passwords and personal information.
- Implement logging and auditing to track system activities and changes made by users.