ACM 321 Fall 2024: Object-Oriented Programming Project Assignment

Project Title: Inventory Management System

Project Overview:

Your team will develop an **Inventory Management System (IMS)** for a small retail store. The IMS will allow the store to manage its inventory, sales, and suppliers through a user-friendly graphical interface. The application must meet the following requirements:

Notes about team formation: Teams must be composed of at **least 2** and **at most 4** students. If you don't join any team within deadline of phase 1, you will randomly be grouped with others by instructor.

1. **Graphical User Interface (GUI):**
    - Design and implement a GUI using **Java Swing**.
    - The GUI should include forms for adding, updating, and deleting inventory items, suppliers, and sales records.
    - Provide user-friendly navigation with menus, buttons, and text fields.
2. **Database Connection:**
    - Use a relational database (**SQLite**) to store inventory data, sales records, and supplier information.
    - Implement CRUD operations (Create, Read, Update, Delete) for all major entities.
3. **File I/O:**
    - Implement functionality to import and export inventory data as **text files** (e.g., CSV or txt file).
    - Provide error handling for invalid file formats or corrupted data.
4. **Object-Oriented Principles:**
    - Use classes, inheritance, polymorphism, interfaces, and abstract classes meaningfully.
    - Create a base class for items and use inheritance to represent specific categories (e.g., electronics, groceries).
    - Implement interfaces for tasks like exporting data.
5. **Customization:**
    - Teams must customize the project by selecting a specific type of store (e.g., bookstore, electronic store, clothing store).
    - The chosen store type should influence the GUI design, inventory categories, and database schema.

Deliverables:

1. Source code with detailed comments.
2. A functional executable file or JAR file.
3. A database schema and sample data.
4. A user manual for the application.
5. A project report documenting the following:
    - Project design and architecture.
    - Explanation of how each OOP concept was applied.
    - Description of customization and its impact on the design.

Grading Criteria:

- **Code Quality (15%)**
    - Proper use of OOP principles (inheritance, polymorphism, etc.).
    - Readability and use of meaningful comments.
    - You are expected to write your own set of classes, interfaces or other program components.
- **Functionality (20%)**
    - All required features are implemented and working correctly.
    - Customization is implemented effectively.
- **GUI Design (15%)**
    - User-friendly and visually appealing design.
    - Effective navigation and layout.
- **Database Integration (15%)**
    - Correct implementation of CRUD operations.
    - Proper handling of database connections and queries.
- **File I/O (10%)**
    - Successful import/export of data.
    - Robust error handling for file operations.
- **Documentation and Presentation (10%)**
    - Comprehensive user manual.
    - Well-written project report.
    - Clear and professional presentation.
- **Completion of Each Phase (15%)**
    - Document related to each phase in timeline must be submitted
    - Each phase corresponds to 5 percent(from Phase 1-3)

Note about grading: If you do not submit your final project and its report and also do not make presentation on time, **you will get zero automatically from the assignment**

Timeline:

1. **Phase 1:** Form teams and finalize the store type.(**Deadline**: 15.12.2024 23:55) **APPROVAL OF INSTRUCTOR REQURIED for chosen topic!** Instructor will change your topic.

2. **Phase 2:** Submit a project proposal including initial database schema and class diagram. (**Deadline**: 22.12.2024 23:55)

3. **Phase 3:** Submit a mid-project progress report with a working prototype.(**Deadline**: 29.12.2024 23:55)

4. **Phase 4:** Final project submission and presentation.(**Deadline**: 04.01.2025 10:00)

   NOTE:

   - If you are not able to submit required document until deadline, you need to submit late document with the other documents in next phase. For example, if you are not able to complete phase 1 document until deadline, upload this document with document of phase 2.
   - Each group member sholud submit doucments individually although they organize the same document within team.

Rules and Policies:

1. Teamwork is mandatory, and roles within the team should be defined and documented.
2. Plagiarism will result in disqualification.
3. During presentation, for each team member, individual questions will be asked.
4. Each team member will be graded individually.
5. Late submissions will incur a penalty unless prior approval is obtained.
