Object-Oriented Programming
Semester 2025-I
Workshop No. 4 — Layers Architecture

Eng. Carlos Andrés Sierra, M.Sc.
Computer Engineering
Universidad Distrital Francisco José de Caldas

This workshop builds upon **Workshop #1** (conceptual design), **Workshop #2** (technical design), and **Workshop #3** (architectural principles) to finalize your **layered application** implementation. You will create a **Java Swing-based GUI** and integrate **file-based persistence** to replicate basic database operations within your simple transactional application.

Workshop Scope and Objectives:

- **Layered Architecture Review:** Confirm your classes are logically divided into presentation, business, and data handling layers.
- **Java Swing UI Implementation:** Develop a small, functional interface allowing end-users to perform core transactions or data manipulations.
- **File Persistence:** Serialize or otherwise store key data (transactions, user details, etc.) to a local file.

Carlos Andrés Sierra, Computer Engineer, M.Sc. in Computer Engineering, Titular Professor at Universidad Distrital Francisco José de Caldas.

Any comment or concern about this workshop can be sent to Carlos A. Sierra at: *cavirguezs@udistrital.edu.co*.

Methodology and Deliverables:

1. **Revisiting Layers and Design:**
   - Check that your existing class diagrams and design documents still hold with this final layer approach.
   - Adjust or refactor as needed to ensure minimal coupling and clear responsibility boundaries.
2. **Swing-based GUI Prototype:**
   - Implement one or more basic forms or windows (using Java Swing) for core actions (e.g., create transaction, list items).
   - Keep the interface sufficiently simple, emphasizing functionality over aesthetics.
3. **File Storage:**
   - Write methods to persist objects (deriving from your business classes) to a file, then reload them when the application restarts.
   - Manage possible data conflicts or validation scenarios in a basic but robust manner.
4. **Documentation and Artifact Submission:**
   - Provide updated UML diagrams (class or sequence) illustrating how each layer communicates in the final solution.
   - Include short code samples or references to newly added classes for the GUI and data access logic.
5. **Final Deliverables:**
   - A PDF combining diagrams, brief implementation notes, and usage instructions.
   - A `Workshop-4` folder in your repository containing the code, documentation, and a `README.md` with build/run steps.

Deadline: Friday, June 27th21st, 2025, 16:00. Late submissions may be subject to penalties.

Notes:

- Use **English** for all written deliverables.
- Cite any references or tutorials that aided your Swing and file IO implementations.
- This final workshop showcases a complete, multi-layer approach for your transactional application, positioning you for additional features or improvements after the course.

Congratulations on reaching the final step of your OOP journey! Focus on integrating a user-friendly GUI, effective file persistence, and a robust layered design to finalize your project successfully.