

Project Assignment: Comprehensive Java Development (Weeks 1-4)

CAT 1 - Paractical

Title: Developing a Dynamic Online Bookstore Management System

Scenario:

TechBook Solutions, a startup building a small-scale **online bookstore**, hired you as a **junior developer**. Your task is to develop the foundation for this web-based application using the knowledge you've gained in **Swing, AWT, Servlets, Session Tracking, and Inter-Servlet Communication**. The application will have both a desktop-based GUI for admin operations (like managing books) and a web-based component to track user sessions and allow users to browse and purchase books.

Your project will be developed using **NetBeans IDE** and deployed on **Apache Tomcat** for the web-based components. You will also be using **Swing and AWT** for the GUI-based desktop application.

Objectives:

By the end of this project, you should be able to:

1. Develop a **Swing & AWT-based GUI** for managing books in the bookstore.
2. Create dynamic web-based components using **Servlets**.
3. Implement **Session Tracking** to manage users' sessions when they log in and browse the bookstore.
4. Enable communication between servlets to ensure smooth data flow across multiple components of the web application.

Project Breakdown (Weeks 1-4)

Week 1-2: Develop the Admin Panel Using Swing & AWT

You will first create a **desktop application** using Swing and AWT for the admin (bookstore manager) to:

- **Add** new books to the bookstore.
- **View** the list of available books.
- **Update** book information (title, author, price, and stock).
- **Delete** books from the system.

Key Components:

- **JFrame**: For the main window.

- **JButton, JTextField, JLabel:** To allow input and actions for adding and updating books.
- **JTable:** To display the list of books in the bookstore.

Deliverable 1:

- A fully functional **Admin Panel** developed in **Swing** and **AWT**, where the admin can manage books. You should be able to perform basic **CRUD** (Create, Read, Update, Delete) operations on the books using the GUI.

Week 3: Create the Web-Based User Section Using Servlets

Now, shift your focus to the **web-based part of the application**. Using **Servlets**, create a basic web interface for users to:

- View available books (fetch data from a servlet).
- Add books to their cart (store this information in the session).
- Remove books from the cart.
- View the contents of their cart.

Key Components:

- **Servlets** to display available books and handle cart operations.
- **HTML** to display the book list.
- A **shopping cart** that tracks user selections across multiple requests using **HTTP sessions**.

Deliverable 2:

- A servlet-based web application where users can:
 - View books (from a pre-populated list or the database).
 - Add books to their cart (using session tracking).
 - View and manage their cart.

Week 4: Implement Session Tracking and Inter-Servlet Communication

This phase requires implementing session tracking and communication between servlets to manage user sessions and data flow in the application. Here's what you need to do:

- Use **Cookies** or **HTTP Sessions** to manage user logins.
- Keep track of the user's **login status**, **shopping cart**, and **order history** using sessions.
- Create a **login servlet** where users can log in and start a session.

- Allow communication between servlets (for example, the login servlet should pass the session to the book viewing and cart servlets).

Key Components:

- **LoginServlet:** Authenticate users and start a session.
- **BookServlet:** Display available books to logged-in users.
- **CartServlet:** Allow users to add/remove items from the cart and view their cart.

Deliverable 3:

- A fully functional **login system** that tracks user sessions and allows them to log in, browse books, add items to the cart, and view their cart across multiple requests.

Additional Requirements

- Use **clear and organized code** with meaningful comments.
- Ensure proper **exception handling** for errors like invalid input or session expiration.
- Follow **best practices** in Java development, including proper folder structure and separation of concerns (e.g., separating logic between GUI and web-based components).

Final Deliverable:

Your final submission should include:

1. **Admin Panel (Swing & AWT):** The desktop GUI for managing books.
2. **Web Application (Servlets):** The web-based components for users to browse and manage their shopping carts.
3. **Session Tracking & Inter-Servlet Communication:** Implement user session tracking to ensure a smooth user experience.
4. **Documentation:** A brief **report** explaining:
 - How each component works (both desktop and web).
 - How you implemented session tracking and inter-servlet communication.
 - Screenshots of the running application (GUI and web pages).

Submit your project as a **NetBeans project file** along with the **war file** for the web application.

Grading Criteria:

- **Functionality** (40%): Does your project meet all the specified requirements? Does it run without errors?
- **Code Quality** (20%): Is your code clean, well-organized, and well-commented?
- **Session Tracking & Communication** (20%): Have you correctly implemented session tracking and servlet communication?
- **Documentation** (10%): Is your documentation clear and informative?
- **Creativity** (10%): Have you added any additional features or demonstrated creativity in your solution?

Final Notes:

- Make sure to **test each component** as you develop it to avoid issues later.
- **Seek help** if you encounter difficulties, but make sure to work through problems and try to find solutions on your own.
- The purpose of this assignment is to apply what you've learned in a real-world scenario, so take the time to understand each part of the project.

Good luck and have fun developing your project!