**Full Project Code with Detailed Comments**

* All Java files have been updated with comments to explain each class, method, and variable used throughout the project.
* **Complete Java Files**
  + OrderEaseApp.java
  + Restaurant.java
  + Order.java
  + Table.java
  + Server.java
  + MenuItem.java

**What It Accomplishes**

The OrderEase application is a streamlined restaurant management system that enhances both guest experience and staff efficiency. It allows guests to interact directly with the menu and place orders. It also provides servers and admins with dedicated role-specific interfaces for efficient management.

* **Guest Role**
  + Guests can view the menu, place orders, call the server, and request the check.
  + The guest interface is simple and user-friendly, offering quick access to core features without the need for logging in.
* **Server Role**
  + Servers can manage their assigned tables, view order statuses, check in with guests, and mark orders as served.
  + Servers receive real-time notifications for guest actions like placing orders or requesting the check, improving service quality.
* **Admin Role**
  + Admins can control the restaurant's menu by adding, removing, or updating menu items, and manage server assignments to tables.
  + Admin functions include real-time menu management and dynamic server resource allocation.

**Key Features**

* Guest Interface
  + View the menu with item details including name, description, price, and availability.
  + Place orders directly at the table without needing a login.
  + Call the server or request the check at any time.
* Role Switching
  + A button on the guest interface allows switching between server and admin roles without requiring any credentials.
* Server Management
  + View all assigned tables and order statuses.
  + Check in with guests and mark their orders as served.
* Admin Management
  + Add, remove, or update menu items directly from the admin interface.
  + Assign or reassign servers to tables based on restaurant needs.

**Multiple Test Scenarios**

* **Test Scenario 1: Guest Interface Testing**
  + **View Menu**
    - Browse available menu items with name, description, price, and availability.
  + **Place an Order**
    - Use the "Place Order" button and select from available items.
    - Ensure the item is added to the order and the assigned server is notified.
  + **Call Server**
    - Use the "Call Server" button to notify the assigned server.
    - Verify that the server receives the notification.
  + **Request Check**
    - Use the "Request Check" button to ask for the bill.
    - Ensure the server is notified.
* **Test Scenario 2: Role Switching and Server Interface Testing**
  + **Switch to Server Role**
    - Select the "Server/Admin" button and choose "Server."
    - The server interface should be displayed without needing a login.
  + **View Assigned Tables**
    - Use the "View Tables" button to see all tables assigned to the server.
    - Verify that the table number, seating duration, and order status are correctly displayed.
  + **Check In with Guests**
    - Select the "Check In" button, enter the table number, and confirm.
    - Ensure the server's check-in with the guest is logged.
  + **Mark Order Served**
    - Use the "Mark Order Served" button, select a table number, and confirm.
    - Verify that the order is marked as served and cleared from the order list.
* **Test Scenario 3: Admin Interface Testing**
  + **Switch to Admin Role**
    - Select the "Server/Admin" button and choose "Admin."
    - The admin interface should be displayed without requiring credentials.
  + **Manage Menu Items**
    - Add a new menu item using the "Add Menu Item" button.
    - Verify that the new item appears in the guest menu view.
    - Use the "Remove Menu Item" button to remove an existing item and confirm that it no longer appears in the menu.
  + **Server Assignment**
    - Use the "Assign Server to Table" button to assign or reassign servers to specific tables.
    - Ensure that the server is correctly assigned and updated for that table.

**How to Run the Program**

* **Prerequisites** 
  + Install Java Development Kit (JDK).
  + An IDE such as VSCode, IntelliJ IDEA, or Eclipse.
  + Ensure that all source code files (OrderEaseApp.java, Restaurant.java, Server.java, Table.java, Order.java, and MenuItem.java) are properly imported into your project.
  + If using VSCode, install the "Extension Pack for Java" and ensure the Java runtime environment is set up.
* **Running the Program**
  + Open the project in your chosen IDE.
  + Compile and run the OrderEaseApp.java class.
  + Use the guest interface to simulate a table interaction.
    - Simulate guest interactions, such as viewing the menu and placing orders.
  + Use the "Server/Admin" button to switch to server or admin roles without entering credentials.
* **Data Storage**
  + Data (tables, servers, orders, and menu items) is saved automatically to a file named restaurant\_data.dat.
  + The data persists across sessions and can be manually saved via the application if needed.