# Service Provider and Customer Management System

Objective:

Develop a REST API using Java Spring Boot to create a service similar to Urban Company, which connects service providers with customers seeking help at their doorstep. The system should include secure registration for both customers and service providers, service request posting, notification to service providers based on their availability and location, calendar management, and a wallet API for transactions. (use Urban Company as a reference)

The key features of the system are:

1. **Secure Registration**:
   1. Customers and service providers must register separately.
   2. Each registration should generate a secure JWT token for authentication and authorization.
2. **Service Request Management**:
   1. Customers can post service requests specifying the category (e.g., plumbing, wiring, painting), description, location, and preferred date.
   2. The system should notify relevant service providers based on their availability and proximity to the customer's location.
3. **Service Provider Availability and Calendar Management**:
   1. Service providers can manage their availability by updating their calendar.
   2. Once a service provider accepts a request, their calendar should be automatically updated to block the time slot. The other notifications sent to other service providers should be disabled.
4. **Service Categories and Toggle Availability**:
   1. Service providers can enlist in multiple categories and manage the services they offer.
   2. They should be able to turn on/off specific services for certain dates.
5. **Wallet and Transactions**:
   1. Implement a wallet API to handle transactions between customers and service providers.
   2. Customers can make payments through the wallet once the services are provided.
   3. Customer should be able to add money make payment for services
   4. Service providers should be able to withdraw balances and view their statements
   5. Implement audit functionality
   6. Deduct Handling fees at 10% – 15% based on the service Category
6. **Notification System**:
   1. Implement a notification system to alert service providers of new service requests. This should be an api where users can view the notification sent to them
   2. Ensure timely and relevant notifications based on service provider availability and location.
7. **Security and Validation**:
   1. Ensure all APIs are secure and validate input data to prevent unauthorized access and data breaches.
8. **Logging and Audit Logging:**
   1. Implement proper request logging and Audit for every api request
   2. All payment transactions should be logged in a separate audit log and table for additional security
9. **Concurrency Handling**
   1. Handle concurrent api requests simulation a large volume of transaction simultaneously. (Use hibernate version entities)
10. **Reporting**
    1. Provide operational metrics on key parameters like
       1. User active
       2. Service Providers active
       3. Active service request
    2. Collect review from customer and service Providers for Quality assurance