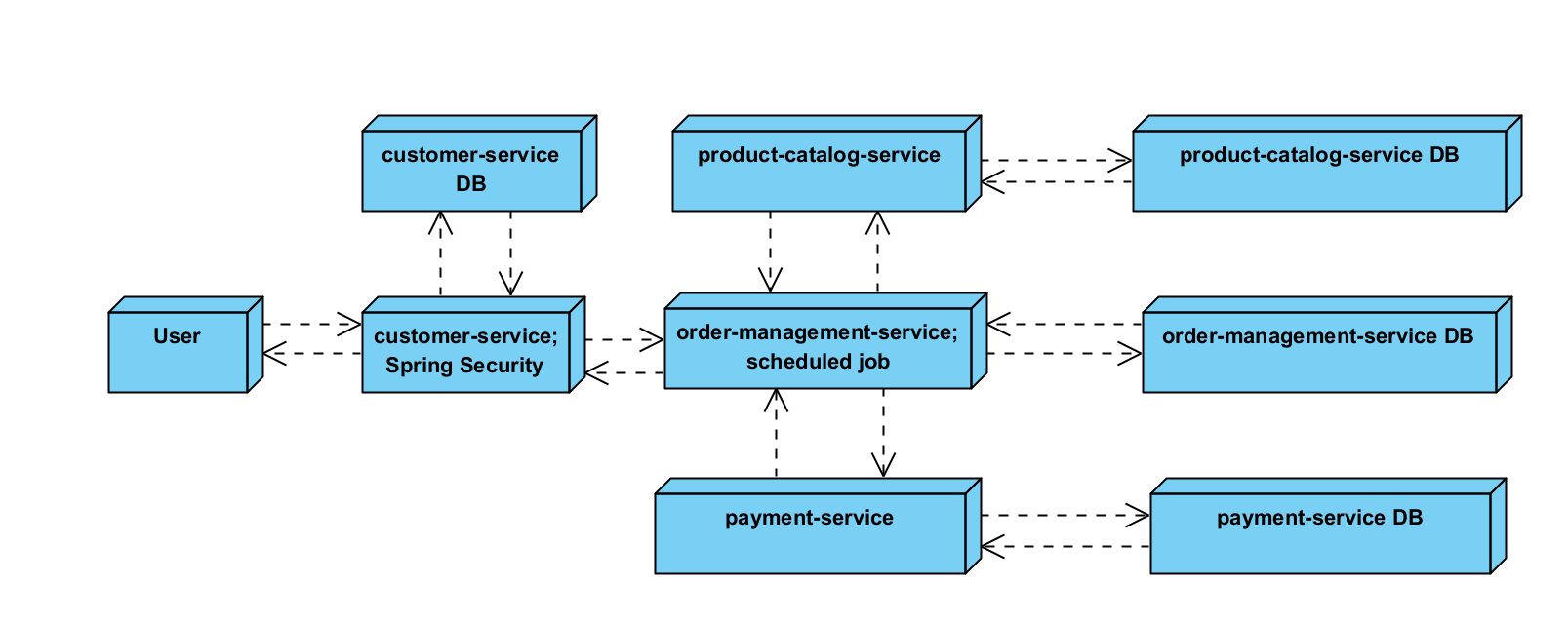
**Workflow and Design Diagram: Sales Management Microservices System**

**Introduction:** The Sales Management System consists of four microservices (Customer, Product Catalog, Order Management, and Payment). These services work together to handle the complete sales process. Below is the workflow and a high-level design to explain how the services interact.

**Workflow**

1. **Customer Registration and Login**:
   * A customer registers through the **Customer Service** and logs in securely using **Spring Security**.
   * Roles (Admin or Customer) are assigned during registration.
2. **Browsing Products**:
   * The customer requests product details through the **Product Catalog Service**.
   * The Product Catalog Service fetches information (name, price, and quantity) from its database.
3. **Placing an Order**:
   * The customer places an order through the **Order Management Service**.
   * The Order Management Service:
     + Verifies the product’s availability by calling the **Product Catalog Service**.
     + Creates an order record in its database.
4. **Processing Payment**:
   * The **Order Management Service** sends a payment request to the **Payment Service**.
   * The Payment Service:
     + Processes the payment.
     + Sends the payment status (success or failure) back to the Order Management Service.
5. **Order Completion**:
   * The Order Management Service updates the order status (e.g., "Payment Successful").
   * Customers can view their order status via the Order Management Service.
6. **Scheduled Job**:
   * The Order Management Service runs a scheduled job to archive completed orders older than six months.

**Design Diagram**



Here’s how the microservices interact:

1. **Customer Service**:
   * Handles user registration and authentication.
   * Database: Stores user information and roles.
2. **Product Catalog Service**:
   * Provides product details to customers.
   * Database: Stores product information (name, price, quantity).
3. **Order Management Service**:
   * Handles order placement and status updates.
   * Communicates with the Product Catalog and Payment Services.
   * Scheduled job: Archives old orders.
   * Database: Stores order details.
4. **Payment Service**:
   * Processes payments for orders.
   * Database: Logs payment transactions.

**Interaction Flow:**

1. **REST API Communication**:
   * Services interact using REST APIs via **Spring’s RestTemplate**.
2. **Database Independence**:
   * Each service uses its own database to ensure loose coupling.
3. **Security**:
   * All endpoints are secured with **Spring Security**.
   * Role-based access:
     + Admin: Full access to all services.
     + Customer: Limited access to product browsing, order placement, and payment.

**Workflow Example**

1. A **Customer** logs in via the Customer Service.
2. The customer browses the **Product Catalog** to select an item.
3. An order is placed via the **Order Management Service**, which verifies product availability.
4. The **Payment Service** processes the payment.
5. The order status is updated, and the customer can view the final status.