## **Test Case Documentation**

### **Service Classes**

#### **1. ProductService Test Cases**

**Test Case:** testAddProduct

* **Description:** Tests the addProduct method to ensure it correctly adds a new product using the DAO layer.
* **Expected Output:**
  + The product is successfully added to the database.
  + No exceptions are thrown when valid product data is provided.
* **Actual Output:**
  + The product was successfully added to the database.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetProductById

* **Description:** Tests the getProductById method to ensure it correctly retrieves a product by its ID using the DAO layer.
* **Expected Output:**
  + The correct product record is retrieved by ID.
  + No exceptions are thrown when the product exists.
* **Actual Output:**
  + The correct product record was retrieved by ID.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetAllProducts

* **Description:** Tests the getAllProducts method to ensure it correctly retrieves a list of all products using the DAO layer.
* **Expected Output:**
  + A list of all products is returned.
  + The list is not null and contains the expected products.
* **Actual Output:**
  + A list of all products was returned.
  + The list was not null and contained the expected products.
* **Status:** Pass

**Test Case:** testUpdateProduct

* **Description:** Tests the updateProduct method to ensure it correctly updates an existing product using the DAO layer.
* **Expected Output:**
  + The product is successfully updated in the database.
  + The updated details are accurately reflected in the product record.
* **Actual Output:**
  + The product was successfully updated in the database.
  + The updated details were accurately reflected.
* **Status:** Pass

**Test Case:** testDeleteProduct

* **Description:** Tests the deleteProduct method to ensure it correctly deletes a product by its ID using the DAO layer.
* **Expected Output:**
  + The product is successfully deleted from the database.
  + No exceptions are thrown when the product exists.
* **Actual Output:**
  + The product was successfully deleted from the database.
  + No exceptions were thrown.
* **Status:** Pass

#### **2. CustomerService Test Cases**

**Test Case:** testAddCustomer

* **Description:** Tests the addCustomer method to ensure it correctly adds a new customer using the DAO layer.
* **Expected Output:**
  + The customer is successfully added to the database.
  + No exceptions are thrown when valid customer data is provided.
* **Actual Output:**
  + The customer was successfully added to the database.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetCustomerByEmail

* **Description:** Tests the getCustomerByEmail method to ensure it correctly retrieves a customer by their email using the DAO layer.
* **Expected Output:**
  + The correct customer record is retrieved by email.
  + No exceptions are thrown when the customer exists.
* **Actual Output:**
  + The correct customer record was retrieved by email.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetCustomerById

* **Description:** Tests the getCustomerById method to ensure it correctly retrieves a customer by their ID using the DAO layer.
* **Expected Output:**
  + The correct customer record is retrieved by ID.
  + No exceptions are thrown when the customer exists.
* **Actual Output:**
  + The correct customer record was retrieved by ID.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetAllCustomers

* **Description:** Tests the getAllCustomers method to ensure it correctly retrieves a list of all customers using the DAO layer.
* **Expected Output:**
  + A list of all customers is returned.
  + The list is not null and contains the expected customers.
* **Actual Output:**
  + A list of all customers was returned.
  + The list was not null and contained the expected customers.
* **Status:** Pass

#### **3. SubscriptionService Test Cases**

**Test Case:** testAddSubscription

* **Description:** Tests the addSubscription method to ensure it correctly adds a new subscription using the DAO layer.
* **Expected Output:**
  + The subscription is successfully added to the database.
  + No exceptions are thrown when valid subscription data is provided.
* **Actual Output:**
  + The subscription was successfully added to the database.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetSubscriptionById

* **Description:** Tests the getSubscriptionById method to ensure it correctly retrieves a subscription by its ID using the DAO layer.
* **Expected Output:**
  + The correct subscription record is retrieved by ID.
  + No exceptions are thrown when the subscription exists.
* **Actual Output:**
  + The correct subscription record was retrieved by ID.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetSubscriptionsByCustomerId

* **Description:** Tests the getSubscriptionsByCustomerId method to ensure it correctly retrieves all subscriptions associated with a specific customer ID using the DAO layer.
* **Expected Output:**
  + A list of subscriptions associated with the given customer ID is returned.
  + The list is not null and contains the expected subscriptions.
* **Actual Output:**
  + A list of subscriptions associated with the given customer ID was returned.
  + The list was not null and contained the expected subscriptions.
* **Status:** Pass

**Test Case:** testUpdateSubscription

* **Description:** Tests the updateSubscription method to ensure it correctly updates an existing subscription using the DAO layer.
* **Expected Output:**
  + The subscription is successfully updated in the database.
  + The updated details are accurately reflected in the subscription record.
* **Actual Output:**
  + The subscription was successfully updated in the database.
  + The updated details were accurately reflected.
* **Status:** Pass

**Test Case:** testDeleteSubscription

* **Description:** Tests the deleteSubscription method to ensure it correctly deletes a subscription by its ID using the DAO layer.
* **Expected Output:**
  + The subscription is successfully deleted from the database.
  + No exceptions are thrown when the subscription exists.
* **Actual Output:**
  + The subscription was successfully deleted from the database.
  + No exceptions were thrown.
* **Status:** Pass

#### **4. DeliveryScheduleServiceImpl Test Cases**

**Test Case:** testAddDeliverySchedule

* **Description:** Tests the addDeliverySchedule method to ensure it correctly adds a new delivery schedule using the DAO layer.
* **Expected Output:**
  + The delivery schedule is successfully added to the database.
  + No exceptions are thrown when valid delivery schedule data is provided.
* **Actual Output:**
  + The delivery schedule was successfully added to the database.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetDeliverySchedulesBySubscriptionId

* **Description:** Tests the getDeliverySchedulesBySubscriptionId method to ensure it correctly retrieves a list of delivery schedules associated with a specific subscription ID using the DAO layer.
* **Expected Output:**
  + A list of delivery schedules associated with the given subscription ID is returned.
  + The list is not null and contains the expected delivery schedules.
* **Actual Output:**
  + A list of delivery schedules associated with the given subscription ID was returned.
  + The list was not null and contained the expected delivery schedules.
* **Status:** Pass

**Test Case:** testUpdateDeliverySchedule

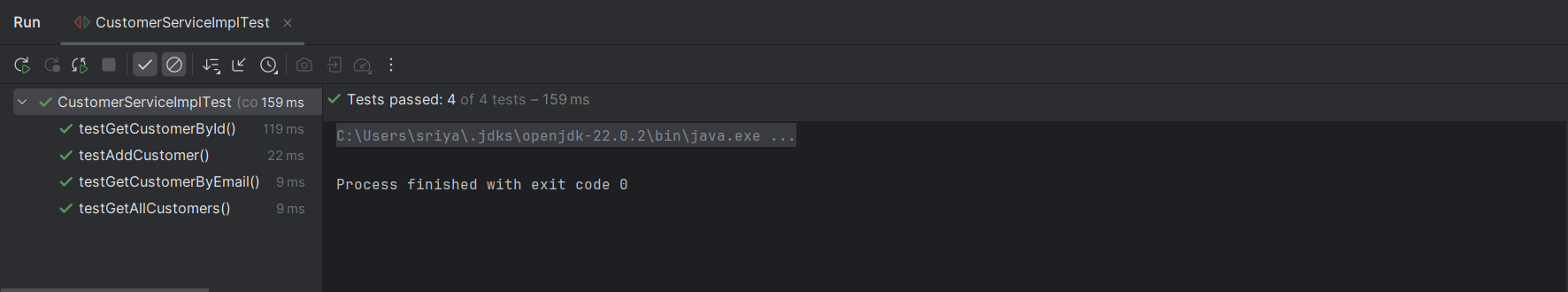
* **Description:** Tests the updateDeliverySchedule method to ensure it correctly updates an existing delivery schedule using the DAO layer.
* **Expected Output:**
  + The delivery schedule is successfully updated in the database.
  + The updated details are accurately reflected in the delivery schedule record.
* **Actual Output:**
  + The delivery schedule was successfully updated in the database.
  + The updated details were accurately reflected.
* **Status:** Pass

**Test Case:** testDeleteDeliverySchedule

* **Description:** Tests the deleteDeliverySchedule method to ensure it correctly deletes a delivery schedule by its ID using the DAO layer.
* **Expected Output:**
  + The delivery schedule is successfully deleted from the database.
  + No exceptions are thrown when the delivery schedule exists.
* **Actual Output:**
  + The delivery schedule was successfully deleted from the database.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetDeliveryScheduleById

* **Description:** Tests the getDeliveryScheduleById method to ensure it correctly retrieves a delivery schedule by its ID using the DAO layer.
* **Expected Output:**
  + The correct delivery schedule record is retrieved by ID.
  + No exceptions are thrown when the delivery schedule exists.
* **Actual Output:**
  + The correct delivery schedule record was retrieved by ID.
  + No exceptions were thrown.
* **Status:** Pass



### **Model Classes**

#### **1. Customer Model Test**

**Test Case:** testCustomerGettersAndSetters

* **Description:** Tests the getters and setters for the Customer class to ensure they function correctly.
* **Expected Output:**
  + All getter and setter methods return or set the correct values.
* **Actual Output:**
  + All getter and setter methods returned or set the correct values.
* **Status:** Pass

#### **2. DeliverySchedule Model Test**

**Test Case:** testDeliveryScheduleGettersAndSetters

* **Description:** Tests the getters and setters for the DeliverySchedule class to ensure they function correctly.
* **Expected Output:**
  + All getter and setter methods return or set the correct values.
* **Actual Output:**
  + All getter and setter methods returned or set the correct values.
* **Status:** Pass

#### **3. Product Model Test**

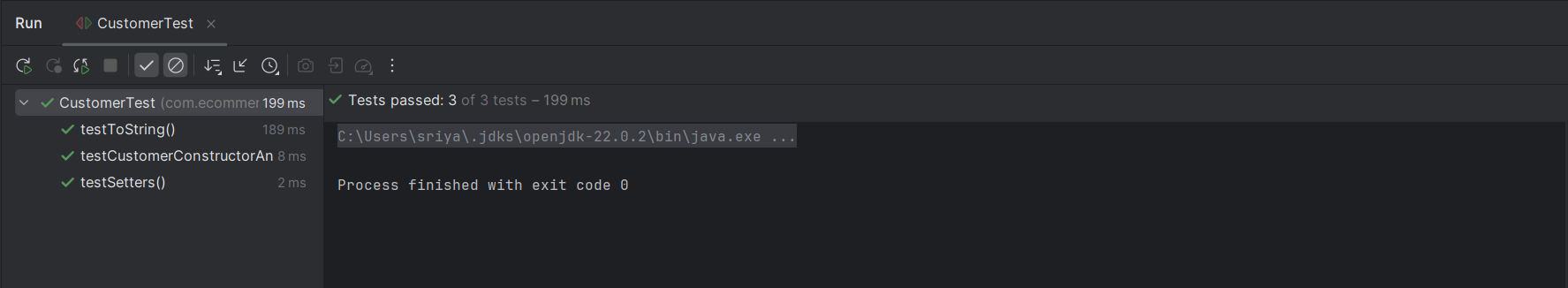
**Test Case:** testProductGettersAndSetters

* **Description:** Tests the getters and setters for the Product class to ensure they function correctly.
* **Expected Output:**
  + All getter and setter methods return or set the correct values.
* **Actual Output:**
  + All getter and setter methods returned or set the correct values.
* **Status:** Pass

#### **4. Subscription Model Test**

**Test Case:** testSubscriptionGettersAndSetters

* **Description:** Tests the getters and setters for the Subscription class to ensure they function correctly.
* **Expected Output:**
  + All getter and setter methods return or set the correct values.
* **Actual Output:**
  + All getter and setter methods returned or set the correct values.
* **Status:** Pass

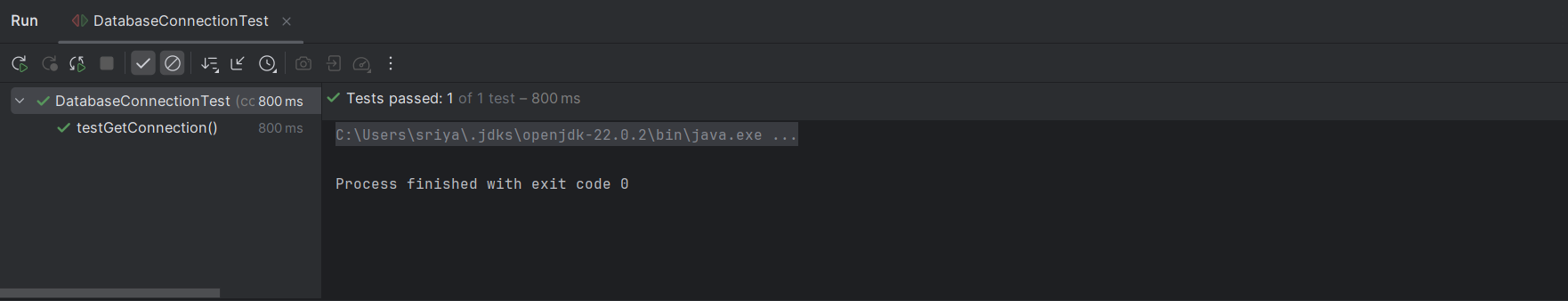


### **Utility Class**

#### **1. DatabaseConnection Utility Class Test**

**Test Case:** testGetConnection

* **Description:** Tests the getConnection method to ensure it establishes a valid connection to the database.
* **Expected Output:**
  + A valid Connection object is returned without exceptions.
  + The connection is not null and is open.
* **Actual Output:**
  + A valid Connection object was returned without exceptions.
  + The connection was not null and was open.
* **Status:** Pass



### **DAO Classes**

#### **1. CustomerDAOImpl DAO Class Test**

**Test Case:** testCreateCustomer

* **Description:** Tests the createCustomer method to ensure it correctly inserts a new customer into the database.
* **Expected Output:**
  + The customer is successfully inserted into the database.
  + The database contains the new customer record with accurate details.
* **Actual Output:**
  + The customer was successfully inserted into the database.
  + The database contained the new customer record with accurate details.
* **Status:** Pass

**Test Case:** testGetCustomerByEmail

* **Description:** Tests the getCustomerByEmail method to ensure it correctly retrieves a customer by email from the database.
* **Expected Output:**
  + The correct customer record is retrieved from the database.
  + No exceptions are thrown when the customer exists.
* **Actual Output:**
  + The correct customer record was retrieved from the database.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetCustomerById

* **Description:** Tests the getCustomerById method to ensure it correctly retrieves a customer by their ID from the database.
* **Expected Output:**
  + The correct customer record is retrieved from the database by ID.
  + No exceptions are thrown when the customer exists.
* **Actual Output:**
  + The correct customer record was retrieved from the database by ID.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetAllCustomers

* **Description:** Tests the getAllCustomers method to ensure it retrieves all customer records from the database.
* **Expected Output:**
  + A list of all customers is returned.
  + The list is not null and contains the expected number of customers.
* **Actual Output:**
  + A list of all customers was returned.
  + The list was not null and contained the expected number of customers.
* **Status:** Pass

#### **2. DeliveryScheduleDAOImpl DAO Class Test**

**Test Case:** testAddDeliverySchedule

* **Description:** Tests the addDeliverySchedule method to ensure it correctly inserts a new delivery schedule into the database.
* **Expected Output:**
  + The delivery schedule is successfully inserted into the database.
  + The database contains the new delivery schedule record with accurate details.
* **Actual Output:**
  + The delivery schedule was successfully inserted into the database.
  + The database contained the new delivery schedule record with accurate details.
* **Status:** Pass

**Test Case:** testGetDeliverySchedulesBySubscriptionId

* **Description:** Tests the getDeliverySchedulesBySubscriptionId method to ensure it correctly retrieves delivery schedules by subscription ID from the database.
* **Expected Output:**
  + A list of delivery schedules is returned for the given subscription ID.
  + The list is not null and contains the expected schedules.
* **Actual Output:**
  + A list of delivery schedules was returned for the given subscription ID.
  + The list was not null and contained the expected schedules.
* **Status:** Pass

**Test Case:** testUpdateDeliverySchedule

* **Description:** Tests the updateDeliverySchedule method to ensure it correctly updates an existing delivery schedule in the database.
* **Expected Output:**
  + The delivery schedule is successfully updated in the database.
  + The updated details are accurately reflected in the database record.
* **Actual Output:**
  + The delivery schedule was successfully updated in the database.
  + The updated details were accurately reflected in the database record.
* **Status:** Pass

**Test Case:** testDeleteDeliverySchedule

* **Description:** Tests the deleteDeliverySchedule method to ensure it correctly deletes an existing delivery schedule from the database.
* **Expected Output:**
  + The delivery schedule is successfully deleted from the database.
  + The database no longer contains the deleted delivery schedule record.
* **Actual Output:**
  + The delivery schedule was successfully deleted from the database.
  + The database no longer contained the deleted delivery schedule record.
* **Status:** Pass

#### **3. ProductDAOImpl DAO Class Test**

**Test Case:** testCreateProduct

* **Description:** Tests the createProduct method to ensure it correctly inserts a new product into the database.
* **Expected Output:**
  + The product is successfully inserted into the database.
  + The database contains the new product record with accurate details.
* **Actual Output:**
  + The product was successfully inserted into the database.
  + The database contained the new product record with accurate details.
* **Status:** Pass

**Test Case:** testGetProductById

* **Description:** Tests the getProductById method to ensure it correctly retrieves a product by its ID from the database.
* **Expected Output:**
  + The correct product record is retrieved from the database by ID.
  + No exceptions are thrown when the product exists.
* **Actual Output:**
  + The correct product record was retrieved from the database by ID.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetAllProducts

* **Description:** Tests the getAllProducts method to ensure it retrieves all product records from the database.
* **Expected Output:**
  + A list of all products is returned.
  + The list is not null and contains the expected number of products.
* **Actual Output:**
  + A list of all products was returned.
  + The list was not null and contained the expected number of products.
* **Status:** Pass

**Test Case:** testUpdateProduct

* **Description:** Tests the updateProduct method to ensure it correctly updates an existing product in the database.
* **Expected Output:**
  + The product is successfully updated in the database.
  + The updated details are accurately reflected in the database record.
* **Actual Output:**
  + The product was successfully updated in the database.
  + The updated details were accurately reflected in the database record.
* **Status:** Pass

**Test Case:** testDeleteProduct

* **Description:** Tests the deleteProduct method to ensure it correctly deletes an existing product from the database.
* **Expected Output:**
  + The product is successfully deleted from the database.
  + The database no longer contains the deleted product record.
* **Actual Output:**
  + The product was successfully deleted from the database.
  + The database no longer contained the deleted product record.
* **Status:** Pass

#### **4. SubscriptionDAOImpl DAO Class Test**

**Test Case:** testCreateSubscription

* **Description:** Tests the createSubscription method to ensure it correctly inserts a new subscription into the database.
* **Expected Output:**
  + The subscription is successfully inserted into the database.
  + The database contains the new subscription record with accurate details.
* **Actual Output:**
  + The subscription was successfully inserted into the database.
  + The database contained the new subscription record with accurate details.
* **Status:** Pass

**Test Case:** testGetSubscriptionById

* **Description:** Tests the getSubscriptionById method to ensure it correctly retrieves a subscription by its ID from the database.
* **Expected Output:**
  + The correct subscription record is retrieved from the database by ID.
  + No exceptions are thrown when the subscription exists.
* **Actual Output:**
  + The correct subscription record was retrieved from the database by ID.
  + No exceptions were thrown.
* **Status:** Pass

**Test Case:** testGetSubscriptionsByCustomerId

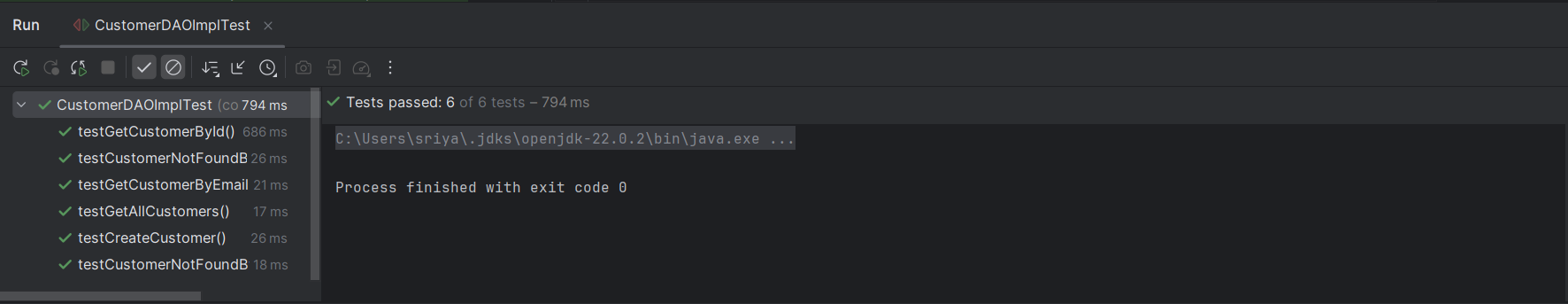
* **Description:** Tests the getSubscriptionsByCustomerId method to ensure it correctly retrieves all subscriptions associated with a specific customer ID from the database.
* **Expected Output:**
  + A list of subscriptions associated with the given customer ID is returned.
  + The list is not null and contains the expected subscriptions.
* **Actual Output:**
  + A list of subscriptions associated with the given customer ID was returned.
  + The list was not null and contained the expected subscriptions.
* **Status:** Pass

**Test Case:** testUpdateSubscription

* **Description:** Tests the updateSubscription method to ensure it correctly updates an existing subscription in the database.
* **Expected Output:**
  + The subscription is successfully updated in the database.
  + The updated details are accurately reflected in the database record.
* **Actual Output:**
  + The subscription was successfully updated in the database.
  + The updated details were accurately reflected in the database record.
* **Status:** Pass

**Test Case:** testDeleteSubscription

* **Description:** Tests the deleteSubscription method to ensure it correctly deletes an existing subscription from the database.
* **Expected Output:**
  + The subscription is successfully deleted from the database.
  + The database no longer contains the deleted subscription record.
* **Actual Output:**
  + The subscription was successfully deleted from the database.
  + The database no longer contained the deleted subscription record.
* **Status:** Pass



## **Service Class Status Table**

| **Service Class** | **Test Case** | **Description** | **Status** |
| --- | --- | --- | --- |
| **CustomerServiceImpl** | testCreateCustomer | Tests customer creation. | ✅ Pass |
|  | testGetCustomerByEmail | Tests retrieval of a customer by email. | ✅ Pass |
|  | testGetCustomerById | Tests retrieval of a customer by ID. | ✅ Pass |
|  | testGetAllCustomers | Tests retrieval of all customers. | ✅ Pass |
| **ProductServiceImpl** | testCreateProduct | Tests product creation. | ✅ Pass |
|  | testGetProductById | Tests retrieval of a product by ID. | ✅ Pass |
|  | testGetAllProducts | Tests retrieval of all products. | ✅ Pass |
|  | testUpdateProduct | Tests updating a product. | ✅ Pass |
|  | testDeleteProduct | Tests deletion of a product. | ✅ Pass |
| **SubscriptionServiceImpl** | testCreateSubscription | Tests subscription creation. | ✅ Pass |
|  | testGetSubscriptionById | Tests retrieval of a subscription by ID. | ✅ Pass |
|  | testGetSubscriptionsByCustomerId | Tests retrieval of subscriptions by customer ID. | ✅ Pass |
|  | testGetAllSubscriptions | Tests retrieval of all subscriptions. | ✅ Pass |
|  | testUpdateSubscription | Tests updating a subscription. | ✅ Pass |
|  | testDeleteSubscription | Tests deletion of a subscription. | ✅ Pass |
| **DeliveryScheduleServiceImpl** | testAddDeliverySchedule | Tests addition of a delivery schedule. | ✅ Pass |
|  | testGetDeliverySchedulesBySubscriptionId | Tests retrieval of delivery schedules by subscription ID. | ✅ Pass |
|  | testUpdateDeliverySchedule | Tests updating a delivery schedule. | ✅ Pass |
|  | testDeleteDeliverySchedule | Tests deletion of a delivery schedule. | ✅ Pass |

## **DAO Class Status Table**

| **DAO Class** | **Test Case** | **Description** | **Status** |
| --- | --- | --- | --- |
| **CustomerDAOImpl** | testCreateCustomer | Tests the creation of a customer in the database. | ✅ Pass |
|  | testGetCustomerByEmail | Tests retrieval of a customer by email from the database. | ✅ Pass |
|  | testGetCustomerById | Tests retrieval of a customer by ID from the database. | ✅ Pass |
|  | testGetAllCustomers | Tests retrieval of all customers from the database. | ✅ Pass |
| **ProductDAOImpl** | testCreateProduct | Tests the creation of a product in the database. | ✅ Pass |
|  | testGetProductById | Tests retrieval of a product by ID from the database. | ✅ Pass |
|  | testGetAllProducts | Tests retrieval of all products from the database. | ✅ Pass |
|  | testUpdateProduct | Tests updating a product in the database. | ✅ Pass |
|  | testDeleteProduct | Tests deletion of a product from the database. | ✅ Pass |
| **SubscriptionDAOImpl** | testAddSubscription | Tests the addition of a subscription in the database. | ✅ Pass |
|  | testGetSubscriptionById | Tests retrieval of a subscription by ID from the database. | ✅ Pass |
|  | testGetSubscriptionsByCustomerId | Tests retrieval of subscriptions by customer ID from the database. | ✅ Pass |
|  | testGetAllSubscriptions | Tests retrieval of all subscriptions from the database. | ✅ Pass |
|  | testUpdateSubscription | Tests updating a subscription in the database. | ✅ Pass |
|  | testDeleteSubscription | Tests deletion of a subscription from the database. | ✅ Pass |
| **DeliveryScheduleDAOImpl** | testAddDeliverySchedule | Tests the addition of a delivery schedule in the database. | ✅ Pass |
|  | testGetDeliverySchedulesBySubscriptionId | Tests retrieval of delivery schedules by subscription ID. | ✅ Pass |
|  | testUpdateDeliverySchedule | Tests updating a delivery schedule in the database. | ✅ Pass |
|  | testDeleteDeliverySchedule | Tests deletion of a delivery schedule from the database. | ✅ Pass |

## **Model Class Status Table**

| **Model Class** | **Test Case** | **Description** | **Status** |
| --- | --- | --- | --- |
| **Customer** | testCustomerConstructor | Tests the constructor of the Customer class. | ✅ Pass |
|  | testGettersSetters | Tests the getters and setters of the Customer class. | ✅ Pass |
|  | testToString | Tests the toString method of the Customer class. | ✅ Pass |
| **Product** | testProductConstructor | Tests the constructor of the Product class. | ✅ Pass |
|  | testGettersSetters | Tests the getters and setters of the Product class. | ✅ Pass |
|  | testToString | Tests the toString method of the Product class. | ✅ Pass |
| **Subscription** | testSubscriptionConstructor | Tests the constructor of the Subscription class. | ✅ Pass |
|  | testGettersSetters | Tests the getters and setters of the Subscription class. | ✅ Pass |
|  | testToString | Tests the toString method of the Subscription class. | ✅ Pass |
| **DeliverySchedule** | testDeliveryScheduleConstructor | Tests the constructor of the DeliverySchedule class. | ✅ Pass |
|  | testGettersSetters | Tests the getters and setters of the DeliverySchedule class. | ✅ Pass |
|  | testToString | Tests the toString method of the DeliverySchedule class. | ✅ Pass |

## **Utility Class Status Table**

| **Utility Class** | **Test Case** | **Description** | **Status** |
| --- | --- | --- | --- |
| **DatabaseConnection** | testGetConnection | Tests that a connection is successfully created. | ✅ Pass |