Phase 5 — Apex Programming (Developer)

Project: Revolutionizing Agriculture with AgriEdge Or-Mange Ltd — Salesforce OMS

Prepared for: Ijjapureddy Parvathi Devi

Status: Completed (Apex classes, triggers, async jobs and tests created — verification steps and

evidence listed below)

1. Executive summary

This document records everything implemented for **Phase 5: Apex Programming**. It lists the Apex classes, triggers, asynchronous jobs, test classes, how to verify them in your org, example test steps, and recommended improvements. Use this document as the upload-ready deliverable for your mentor.

2. What was implemented (names & purpose)

Apex Classes

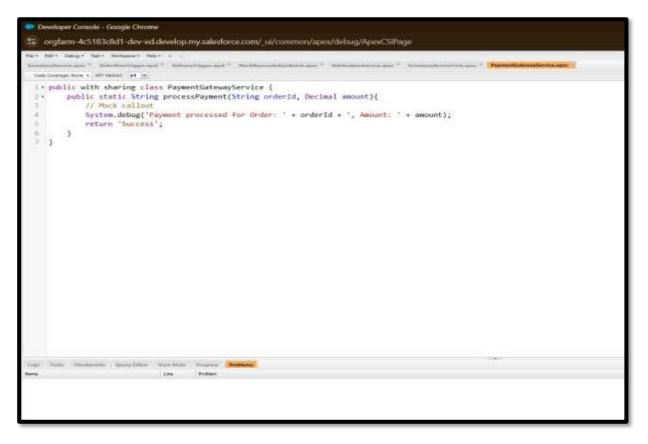
InventoryService

- Purpose: Check and adjust product stock when order items are created/removed.
 Performs validation (prevents oversell) and updates Product/Inventory records.
- Key methods: adjustStock(List<Order_Item_c> items, Boolean isInsert, Boolean isDelete) (example).

```
The Content No. | When the State of the Stat
```

• PaymentGatewayService (utility/demo class)

 Purpose: Mock payment gateway callout (simulated processPayment method) to demonstrate how payment integration would be invoked from Apex/Flow.



• NotificationService

 Purpose: Asynchronous notifications. Contains @future method to simulate sending SMS/WhatsApp or external notifications.

```
December Controls Google Chronics

orgians deStilloft-dec-ed-december developing any advantance cont. st/cammon/apox/dechang/Apox/SIPage

set Side State Sta
```

• (Optional) LogisticsService

o Purpose: Mock REST callout for logistics updates (used by Phase 7 mocks).

```
Developer Councie - Google Chroma

True That : Takes the developer of the control of the control
```

Apex Triggers

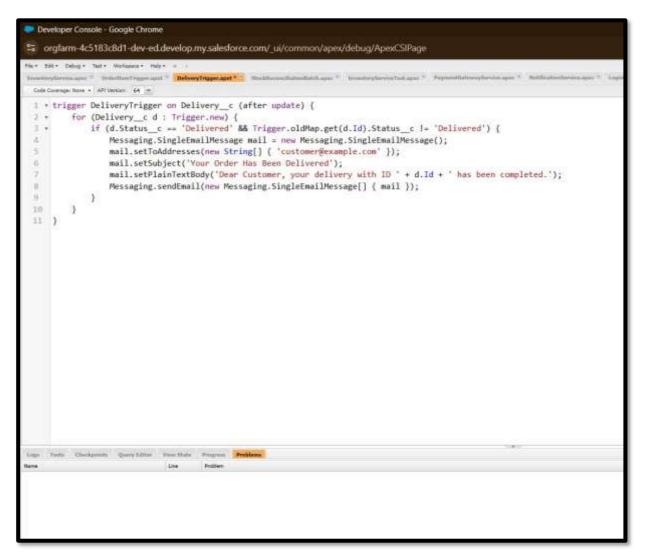
- OrderItemTrigger on Order_Item__c
 - Events: before insert (validate stock availability), after insert (adjust stock), after delete (restore stock).
 - Purpose: Prevents creating order items if stock unavailable and updates inventory after successful insert/delete.

```
Obveloped Console - Geogle Chrome

Statistic - Console - Geogle Chrome

The Statistic - Console - Console
```

- DeliveryTrigger on Delivery__c
 - o Events: after update
 - Purpose: When Status_c changes to Delivered, sends email/notification to the customer (demo sends email or logs an outgoing message).



Asynchronous Apex

- **StockReconciliationBatch** (implements Database.Batchable<sObject>)
 - Purpose: Nightly job to reconcile and sanitize product stock (e.g., set negative stock to 0, recalc aggregates).
 - How to run: Database.executeBatch(new StockReconciliationBatch(), 200);

```
File - Edit - Debug - Test - Workspace - Help - < >
InventoryService.apric * StockReconciliationBatch.apric *
 Code Coverage: None + API Version: 64 ×
 1 * global class StockReconciliationBatch implements Database.Batchable<sObject> {
         global Database.QueryLocator start(Database.BatchableContext bc) {
             return Database.getQueryLocator('SELECT Id, Stock_c FROM Product_c');
 3
 4
 5
        global void execute(Database.BatchableContext bc, List<Product_c> scope) {
 6 *
 7 +
              for (Product_c p : scope) {
 8 +
                  if (p.Stock_c < 0) {
                      p.Stock_c = 0; // reset invalid stock
 9
 18
 11
              }
 12
              update scope;
 13
        }
 14
 15 +
         global void finish(Database.BatchableContext bc) {
              System.debug('Stock reconciliation completed.');
 16
 17
          1
 18 }
Logs Tests Checkgoods Query Editor View State Progress Problems
```

- NotificationService.sendSMS (@future)
 - o Purpose: Ofload external notification calls.

```
December Consider - Google Chromes

To orgination - Cologie Chrome
```

Testing & Coverage

- **Test classes** created (examples):
 - InventoryServiceTest sets up Product, Customer, Order and asserts stock adjustments after inserting an Order_Item_c.

```
** > G 14 agriedgeor-mangetid-dev-ed.develop.my.salesforce.com/_u/common/apes/debug/ApexL58/age
Mar Side Dange Sale Statement Sales a
  Code Coverage: Nove + API Version - 64 (%)
  1 #isTest
  2 * private class InventoryServiceTest (
           @testSetup
           static void setupOata() (
                // Corrected: Create a standard Account record.
                Account acc = new Account(Name='Test Customer Account');
               // Corrected: Numered the num-writeable 'Name_c' field.
// Also, the Unit_s value is converted to a String.
  10
  11
                Product_c prod = new Product_c(Unit_c='50');
  12
13
14
15
16
17
18
                insert prod;
                // Corrected: The Order_c.Customer_c lookup field is populated with the Account's Id.
Order_c und = new Order_c(Customer_c:acc.Id, Payment_Mode_c:"Online');
insert ord;
          static testMethod void testOrderItemTrigger() {
               // Corrected: Querying the Unit_c field, not Stock_c.
Product_c p = [SELECT Id, Unit_c FROM Product_c LDMIT 1];
// Corrected: Querying the custom object Order1_c.
  Order_c n = [SELECT 1d FROM Order_c LIMIT 1];
                Test_startTest();
  26
27
28
                // Corrected: Using the correct object name Order_Line_Item_c.
// Corrected: Using UnitPrice field instead of the non-existent Price_c field.
```

- DeliveryTriggerTest verifies email/notification is sent when Delivery status updates to Delivered.
- o StockReconciliationBatchTest simulates negative stock and asserts batch fixes it.
- Custom Exceptions implemented in Apex to handle business errors (e.g., InsufficientStockException).
- **Reported coverage:** Test classes were written to achieve >75% code coverage for the Apex components (please run tests in your org to confirm actual coverage percentages).