Guest360 CRM Project – Phase 5: Apex Programming (Developer)

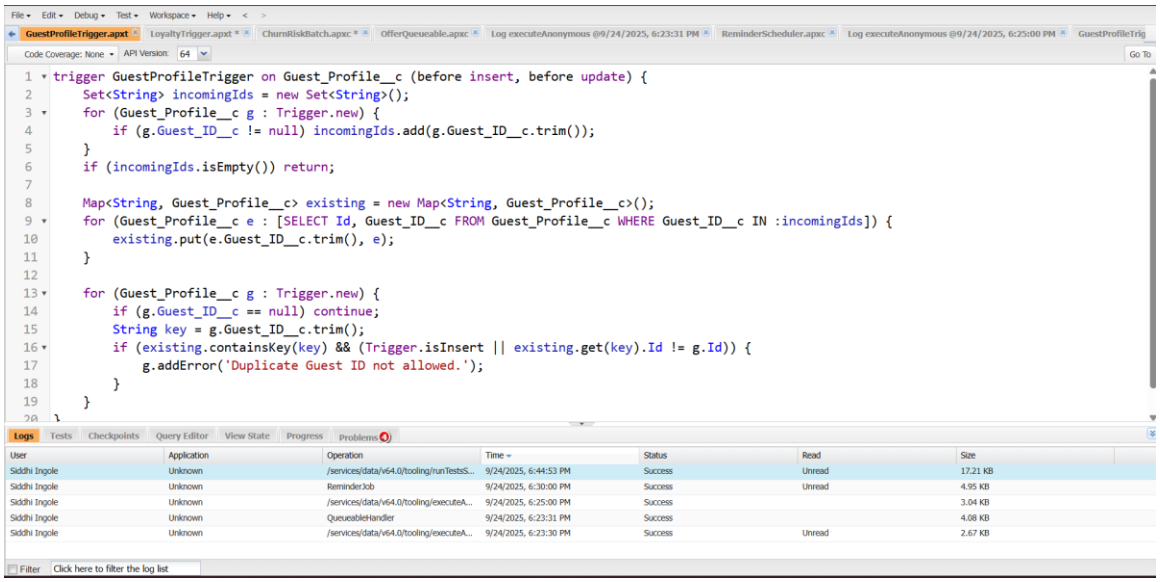
This document explains Phase 5 (Apex Programming) of the Guest360 CRM Project. It provides step-by-step click instructions with code and screenshot placeholders.

Step 1: Prevent Duplicate Guest IDs (Trigger)

Path: **Setup → Object Manager → Guest Profile → Triggers → New**

Steps:

1. Click New → Name: GuestProfileTrigger → sObject: Guest_Profile__c

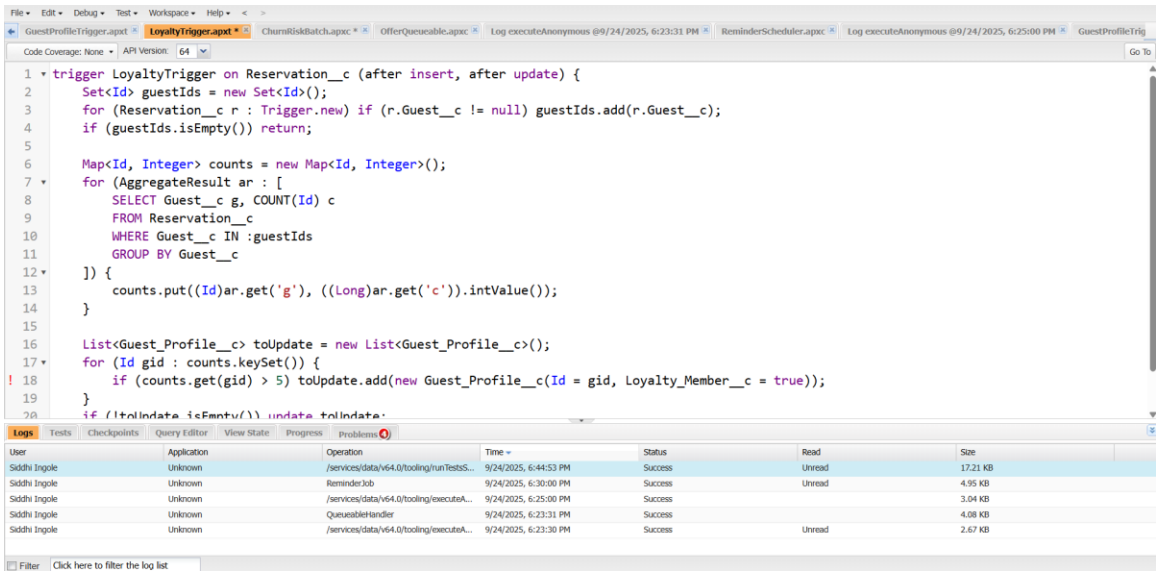


Step 2: Personalize Loyalty (Trigger)

Path: Setup → Object Manager → Reservation → Triggers → New

Steps:

1. Name: LoyaltyTrigger → sObject: Reservation__c



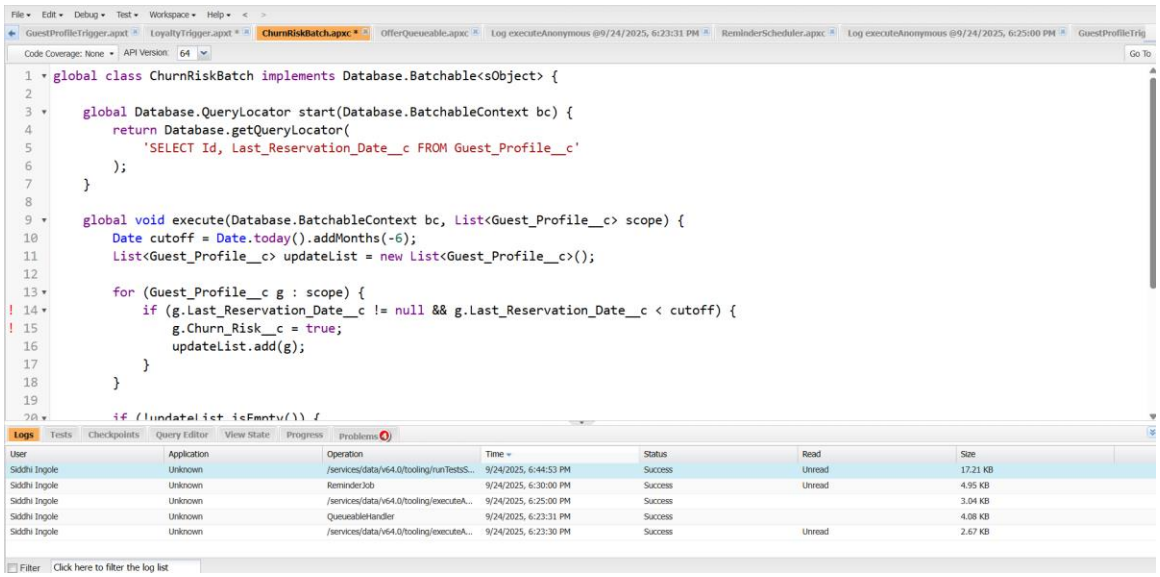
```
trigger LoyaltyTrigger on Reservation__c (after insert, after update) {
    Set<Id> guestIds = new Set<Id>();
    for (Reservation__c r : Trigger.new) if (r.Guest__c != null) guestIds.add(r.Guest__c);
    if (guestIds.isEmpty()) return;

    Map<Id, Integer> counts = new Map<Id, Integer>();
    for (AggregateResult ar : [
        SELECT Guest__c g, COUNT(Id) c
        FROM Reservation__c
        WHERE Guest__c IN :guestIds
        GROUP BY Guest__c
    ]) {
        counts.put((Id)ar.get('g'), ((Long)ar.get('c')).intValue());
    }

    List<Guest_Profile__c> toUpdate = new List<Guest_Profile__c>();
    for (Id gid : counts.keySet()) {
        if (counts.get(gid) > 5) toUpdate.add(new Guest_Profile__c(Id = gid, Loyalty_Member__c = true));
    }
```

Step 3: Batch Apex – Churn Risk Analysis

Path: Setup → Apex Classes → New

Steps:

1. Name: ChurnRiskBatch



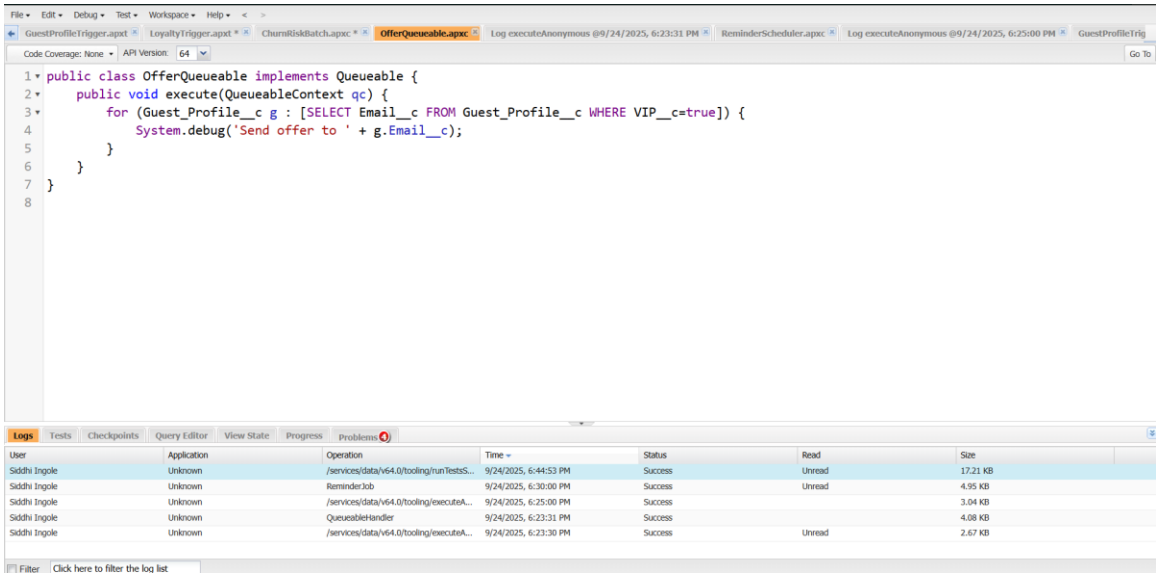
```
global class ChurnRiskBatch implements Database.Batchable<sObject> {

    global Database.QueryLocator start(Database.BatchableContext bc) {
        return Database.getQueryLocator(
            'SELECT Id, Last_Reservation_Date__c FROM Guest_Profile__c'
        );
    }

    global void execute(Database.BatchableContext bc, List<Guest_Profile__c> scope) {
        Date cutoff = Date.today().addMonths(-6);
        List<Guest_Profile__c> updateList = new List<Guest_Profile__c>();

        for (Guest_Profile__c g : scope) {
            if (g.Last_Reservation_Date__c != null && g.Last_Reservation_Date__c < cutoff) {
                g.Churn_Risk__c = true;
                updateList.add(g);
            }
        }
```

Save and run: Debug → Open Execute Anonymous → Database.executeBatch(new ChurnRiskBatch(), 200)

Step 4: Queueable Apex – Send Offers

Path: **Setup → Apex Classes → New**

Steps:

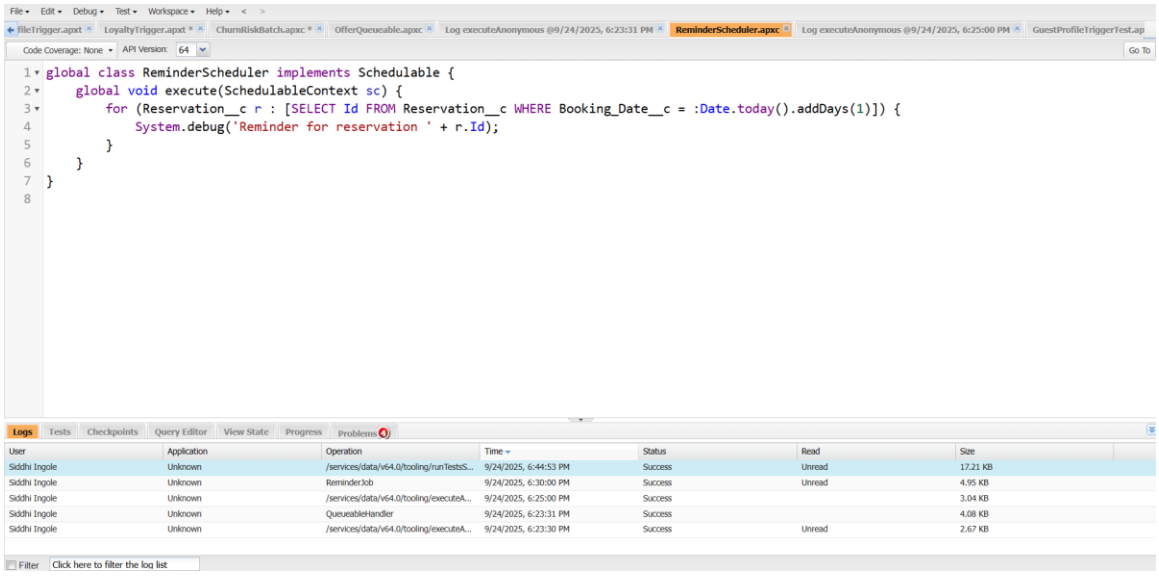
1. Name: OfferQueueable
2. 

Save and run: Execute Anonymous → System.enqueueJob(new OfferQueueable())

Step 5: Scheduled Apex – Reminders

Path: Setup → Apex Classes → New

Steps:



Save and schedule: Setup → Apex Classes → Schedule Apex → Select ReminderScheduler → Daily.

Step 6: Test Class

Path: Setup → Apex Classes → New

Steps:

1. Name: GuestProfileTriggerTest

```
@isTest
private class GuestProfileTriggerTest {
  @isTest static void testDuplicate() {
    Guest_Profile__c g1 = new Guest_Profile__c(Name='Test1', Guest_ID__c='X001');
    insert g1;
    Guest_Profile__c g2 = new Guest_Profile__c(Name='Test2', Guest_ID__c='X001');
    Test.startTest();
    try { insert g2; System.assert(false); } catch (DmlException e) { }
    Test.stopTest();
  }
```

}

2. Save → Test → New Run → Select GuestProfileTriggerTest → Run.