Comprehensive Referral System Test Specification

Complete Testing Framework for Partner/Influencer/Customer Pathways

📋 Test Overview

This specification covers all referral pathways, account creation flows, commission calculations, discount applications, and cross-channel interactions in the Pawtraits referral ecosystem.

🧪 Test Environment Setup

Required Test Accounts

Admin Test Accounts

```
ADMIN_TEST_001
- Email: admin.test@pawtraits.com
- Role: Admin
- Purpose: Create QR codes, manage partners/influencers, view analytics
- Permissions: Full system access

ADMIN_TEST_002
- Email: admin.campaigns@pawtraits.com
- Role: Admin
- Purpose: Campaign management and financial oversight
- Permissions: Full system access
```

Partner Test Accounts

```
PARTNER_TEST_001
- Business: Happy Paws Grooming
- Email: groomer.test@pawtraits.com
- QR Code: PART-GROOM001
- Status: Not yet registered
- Purpose: Test partner registration flow

PARTNER_TEST_002
- Business: City Veterinary Clinic
- Email: vet.test@pawtraits.com
- QR Code: PART-VET001
- Status: Pre-approved
- Purpose: Test active partner commissions

PARTNER_TEST_003
- Business: Pet Paradise Store
- Email: store.test@pawtraits.com
- QR Code: PART-STORE001
- Status: Suspended
- Purpose: Test suspended partner scenarios
```

Influencer Test Accounts

```
INFLUENCER_TEST_001
- Display Name: Pet Lover Jessica
- Email: jessica.influencer@pawtraits.com
- Instagram: @testpetlover_jess
- Status: Application submitted
- Purpose: Test influencer approval flow

INFLUENCER_TEST_002
- Display Name: Fluffy Tails Creator
- Email: fluffy.creator@pawtraits.com
- TikTok: @fluffytails_test
- Status: Approved
- Referral Code: INFL-FLUFFY001
- Purpose: Test active influencer commissions

INFLUENCER_TEST_003
- Display Name: Dog Whisperer Pro
- Email: dogpro.influencer@pawtraits.com
- Instagram: @dogwhisperer_test
- Status: Suspended
- Purpose: Test suspended influencer scenarios
```

Customer Test Accounts

```
CUSTOMER_TEST_001
- Name: Sarah Mitchell
- Email: sarah.customer@pawtraits.com
- Referral Code: CUST-SARAH001
- Purpose: Test customer referral creation

CUSTOMER_TEST_002
- Name: Mike Rodriguez
- Email: mike.customer@pawtraits.com
- Purpose: Test friend referral signup

CUSTOMER_TEST_003
- Name: Lisa Thompson
- Email: lisa.customer@pawtraits.com
- Purpose: Test influencer follower signup

CUSTOMER_TEST_004
- Name: James Wilson
- Email: james.customer@pawtraits.com
- Purpose: Test partner QR code signup

CUSTOMER_TEST_005
- Name: Emma Davis
- Email: emma.customer@pawtraits.com
- Purpose: Test multi-generational referrals
```

Test Payment Methods

```
TEST_CARD_SUCCESS: 4242424242424242 (Visa)
```

```
TEST_CARD_DECLINE: 4000000000000002 (Visa)
TEST_CARD_INSUFFICIENT: 4000000000009995 (Visa)
Expiry: Any future date
CVC: Any 3 digits
```

🧪 Test Scenarios

Test Suite 1: Partner Registration & Commission Flow

Test Case 1.1: Partner QR Code Registration

Objective: Test partner registration via QR code scan

Pre-conditions:

- QR code PART-GROOM001 exists in system
- No partner account exists for groomer.test@pawtraits.com

Test Steps:

1. Navigate to `/referral/PART-GROOM001`
2. Verify redirect to partner signup page
3. Fill partner registration form:
   - Business Name: "Happy Paws Grooming"
   - Contact Name: "Test Groomer"
   - Email: "groomer.test@pawtraits.com"
   - Phone: "+44 1234 567890"
   - Business Type: "Groomer"
4. Submit registration
5. Admin approves partner application
6. Partner receives approval email

Expected Results:

- ✅ Partner account created with status "pending"
- ✅ QR code status changes from "pre_registration" to "partner_registered"
- ✅ Partner receives registration confirmation email
- ✅ After admin approval, partner can access dashboard
- ✅ QR code becomes active for customer use

Database Verification:

```
-- Check partner record
SELECT * FROM partners WHERE email = 'groomer.test@pawtraits.com';

-- Check QR code status
SELECT status FROM pre_registration_codes WHERE code = 'PART-GROOM001';

-- Verify linkage
```

```
SELECT partner_id FROM pre_registration_codes WHERE code = 'PART-GROOM001';
```

Test Case 1.2: Customer Signup via Partner QR Code

Objective: Test customer account creation through partner QR code

Pre-conditions:

- Partner PART-GROOM001 is approved and active
- No customer account exists for james.customer@pawtraits.com

Test Steps:

1. Navigate to `/referral/PART-GROOM001`
2. Verify redirect to customer signup with 20% discount banner
3. Fill customer registration form:
   - Name: "James Wilson"
   - Email: "james.customer@pawtraits.com"
   - Password: "TestPass123!"
4. Submit registration
5. Verify 20% discount is automatically applied to account
6. Create customer lineage record

Expected Results:

- ✅ Customer account created successfully
- ✅ 20% discount available for first order
- ✅ Customer linked to partner PART-GROOM001
- ✅ Customer lineage record created with partner as root referrer
- ✅ Welcome email sent to customer

Database Verification:

```
-- Check customer account
SELECT * FROM customers WHERE email = 'james.customer@pawtraits.com';

-- Check lineage tracking
SELECT * FROM customer_acquisition_lineage WHERE customer_id =
[james_customer_id];

-- Verify discount availability
SELECT * FROM customer_discounts WHERE customer_id = [james_customer_id]
AND discount_type = 'first_order_partner';
```

Test Case 1.3: Partner Commission Calculation - First Order

Objective: Test 20% commission calculation on customer's first order

Pre-conditions:

- Customer CUSTOMER_TEST_004 (James) linked to PARTNER_TEST_002

- Customer has 20% first-order discount available

Test Steps:

1. Login as James Wilson (james.customer@pawtraits.com)
2. Add Golden Retriever portrait (£50) to cart
3. Proceed to checkout
4. Verify 20% discount applied: £50 → £40
5. Complete payment with TEST_CARD_SUCCESS
6. Verify order confirmation

Expected Results:

- ✅ Customer pays £40 (£50 - £10 discount)
- ✅ Partner earns £10 commission (20% of £50 original value)
- ✅ Order status: "completed"
- ✅ Commission status: "earned"
- ✅ Partner commission balance updated

Database Verification:

```
-- Check order details
SELECT total, discount_amount FROM orders WHERE customer_id = [james_id];

-- Check commission earned
SELECT commission_amount, commission_rate
FROM referral_commissions
WHERE referral_id IN (
    SELECT id FROM referrals WHERE referee_customer_id = [james_id]
);

-- Check partner earnings
SELECT total_commission_earned FROM partners WHERE email =
'vet.test@pawtraits.com';
```

Test Case 1.4: Partner Commission - Subsequent Order

Objective: Test 5% commission on customer's subsequent orders

Pre-conditions:

- Customer James has completed first order
- Partner has received 20% commission on first order

Test Steps:

1. Login as James Wilson
2. Add Cat Portrait (£40) to cart
3. Proceed to checkout (no discount applied - not first order)
4. Complete payment: £40
5. Verify commission calculation

Expected Results:

- ✅ Customer pays full £40 (no discount)
- ✅ Partner earns £2 commission (5% of £40)
- ✅ Commission rate correctly applied as 5%
- ✅ Partner's total earnings updated

Database Verification:

```
-- Check second order commission
SELECT commission_amount, commission_rate, is_first_order
FROM referral_commissions
WHERE order_id = [second_order_id];

-- Verify commission rate is 5%
SELECT commission_rate FROM referral_commissions
WHERE order_id = [second_order_id];
-- Expected: 0.05
```

Test Suite 2: Customer Referral System

Test Case 2.1: Customer Referral Code Generation

Objective: Test automatic generation of customer referral codes

Pre-conditions:

- Customer CUSTOMER_TEST_001 (Sarah) has active account

Test Steps:

1. Login as Sarah Mitchell
2. Navigate to referral section in dashboard
3. Verify referral code is displayed
4. Test code uniqueness across system

Expected Results:

- ✅ Referral code format: CUST-XXXXX (5 random characters)
- ✅ Code is unique across all referral types
- ✅ Code is immediately available after account creation
- ✅ Share options are displayed (email, text, social media)

Database Verification:

```
-- Check referral code exists
SELECT referral_code FROM customer_referral_codes
WHERE customer_id = [sarah_id];

-- Verify uniqueness
```

```
SELECT COUNT(*) FROM (
    SELECT referral_code FROM customer_referral_codes
    UNION ALL
    SELECT referral_code FROM partners
    UNION ALL
    SELECT referral_code FROM influencers
) WHERE referral_code = [sarah_code];
-- Expected: 1
```

Test Case 2.2: Friend Signup via Customer Referral

Objective: Test friend account creation using customer referral code

Pre-conditions:

- Customer Sarah has referral code CUST-SARAH001
- No account exists for mike.customer@pawtraits.com

Test Steps:

1. Navigate to `/referral/CUST-SARAH001`
2. Verify friend signup page with referrer information
3. Fill friend registration form:
    - Name: "Mike Rodriguez"
    - Email: "mike.customer@pawtraits.com"
    - Password: "FriendPass123!"
4. Submit registration
5. Verify 20% first-order discount applied

Expected Results:

- ✅ Mike's account created with 20% first-order discount
- ✅ Customer referral record created linking Sarah → Mike
- ✅ Referral status: "signup" (not yet "purchased")
- ✅ Sarah does not yet receive credit (awaiting Mike's purchase)
- ✅ Mike gets welcome email with discount information

Database Verification:

```
-- Check referral relationship
SELECT referrer_customer_id, referee_customer_id, status
FROM customer_referrals
WHERE referral_code = 'CUST-SARAH001';

-- Verify Sarah hasn't earned credit yet
SELECT COUNT(*) FROM customer_discount_rewards
WHERE customer_id = [sarah_id];
-- Expected: 0

-- Check Mike's first-order discount
SELECT discount_percentage FROM customer_discounts
WHERE customer_id = [mike_id] AND discount_type = 'first_order_referral';
```

Test Case 2.3: Customer Credit Reward on Friend's Purchase

Objective: Test fixed credit reward when referred friend makes first purchase

Pre-conditions:

- Mike is referred by Sarah (CUST-SARAH001)
- Mike has 20% first-order discount
- Sarah has not yet received credit

Test Steps:

1. Login as Mike Rodriguez
2. Add Dog Portrait (£60) to cart
3. Verify 20% discount applied: £60 → £48
4. Complete payment with TEST_CARD_SUCCESS
5. Check Sarah's account for earned credit

Expected Results:

- ✅ Mike pays £48 (£60 - £12 discount)
- ✅ Sarah earns £12 credit (20% of Mike's £60 order value)
- ✅ Customer referral status changes to "purchased"
- ✅ Sarah receives notification of earned credit
- ✅ Credit is immediately available for use

Database Verification:

```
-- Check referral status updated
SELECT status, referee_first_order_value
FROM customer_referrals
WHERE referrer_customer_id = [sarah_id] AND referee_customer_id =
[mike_id];

-- Check Sarah's earned credit
SELECT credit_amount, earned_from_order_value
FROM customer_discount_rewards
WHERE customer_id = [sarah_id];
-- Expected: credit_amount = 12.00, earned_from_order_value = 60.00
```

Test Case 2.4: Customer Credit Redemption

Objective: Test application of earned credits to customer's order

Pre-conditions:

- Sarah has £12 credit from Mike's referral
- Credit status is "earned"

Test Steps:

1. Login as Sarah Mitchell
2. Add Cat Portrait (£45) to cart
3. Proceed to checkout
4. Verify £12 credit is available for application
5. Apply credit to reduce order total
6. Complete payment

Expected Results:

- ✅ Order total: £45 - £12 = £33
- ✅ Credit status changes to "applied"
- ✅ Sarah pays £33
- ✅ Credit balance reduces to £0
- ✅ Order completes successfully

Database Verification:

```
-- Check credit application
SELECT status, actual_credit_applied
FROM customer_discount_rewards
WHERE customer_id = [sarah_id];

-- Check order total
SELECT total, credits_applied
FROM orders
WHERE customer_id = [sarah_id] AND id = [latest_order_id];
```

Test Suite 3: Influencer System

Test Case 3.1: Influencer Application and Approval

Objective: Test influencer application submission and approval process

Pre-conditions:

- No influencer account exists for jessica.influencer@pawtraits.com

Test Steps:

1. Navigate to `/signup/influencer`
2. Fill influencer application:
   - Display Name: "Pet Lover Jessica"
   - Email: "jessica.influencer@pawtraits.com"
   - Primary Platform: "Instagram"
   - Handle: "@testpetlover_jess"
   - Follower Count: 45200
   - Pet 1: "Max, Golden Retriever, 3"
   - Pet 2: "Bella, Labrador, 5"
3. Submit application
4. Admin reviews and approves application

5. Check approval benefits

Expected Results:

- ✅ Application submitted with status "pending"
- ✅ Admin can review application with scoring interface
- ✅ Upon approval, influencer receives:
    - $100 credit automatically added
    - Unique referral code (INFL-JESSICA001)
    - Access to influencer dashboard
    - Approval confirmation email
- ✅ Influencer can login and access dashboard

Database Verification:

```
-- Check influencer record
SELECT approval_status, referral_code
FROM influencers
WHERE email = 'jessica.influencer@pawtraits.com';

-- Check $100 credit added
SELECT credit_amount, credit_type
FROM influencer_credits
WHERE influencer_id = [jessica_id];
-- Expected: 100.00, 'signup_bonus'
```

Test Case 3.2: Follower Signup via Influencer Code

Objective: Test follower account creation with lifetime 10% discount

Pre-conditions:

- Influencer INFLUENCER_TEST_002 is approved with code INFL-FLUFFY001
- No account exists for lisa.customer@pawtraits.com

Test Steps:

1. Navigate to `/referral/INFL-FLUFFY001`
2. Verify follower signup page with influencer information
3. Fill follower registration:
    - Name: "Lisa Thompson"
    - Email: "lisa.customer@pawtraits.com"
    - Password: "FollowerPass123!"
4. Submit registration
5. Verify lifetime discount setup

Expected Results:

- ✅ Lisa's account created successfully
- ✅ 10% lifetime discount applied to account
- ✅ Influencer referral record created

- ✅ Lisa linked to Fluffy Tails Creator as referrer
- ✅ Welcome email mentions lifetime 10% discount

Database Verification:

```
-- Check follower account
SELECT * FROM customers WHERE email = 'lisa.customer@pawtraits.com';

-- Check lifetime discount
SELECT discount_percentage, is_lifetime
FROM customer_discounts
WHERE customer_id = [lisa_id] AND discount_type = 'influencer_follower';

-- Check influencer referral
SELECT influencer_id, customer_id
FROM influencer_referrals
WHERE referral_code = 'INFL-FLUFFY001';
```

Test Case 3.3: Influencer Commission on Follower Purchase

Objective: Test 10% lifetime commission on follower purchases

Pre-conditions:

- Lisa is a follower of INFL-FLUFFY001
- Lisa has 10% lifetime discount
- Influencer has 10% lifetime commission rate

Test Steps:

1. Login as Lisa Thompson
2. Add Premium Dog Portrait (£80) to cart
3. Verify 10% discount applied: £80 → £72
4. Complete payment with TEST_CARD_SUCCESS
5. Check influencer commission earned

Expected Results:

- ✅ Lisa pays £72 (£80 - £8 discount)
- ✅ Influencer earns £8 commission (10% of £80)
- ✅ Commission status: "earned"
- ✅ Influencer total earnings updated
- ✅ Referral marked as "converted"

Database Verification:

```
-- Check influencer commission
SELECT commission_earned, commission_rate
FROM influencer_referrals
WHERE customer_id = [lisa_id];

-- Check total influencer earnings
```

```
SELECT total_commission_earned
FROM influencers
WHERE referral_code = 'INFL-FLUFFY001';
```

Test Case 3.4: Lifetime Discount on Repeat Orders

Objective: Test that 10% discount applies to ALL follower orders

Pre-conditions:

- Lisa has completed first order with 10% discount
- Influencer earned commission on first order

Test Steps:

1. Login as Lisa Thompson (30 days later)
2. Add Cat Portrait (£50) to cart
3. Verify 10% discount still applied: £50 → £45
4. Complete second payment
5. Verify influencer earns commission on repeat order

Expected Results:

- ✅ Lisa pays £45 (£50 - £5 discount) on repeat order
- ✅ Influencer earns £5 commission (10% of £50)
- ✅ Lifetime discount continues working
- ✅ Lifetime commission continues working
- ✅ Both parties benefit from ongoing relationship

Database Verification:

```
-- Check second order discount
SELECT total, discount_amount
FROM orders
WHERE customer_id = [lisa_id] AND id = [second_order_id];

-- Check second commission
SELECT commission_earned
FROM influencer_referrals
WHERE customer_id = [lisa_id] AND order_id = [second_order_id];
```

Test Suite 4: Multi-Generational Tracking

Test Case 4.1: Second-Generation Customer Referral

Objective: Test lineage tracking when referred customer refers someone else

Pre-conditions:

- Mike was referred by Sarah (customer referral)

- Mike has completed his first order
- No account exists for emma.customer@pawtraits.com

Test Steps:

1. Login as Mike Rodriguez
2. Navigate to referral dashboard
3. Note Mike's referral code: CUST-MIKE001
4. Navigate to `/referral/CUST-MIKE001` (as Emma)
5. Create Emma's account via Mike's referral
6. Emma completes first order
7. Verify lineage tracking back to Sarah

Expected Results:

- ✅ Emma account created via Mike's referral
- ✅ Mike earns credit from Emma's purchase
- ✅ Lineage tracking shows: Sarah → Mike → Emma
- ✅ Sarah is recorded as "root referrer" for Emma
- ✅ Generation depth recorded as 2 for Emma

Database Verification:

```
-- Check Emma's lineage
SELECT
    direct_referrer_id,
    root_referrer_id,
    generation_depth,
    referral_path
FROM customer_acquisition_lineage
WHERE customer_id = [emma_id];

-- Expected: direct_referrer_id = mike_id, root_referrer_id = sarah_id,
generation_depth = 2
```

Test Case 4.2: Partner Viral Impact Tracking

Objective: Test attribution of multi-generation customers back to original partner

Pre-conditions:

- James was acquired via Partner PART-VET001
- James referred Tom (generation 1)
- Tom referred Amy (generation 2)
- Amy places an order

Test Steps:

1. Create referral chain: Partner → James → Tom → Amy
2. Amy places £60 order
3. Check revenue attribution back to original partner

4. Verify viral multiplier calculations

Expected Results:

- ✅ Amy's purchase attributed to Partner PART-VET001
- ✅ Revenue attribution with generation weighting applied
- ✅ Partner viral impact dashboard shows multi-generation revenue
- ✅ ROI calculation includes viral effects

Database Verification:

```
-- Check revenue attribution
SELECT
    root_referrer_type,
    root_referrer_id,
    generation_depth,
    attributed_revenue
FROM revenue_attribution
WHERE customer_id = [amy_id];

-- Check partner viral impact
SELECT
    total_customers,
    viral_customers,
    viral_revenue
FROM partner_viral_impact
WHERE partner_id = [vet_partner_id];
```

Test Suite 5: Cross-Channel Integration

Test Case 5.1: Influencer Follower Becomes Customer Referrer

Objective: Test when influencer follower starts referring friends

Pre-conditions:

- Lisa was acquired via Influencer INFL-FLUFFY001
- Lisa has made purchases and is happy customer
- Lisa wants to refer her friend Kate

Test Steps:

1. Lisa refers Kate using her customer referral code CUST-LISA001
2. Kate signs up and makes purchase
3. Verify multiple attribution:
   - Lisa earns credit from Kate's purchase
   - Influencer continues earning from Lisa's relationship
4. Check lineage tracking

Expected Results:

- ✅ Kate linked to Lisa as direct referrer
- ✅ Kate linked to Influencer as root referrer
- ✅ Lisa earns customer referral credit
- ✅ Influencer continues earning from Lisa
- ✅ Cross-channel attribution maintained

Database Verification:

```
-- Check Kate's dual attribution
SELECT * FROM customer_acquisition_lineage WHERE customer_id = [kate_id];

-- Check Lisa's earnings from referral
SELECT * FROM customer_discount_rewards WHERE customer_id = [lisa_id];

-- Check influencer still earning from Lisa
SELECT * FROM influencer_referrals WHERE customer_id = [lisa_id];
```

Test Suite 6: Financial Integration & Edge Cases

Test Case 6.1: Payment Failure Scenarios

Objective: Test commission/credit behavior when payments fail

Pre-conditions:

- Customer has earned credits/discounts
- Partner/Influencer should earn commission

Test Steps:

1. Apply credits/discounts to order
2. Attempt payment with TEST_CARD_DECLINE
3. Verify credits/commissions are not awarded
4. Retry with successful payment
5. Verify proper crediting occurs

Expected Results:

- ✅ Failed payment does not award commissions
- ✅ Credits/discounts remain available after failure
- ✅ Successful retry properly awards all benefits
- ✅ No duplicate commissions created

Test Case 6.2: Refund Impact on Commissions

Objective: Test commission reversal when orders are refunded

Pre-conditions:

- Partner/Influencer earned commission on completed order
- Customer requests refund

Test Steps:

1. Process refund for order with commissions
2. Verify commission reversal
3. Check partner/influencer balance updates

Expected Results:

- ✅ Commissions reversed from partner/influencer accounts
- ✅ Customer credits reversed if applicable
- ✅ Financial records maintain accuracy

Test Case 6.3: Account Suspension Impact

Objective: Test referral behavior when accounts are suspended

Pre-conditions:

- Partner/Influencer with active referrals gets suspended

Test Steps:

1. Suspend partner/influencer account
2. Test QR code/referral link behavior
3. Verify existing customer relationships

Expected Results:

- ✅ New referrals blocked for suspended accounts
- ✅ Existing customer relationships maintained
- ✅ Appropriate error messages displayed

Test Suite 7: Organic Customer Acquisition

Test Case 7.1: Direct Customer Signup

Objective: Test customer account creation without any referral codes

Pre-conditions:

- No referral code in URL
- No QR code scanning

- Customer finds site through organic search/marketing

Test Steps:

1. Navigate directly to `/signup` (no referral parameters)
2. Fill customer registration form:
    - Name: "Alex Johnson"
    - Email: "alex.organic@pawtraits.com"
    - Password: "OrganicUser123!"
3. Submit registration
4. Verify NO discounts applied
5. Verify NO lineage record created

Expected Results:

- ✅ Customer account created successfully
- ✅ NO discounts available (standard pricing only)
- ✅ NO customer_acquisition_lineage record
- ✅ referrer_id fields remain NULL
- ✅ Customer marked as "organic" acquisition
- ✅ Standard welcome email (no discount messaging)

Database Verification:

```
-- Check customer account with no referral data
SELECT * FROM customers WHERE email = 'alex.organic@pawtraits.com';

-- Verify NO lineage tracking
SELECT COUNT(*) FROM customer_acquisition_lineage
WHERE customer_id = [alex_id];
-- Expected: 0

-- Verify NO discounts
SELECT COUNT(*) FROM customer_discounts
WHERE customer_id = [alex_id];
-- Expected: 0
```

Test Case 7.2: Organic Customer First Order

Objective: Test standard pricing without any discounts or commissions

Pre-conditions:

- Alex is organic customer (no referral history)
- No discounts available

Test Steps:

1. Login as Alex Johnson
2. Add Dog Portrait (£50) to cart
3. Proceed to checkout

4. Verify NO discount options shown
5. Complete payment with standard pricing
6. Verify NO commissions generated

Expected Results:

- ✅ Customer pays full £50 (no discounts)
- ✅ Order status: "completed"
- ✅ NO commission records created
- ✅ NO partner/influencer earnings affected
- ✅ Pure revenue (no margin reduction)

Database Verification:

```
-- Check order at full price
SELECT total, discount_amount FROM orders
WHERE customer_id = [alex_id];
-- Expected: total = 50.00, discount_amount = 0.00

-- Verify NO commissions generated
SELECT COUNT(*) FROM referral_commissions
WHERE order_id = [alex_order_id];
-- Expected: 0

SELECT COUNT(*) FROM influencer_referrals
WHERE customer_id = [alex_id];
-- Expected: 0
```

Test Case 7.3: Organic Customer Repeat Orders

Objective: Test continued standard pricing on subsequent orders

Pre-conditions:

- Alex has completed first organic order
- No referral relationships established

Test Steps:

1. Login as Alex Johnson (30 days later)
2. Add Cat Portrait (£45) to cart
3. Proceed to checkout
4. Verify continued standard pricing
5. Complete second payment

Expected Results:

- ✅ Customer pays full £45 (no discounts)
- ✅ NO commissions on repeat order
- ✅ Continues as organic customer
- ✅ No referral status changes

Test Case 7.4: Organic Customer Becomes Referrer

Objective: Test when organic customer starts referring friends

Pre-conditions:

- Alex is established organic customer
- Alex has made multiple orders

Test Steps:

1. Login as Alex Johnson
2. Navigate to referral dashboard
3. Note Alex's referral code: CUST-ALEX001
4. Friend uses Alex's code to signup
5. Verify Alex earns standard referral credit

Expected Results:

- ✅ Alex can refer friends despite being organic
- ✅ Friend gets 20% discount as normal
- ✅ Alex earns credit when friend purchases
- ✅ Alex's organic status preserved in lineage
- ✅ Friend shows Alex as direct_referrer, NULL root_referrer

Test Case 7.5: Mixed Customer Analytics

Objective: Test analytics differentiation between organic and referred

Pre-conditions:

- Database contains mix of organic and referred customers
- Various order histories exist

Test Steps:

1. Query customer acquisition analytics
2. Verify organic vs referred segmentation
3. Check revenue attribution accuracy
4. Validate commission calculations exclude organic

Expected Results:

- ✅ Clear separation of organic vs referred customers
- ✅ Revenue attribution only for referred customers
- ✅ Commission calculations accurate
- ✅ Analytics show true acquisition sources

Database Verification:

```
-- Customer segmentation query
SELECT
  CASE
    WHEN cal.customer_id IS NULL THEN 'organic'
    ELSE 'referred'
  END as acquisition_type,
  COUNT(*) as customer_count,
  SUM(o.total) as total_revenue
FROM customers c
LEFT JOIN customer_acquisition_lineage cal ON c.id = cal.customer_id
LEFT JOIN orders o ON c.id = o.customer_id
GROUP BY acquisition_type;
```

Test Suite 8: Baseline System Behavior (MISSING)

Test Case 8.1: URL Direct Access

Objective: Test behavior when accessing site without referral parameters

Test Steps:

1. Navigate to `/signup` directly
2. Navigate to `/` homepage directly
3. Navigate to `/products` directly
4. Verify no referral tracking cookies set
5. Verify standard user experience

Test Case 8.2: Invalid Referral Code Fallback

Objective: Test fallback to organic when referral codes are invalid

Test Steps:

1. Navigate to `/referral/INVALID-CODE`
2. Verify graceful fallback to standard signup
3. Complete registration as organic customer
4. Verify no broken referral records

Test Case 8.3: Expired Referral Code Handling

Objective: Test behavior with expired partner/influencer codes

Test Steps:

1. Navigate to expired partner QR code
2. Verify appropriate error messaging
3. Option to continue as organic customer
4. No broken database records

Priority Assessment

P0 (Critical):

- Test Cases 7.1, 7.2 - Basic organic customer flow
- Mixed analytics verification

P1 (High):

- Repeat order behavior for organic customers
- Invalid code fallback testing

P2 (Medium):

- Organic-to-referrer transition scenarios
- Edge cases with expired codes

Impact of Missing Tests

Without these tests, the system could have:

- ✅ **False discount application** to organic customers
- ✅ **Incorrect commission calculations**
- ✅ **Broken analytics** (can't differentiate acquisition sources)
- ✅ **Database integrity issues** (NULL foreign keys not handled)
- ✅ **Revenue leakage** (unexpected discounts applied)

Implementation Priority

1. **Immediately add Test Suite 7.1 & 7.2** - Core organic customer flow
2. **Add analytics verification** - Ensure proper customer segmentation
3. **Add edge case handling** - Invalid/expired code scenarios
4. **Add database integrity tests** - NULL value handling

📊 Test Execution Matrix

Priority Levels

- **P0 (Critical)**: Account creation, payment processing, commission calculation
- **P1 (High)**: Referral flows, discount application, credit redemption
- **P2 (Medium)**: Analytics, reporting, edge cases
- **P3 (Low)**: UI/UX improvements, nice-to-have features

Test Environment Requirements

- **Database**: Clean test database with sample data
- **Payment Processing**: Stripe test mode enabled
- **Email System**: Email capture/testing system
- **Admin Access**: Full admin privileges for approval workflows
- **Performance**: Ability to simulate multiple concurrent users

Success Criteria

- **100% pass rate** on P0 tests
- **95%+ pass rate** on P1 tests
- **All financial calculations accurate** to penny precision
- **No data leakage** between test accounts
- **Complete audit trail** for all transactions

Reporting Requirements

- Test execution summary with pass/fail rates
- Detailed failure analysis with reproduction steps
- Performance metrics for key user flows
- Security validation results
- Database integrity verification results

🔍 Test Data Validation Queries

Commission Accuracy Validation

```
-- Validate partner commissions match expected rates
SELECT
    p.business_name,
    SUM(CASE WHEN rc.is_first_order = true THEN rc.commission_amount ELSE 0
END) as first_order_commission,
    SUM(CASE WHEN rc.is_first_order = false THEN rc.commission_amount ELSE
0 END) as recurring_commission,
    COUNT(DISTINCT rc.order_id) as total_orders
FROM partners p
JOIN referrals r ON p.id = r.partner_id
JOIN referral_commissions rc ON r.id = rc.referral_id
WHERE p.email LIKE '%.test@pawtraits.com'
GROUP BY p.id, p.business_name;
```

Customer Credit Balance Validation

```
-- Validate customer credit balances
SELECT
    c.name,
    c.email,
    SUM(CASE WHEN cdr.status = 'earned' THEN cdr.credit_amount ELSE 0 END)
as available_credits,
    SUM(CASE WHEN cdr.status = 'applied' THEN cdr.actual_credit_applied
ELSE 0 END) as used_credits
FROM customers c
LEFT JOIN customer_discount_rewards cdr ON c.id = cdr.customer_id
WHERE c.email LIKE '%.customer@pawtraits.com'
GROUP BY c.id, c.name, c.email;
```

Lineage Tracking Validation

```
-- Validate customer lineage chains
SELECT
    c.name as customer,
    cal.generation_depth,
    cal.root_referrer_type,
    COALESCE(p.business_name, i.display_name, c2.name) as root_referrer
FROM customers c
JOIN customer_acquisition_lineage cal ON c.id = cal.customer_id
LEFT JOIN partners p ON cal.root_referrer_id = p.id AND
cal.root_referrer_type = 'partner'
LEFT JOIN influencers i ON cal.root_referrer_id = i.id AND
cal.root_referrer_type = 'influencer'
LEFT JOIN customers c2 ON cal.root_referrer_id = c2.id AND
cal.root_referrer_type = 'customer'
WHERE c.email LIKE '%.customer@pawtraits.com'
ORDER BY cal.generation_depth;
```

This comprehensive test specification ensures complete validation of all referral pathways, financial calculations, and cross-channel integrations in your system. Each test case includes specific verification steps to ensure accuracy and reliability.