

Test Plan – Mobile User Order & Delivery System

1. Document Information

Project Name: Mobile User Order & Delivery System

Role: Test Engineer

Version: 1.0

Date: 2026-01-20

Prepared By: QA / Test Engineering

2. Objective

The objective of this test plan is to validate the functionality, reliability, data integrity, and user experience of the **Cart, Checkout, Order, Address (Set A & Set B), Tracking, and Authentication** modules.

This test plan ensures:

- Orders always have a valid delivery address
 - Address changes persist after refresh
 - Products, totals, and tracking numbers are correctly stored and retrieved
 - System behaves correctly for both API (Postman) and UI
-

3. Scope

In Scope

- User authentication (login/logout)
- Cart management (add/update/delete items)
- Cart total calculation
- Checkout process
- Order creation
- Order items persistence
- Delivery address logic (Set A & Set B)
- Order tracking
- Cancel order (soft delete)
- UI refresh persistence
- REST API validation via Postman

Out of Scope

- Payment gateway validation (real transactions)
 - Performance/load testing
 - Security penetration testing
-

4. Test Environment

Component Value

OS Windows 10 / 11

Server XAMPP (Apache + MySQL)

Backend PHP (PDO)

Frontend HTML, CSS, JavaScript

Database MySQL

API Tool Postman

Browser Chrome / Edge

5. Test Data

- Test users (registered users)
 - Products with price & stock
 - Valid Set A addresses
 - Valid Set B (whitelisted) addresses
 - Multiple cart items
-

6. Assumptions

- User must be logged in
 - Set A address exists before checkout
 - Database tables exist and relationships are correct
 - Sessions are enabled and persistent
-

7. Entry & Exit Criteria

Entry Criteria

- Application deployed on XAMPP
- Database connected successfully
- APIs accessible via browser/Postman

Exit Criteria

- All critical test cases pass
 - No data loss after refresh
 - No blocking or critical defects open
-

8. Test Scenarios & Test Cases

8.1 Authentication

TC ID	Scenario	Expected Result
AUTH-01	Login with valid credentials	Login successful
AUTH-02	Logout	Session destroyed & redirected
AUTH-03	Access API without login	401 Unauthorized

8.2 Cart Management

TC ID	Scenario	Expected Result
CART-01	Add product to cart	Product added
CART-02	Add same product again	Quantity increments
CART-03	Update quantity	Quantity updated
CART-04	Remove product	Item removed
CART-05	Fetch cart	Correct items returned
CART-06	Calculate total	Total = price × quantity

8.3 Checkout

TC ID	Scenario	Expected Result
CHK-01	Checkout with empty cart	Error message
CHK-02	Checkout without Set A address	Block checkout
CHK-03	Successful checkout	Order created
CHK-04	Cart cleared after checkout	Cart empty

8.4 Orders

TC ID	Scenario	Expected Result
ORD-01	Fetch orders	Orders listed
ORD-02	View order items	Items visible
ORD-03	View order total	Correct total
ORD-04	Order refresh	Data persists

8.5 Delivery Address Logic

TC ID	Scenario	Expected Result
ADD-01	Order placed with Set A	Delivery address visible

ADD-02 Delivery Address shows "Not set" before Set A UI indicates missing

ADD-03 Change address using Set B

Address updated

ADD-04 Refresh after Set B change

Address persists

ADD-05 No Set A selected

Checkout blocked

8.6 Tracking

TC ID Scenario Expected Result

TRK-01 Tracking number generated Unique value

TRK-02 Fetch order tracking Correct status

TRK-03 Cancel order Tracking soft-deleted

8.7 Cancel Order

TC ID Scenario Expected Result

CAN-01 Cancel active order Order soft-deleted

CAN-02 Refresh orders Cancelled order hidden

9. API Testing (Postman)

API Method Validation

/cart_api.php GET Items + total

/checkout_api.php POST Order + tracking

/orders_api.php GET Orders + items + address

/orders_api.php PATCH Address updated

/orders_api.php DELETE Order cancelled

10. Defect Management

Severity Description

Critical Data loss, checkout blocked

High Address not persisting

Medium UI not updating

Low Cosmetic issues

11. Risks & Mitigation

Risk	Mitigation
Session loss	Validate session handling
Address inconsistency	Enforce Set A before checkout
Data mismatch	DB constraints & joins

12. Sign-Off

Role Name Date

Product Owner

QA Lead

Developer

Status: READY FOR EXECUTION

I've prepared a **complete, professional Test Plan** for your entire system **as a Test Engineer**, covering:

- Cart
- Checkout
- Orders

- Delivery Address (Set A & Set B)
- Tracking
- Cancel Order
- Login / Logout
- UI + API (Postman)
- Data persistence after refresh

It is now visible in the **canvas** on the right as:

 “Test Plan – Mobile User Order & Delivery System”

What this Test Plan gives you

- Clear **test objectives**
- Defined **scope & assumptions**
- Detailed **test scenarios & test cases**
- API testing matrix (Postman-ready)
- Address logic validation (your Set A / Set B requirement)
- Risks, mitigation, and sign-off section