Test Case Execution Sheet

Project

Mobile User Order & Delivery Management System

Tester

Test Cycle

System Testing / Regression / UAT

Environment

- OS: Windows 10

- Server: XAMPP (Apache + MySQL)

- Browser: Chrome / Edge

- API Tool: Postman

Execution Status Legend

- **PASS** – Test executed successfully

- **FAIL** – Test executed but failed

- **BLOCKED** – Cannot execute due to dependency

- **NOT RUN** – Not executed yet

TC-01: User Login

Field	Value
Test Case ID	TC-01
Feature	Authentication
Scenario	User logs in with valid credentials
Precondition	User exists in DB
Steps	1. Open login page2. Enter valid email/password3. Click Login
Expected Result	User redirected to profile page
Actual Result	
Status	NOT RUN
Defect ID	

TC-02: Add Product to Cart

Field	Value
Test Case ID	TC-02
Feature	Cart

Scenario	Add product to cart
Precondition	User logged in
Steps	1. Call POST /cart_api.php2. Pass product_id
Expected Result	Product added with quantity = 1
Actual Result	
Status	NOT RUN
Defect ID	

TC-03: View Cart & Total Price

Field	Value
Test Case ID	TC-03
Feature	Cart
Scenario	View cart items with total
Precondition	Cart contains items
Steps	1. Call GET /cart_api.php
Expected Result	Items + total_price returned

Actual Result	
Status	NOT RUN
Defect ID	

TC-04: Checkout with Set A- Address

Field	Value
Test Case ID	TC-04
Feature	Checkout
Scenario	Place order using Set A address
Precondition	Cart not empty, Set A address exists
Steps	1. Call POST /checkout_api.php2. Provide card_number & address_id
Expected Result	Order created, tracking number generated
Actual Result	
Status	NOT RUN
Defect ID	

TC-05: Tracking Number Generation

Field	Value
Test Case ID	TC-05
Feature	Order Tracking
Scenario	Verify tracking record
Precondition	Checkout successful
Steps	1. Query order_tracking table
Expected Result	Tracking number exists & status=Processing
Actual Result	
Status	NOT RUN
Defect ID	

TC-06: View Orders with Delivery Address

Field	Value
Test Case ID	TC-06
Feature	Orders

Scenario	Orders show delivery address
Precondition	Order placed
Steps	1. Open orders.php2. Load orders
Expected Result	Delivery Address visible
Actual Result	
Status	NOT RUN
Defect ID	

TC-07: Update Delivery Address via Set B

Field	Value
Test Case ID	TC-07
Feature	Address Update
Scenario	Change delivery address
Precondition	Whitelisted address exists
Steps	1. Click change address link2. Select Set B address
Expected Result	Address updated & persists after refresh

Actual Result	
Status	NOT RUN
Defect ID	

TC-08: Cancel Order (Soft Delete)

Field	Value
Test Case ID	TC-08
Feature	Orders
Scenario	Cancel active order
Precondition	Order exists
Steps	1. Click Cancel Order
Expected Result	Order hidden from UI, deleted_at set
Actual Result	
Status	NOT RUN
Defect ID	

TC-09: Logout

Field	Value
Test Case ID	TC-09
Feature	Authentication
Scenario	User logs out
Precondition	User logged in
Steps	1. Click Logout button
Expected Result	Session destroyed, redirected to login
Actual Result	
Status	NOT RUN
Defect ID	

Tester Sign-Off

Name	Signature	Date

QA Approval

Name   Signature   Date

Your **Test Case Execution Sheet** has been created and is now visible in the canvas 📋

It's written exactly like a **real QA execution artifact**, not just theory, and includes:

- Clear **execution-ready test cases**

- Fields for **Actual Result, Status, Defect ID**

- Covers the **entire flow you built**:

    o Login / Logout

    o Cart + total price

    o Checkout with **Set A delivery address**

    o Tracking number generation

    o Orders UI

    o Address update via **Set B**

    o Cancel order (soft delete)

How you should use this (important for exams / projects)

1. **Duplicate the sheet** per test cycle (System Test, UAT, Regression)

2. Fill in:

   o   Actual Result

   o   Status (PASS / FAIL)

   o   Defect ID (if any)

3. Sign off as **Tester**

4. Get **QA Approval** (even if it's simulated)