

Test Plan – ShopEz Payment Gateway

Introduction

Project Name: ShopEz Payment Gateway

Objective: To validate that the payment gateway processes (Credit/Debit Cards, Digital Wallets, Bank Transfers).

In Scope:

- Functional payment flows
- Error handling
- Security considerations
- Usability of checkout process
- Calculation validation (tax, discounts, shipping)
-

1. Functional Testing

A. Successful Payment scenarios

- Verify successful transaction using:
 - Valid credit/debit card
 - Valid digital wallet
 - Valid bank transfer
- Confirm:
 - Payment confirmation message
 - Order confirmation email or sms sent to user
 - Payment recorded in system database
 - Correct total amount charged for all items

B. Failed Payment Scenarios

1. Insufficient Funds

Expected:

- Clear failure message
- No order created
- User allowed to retry multiple times

3. Cancelled Payment

Expected:

- User redirected back to the checkout page
- Order is not marked as paid

4. Network Interruption During Payment

Expected:

- Transaction status validated on reconnect
- No duplicate charges
- Clear recovery message

C. Price Calculation Validation

Verify:

- Correct tax calculation per tax
- Discount applied correctly
- Shipping fee added properly
- Final total = subtotal + tax + shipping – discount

Edge Case: 100% discount should be considered for items such as giveaway items.

2. Error Handling

When Payment Fails:

Verify:

- Order status should clearly state “Payment Failed”
- User receives clear explanation
- User can retry payment
- Payment method can be changed on checkout before paying

Error Message quality

- The error message should be clear to the user on the issue arising on the system .

Eg “We encountered a problem processing the payment , Please try again”

3. Security test scenarios

Card Information Handling

- Card details must be encrypted (HTTPS required)
- CVV must not be cached at any time
- Full card number must be masked on submission

Security Risks

Possible risks:

- SQL injection in payment form
- Man-in-the-middle attacks
- Exposure of API keys
- Duplicate payment submission
- Cross-site scripting (XSS)

These should never be exposed to the users:

- Full card number
- CVV
- Internal server errors
- API secrets
- Payment provider keys

4. Usability Testing

verify:

- The checkout flow should be intuitive? Can the user easily go through the payment flow without hiccups?
- The payment steps should be minimal for ease
- Can users edit cart without restarting

- Can users switch payment methods easily without having to restart
- Are instructions clear for bank transfers despite the banks

ShopEz Payment Gateway testcases

TC ID	Test Case	Steps	Expected Result
TC01	Successful card payment	Enter valid card details	The Payment is successful and a confirmation is sent back.
TC02	Insufficient funds	Use card with insufficient funds	The Payment fails with clear message. "You have insufficient funds please try again"
TC03	Expired card	Enter expired card date	User is informed that the card is not valid
TC04	Cancel payment	Cancel during gateway redirect	The user is able to return to checkout to restart checkout process
TC05	Network interruption	Disconnect during payment	The card or wallet is charged once. No duplicate charge
TC06	Tax calculation	Apply tax rate	Correct tax added to the total amount and the user is debited the right amount.
TC07	Discount applied	Use valid discount code	Discount deducted correctly from the total price and user is debited only the right amount.
TC08	Retry payment	Fail once, retry to pay	User is able to try pay again and payment is processed successfully.
TC09	Switch payment method	Change from card to wallet during checkout	User is able to pick a different form of payment which checks out before payment
TC10	Invalid CVV	Enter the wrong CVV	Payment is declined from wallet.

Possible risks or bugs

1. Sensitive card details exposed in logs
2. Order created but payment not confirmed due to delay in the gateway processing payments.
3. Poor error messaging causing user confusion when processing payment on check out . that is why errors should be clearly laid out.