

# Test Cases Report - 65 Test Cases (Separated Steps & Data)

Total Test Cases	65
Categories	Boundary, Functional, Performance, Security, Workflow Automation, Error Handling
Priority Levels	High, Medium, Critical, Low
Generated On	2025-08-03 15:31:27

## Test Case 1: Successful User Registration with Mandatory Fields

Test ID	TC_001
Module	User Registration > Customer Onboarding
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	5 minutes
Description	Verifies that a new user can register successfully when all mandatory fields are provided with valid data.
Objective	To ensure the registration process works with valid inputs and creates a new user account.
Preconditions	User is not already registered; Registration page is accessible
Expected Result	User account is created, confirmation email is sent, and user is redirected to the welcome page with a success message.

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to registration page
Step Number	2
Action	Enter full name in the name field
Step Number	3
Action	Enter email address in the email field
Step Number	4
Action	Enter phone number in the phone field
Step Number	5

Action	Select country from dropdown
Step Number	6
Action	Set password in the password field
Step Number	7
Action	Click register button

**Test Data (Values Only):**

full_name	Ahmed Mostafa
email	ahmed.mostafa@corp.com
phone	01234567890
country	Egypt
password	StrongPassw0rd!

Validation Criteria	User account exists in database; Confirmation email received; Welcome page displays correct user details
Dependencies	Email service operational; Database connectivity
Notes	Covers basic registration flow with valid data.

**Test Case 2: Registration Fails with Invalid Email Format**

Test ID	TC_002
Module	User Registration > Form Validation
Category	Functional
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	3 minutes
Description	Ensures that the system rejects registration attempts with improperly formatted email addresses.
Objective	To validate email format enforcement during registration.
Preconditions	Registration page is accessible
Expected Result	Registration is blocked, and an error message is displayed indicating invalid email format. No account created.

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to registration page
Step Number	2

Action	Enter full name in the name field
Step Number	3
Action	Enter invalid email address in the email field
Step Number	4
Action	Enter phone number in the phone field
Step Number	5
Action	Select country from dropdown
Step Number	6
Action	Set password in the password field
Step Number	7
Action	Click register button

**Test Data (Values Only):**

full_name	Sara Nabil
email	sara.nabil@corp
phone	01122334455
country	Egypt
password	ValidPass123!

Validation Criteria	Error message is shown for invalid email; No user account is created
Dependencies	Client-side and server-side validation enabled
Notes	Tests email format validation logic.

**Test Case 3: Session Timeout After Inactivity**

Test ID	TC_003
Module	Authentication > Session Management
Category	Security
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	35 minutes
Description	Checks that user sessions expire after a defined period of inactivity.
Objective	To ensure session security and compliance with timeout policy.
Preconditions	User is registered; User is logged in

Expected Result	Session expires after 30 minutes of inactivity; user is logged out and prompted to log in again.
-----------------	--

**Test Steps (Actions Only):**

Step Number	1
Action	Login with valid credentials
Step Number	2
Action	Remain inactive on dashboard
Step Number	3
Action	Attempt to perform an action after timeout period

**Test Data (Values Only):**

username	ahmed.mostafa@corp.com
password	StrongPassw0rd!
timeout_period	30 minutes
Validation Criteria	Session expires after timeout; User is redirected to login page
Dependencies	Session management configured; Authentication service operational
Notes	Verifies compliance with session timeout policy.

**Test Case 4: Auto-Rejection of Pending Transactions After 45 Days**

Test ID	TC_004
Module	Transaction Handling > Auto-Rejection Policy
Category	Functional
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	4 minutes
Description	Ensures that transactions not approved or released within 45 days are automatically rejected.
Objective	To validate business rule for transaction auto-rejection.
Preconditions	User has a pending transaction older than 45 days
Expected Result	Transaction is marked as auto-rejected, and user is notified of the rejection.

**Test Steps (Actions Only):**

Step Number	1
Action	Login as authorized user
Step Number	2
Action	Navigate to pending transactions
Step Number	3
Action	View status of transaction older than 45 days

**Test Data (Values Only):**

username	admin@corp.com
password	AdminPass123!
transaction_id	TXN123456
transaction_age	46 days
Validation Criteria	Transaction status is 'auto-rejected'; User receives notification
Dependencies	Transaction processing service operational
Notes	Validates enforcement of auto-rejection policy.

**Test Case 5: Field Length Enforcement for Numeric Field**

Test ID	TC_005
Module	User Registration > Form Validation
Category	Boundary
Priority	Medium
Test Type	Negative
Risk Level	Medium
Estimated Time	3 minutes
Description	Tests that numeric fields enforce maximum allowed length and reject excess input.
Objective	To ensure strict field length validation for numeric fields.
Preconditions	Registration page is accessible
Expected Result	Registration is blocked, and an error message is displayed indicating phone number exceeds allowed

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to registration page
Step Number	2

Action	Enter full name in the name field
Step Number	3
Action	Enter email address in the email field
Step Number	4
Action	Enter phone number exceeding allowed length in the phone field
Step Number	5
Action	Select country from dropdown
Step Number	6
Action	Set password in the password field
Step Number	7
Action	Click register button

**Test Data (Values Only):**

full_name	Mona Khaled
email	mona.khaled@corp.com
phone	0123456789012345
country	Egypt
password	MonaPass123!

Validation Criteria	Error message for field length violation; No user account is created
Dependencies	Field length validation implemented
Notes	Covers numeric field length boundary.

**Test Case 6: Alphanumeric Field SWIFT Compliance Validation**

Test ID	TC_006
Module	User Registration > Form Validation
Category	Functional
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	4 minutes
Description	Ensures that only SWIFT-compliant characters are accepted in alphanumeric fields.
Objective	To validate character restrictions for SWIFT compliance.
Preconditions	Registration page is accessible

Expected Result	Registration is blocked, and an error message is displayed indicating invalid characters in the alphanumeric field
-----------------	--

Test Steps (Actions Only):

Step Number	1
Action	Navigate to registration page
Step Number	2
Action	Enter full name in the name field
Step Number	3
Action	Enter email address in the email field
Step Number	4
Action	Enter phone number in the phone field
Step Number	5
Action	Enter invalid characters in alphanumeric field
Step Number	6
Action	Select country from dropdown
Step Number	7
Action	Set password in the password field
Step Number	8
Action	Click register button

Test Data (Values Only):

full_name	Omar#Ali
email	omar.ali@corp.com
phone	01022334455
alphanumeric_field	Omar#Ali
country	Egypt
password	OmarPass123!
Validation Criteria	Error message for invalid characters; No user account is created
Dependencies	SWIFT compliance validation implemented
Notes	Tests SWIFT character compliance.

Test Case 7: Entitlement Assignment During Onboarding

Test ID	TC_007
Module	User Registration > Entitlement Management
Category	Integration
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	6 minutes
Description	Verifies that entitlements for payment modules are correctly assigned based on user role and country
Objective	To ensure correct entitlement mapping at registration.
Preconditions	Registration page is accessible; Entitlement rules configured
Expected Result	User is registered with correct entitlements for payment modules as per country and role; entitlement

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to registration page
Step Number	2
Action	Enter user details in all required fields
Step Number	3
Action	Select country from dropdown
Step Number	4
Action	Select user role from dropdown
Step Number	5
Action	Verify entitlement checkboxes are displayed
Step Number	6
Action	Complete registration

### Test Data (Values Only):

full_name	Hassan Youssef
email	hassan.youssef@corp.com
phone	01055667788
country	Egypt
role	Corporate Admin
Validation Criteria	Entitlements assigned as per rules; User profile displays correct entitlements
Dependencies	Entitlement mapping rules implemented



Notes	Covers entitlement logic during onboarding.
-------	---

### Test Case 8: Dropdown Default Value Based on Country

Test ID	TC_008
Module	User Registration > UI/UX Elements
Category	Usability
Priority	Medium
Test Type	Positive
Risk Level	Medium
Estimated Time	2 minutes
Description	Checks that dropdowns display correct default values based on selected country.
Objective	To ensure UI reflects country-specific default selections.
Preconditions	Registration page is accessible
Expected Result	Entitlement dropdown displays the correct default value according to Egypt-specific logic.

#### Test Steps (Actions Only):

Step Number	1
Action	Navigate to registration page
Step Number	2
Action	Select country from dropdown
Step Number	3
Action	Observe default value in entitlement dropdown

#### Test Data (Values Only):

country	Egypt
Validation Criteria	Dropdown default matches country rules
Dependencies	Country-specific UI logic implemented
Notes	Validates dynamic UI behavior.

### Test Case 9: Performance of Registration Form Submission

Test ID	TC_009
Module	User Registration > Performance
Category	Performance

Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	10 minutes
Description	Evaluates the response time of the registration form submission when multiple users register simultaneously.
Objective	To ensure registration form submission meets performance standards.
Preconditions	Registration page is accessible; Load testing tools available
Expected Result	All registration submissions are processed within 2 seconds per user, with no failures or timeouts.

**Test Steps (Actions Only):**

Step Number	1
Action	Simulate multiple users accessing registration page

Step Number	2
Action	Enter valid registration details for each user

Step Number	3
Action	Submit registration forms concurrently

Step Number	4
Action	Record response times for each submission

**Test Data (Values Only):**

user_count	100
registration_details	{ "full_name": "Test User", "email": "test.user+{n}@corp.com", "phone": "01000000000{n}", "country": "Egypt" }

Validation Criteria	All submissions <2s; No errors or timeouts
Dependencies	Performance environment available
Notes	Tests scalability and performance under load.

**Test Case 10: Error Handling for Duplicate Email Registration**

Test ID	TC_010
Module	User Registration > Error Handling
Category	Error Handling
Priority	Medium
Test Type	Negative
Risk Level	Medium

<b>Estimated Time</b>	3 minutes
<b>Description</b>	Ensures the system prevents registration with an email address that is already in use.
<b>Objective</b>	To validate error handling for duplicate email addresses.
<b>Preconditions</b>	User with given email already exists
<b>Expected Result</b>	Registration is blocked, and an error message is displayed indicating the email is already registered.

### ***Test Steps (Actions Only):***

<b>Step Number</b>	1
<b>Action</b>	Navigate to registration page
<b>Step Number</b>	2
<b>Action</b>	Enter full name in the name field
<b>Step Number</b>	3
<b>Action</b>	Enter already registered email in the email field
<b>Step Number</b>	4
<b>Action</b>	Enter phone number in the phone field
<b>Step Number</b>	5
<b>Action</b>	Select country from dropdown
<b>Step Number</b>	6
<b>Action</b>	Set password in the password field
<b>Step Number</b>	7
<b>Action</b>	Click register button

### ***Test Data (Values Only):***

<b>full_name</b>	Ali Hassan
<b>email</b>	ahmed.mostafa@corp.com
<b>phone</b>	01233445566
<b>country</b>	Egypt
<b>password</b>	AliPass123!
<b>Validation Criteria</b>	Error message for duplicate email; No new user account is created
<b>Dependencies</b>	Duplicate email check implemented
<b>Notes</b>	Ensures uniqueness of email addresses.

# Test Case 11: Successful Customer Onboarding with Module Entitlements

Test ID	TC_011
Module	Corporate Module > Customer Onboarding
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	10 minutes
Description	Validates that a new customer can be onboarded with correct entitlements for all governmental paym
Objective	Ensure onboarding process assigns requested entitlements and displays them correctly in the custom
Preconditions	Admin user is logged into the admin portal; All payment modules and sub-types are configured
Expected Result	Customer is onboarded successfully; all selected entitlements and sub-types are visible and correctly

## Test Steps (Actions Only):

Step Number	1
Action	Navigate to customer onboarding section
Step Number	2
Action	Enter customer details
Step Number	3
Action	Select payment module entitlements
Step Number	4
Action	Assign sub-type entitlements under Governmental Payments
Step Number	5
Action	Submit onboarding form
Step Number	6
Action	Open customer profile to verify entitlements

## Test Data (Values Only):

customer_name	ACME Corp
country	Egypt
GCIF	EG123456
entitlements	Governmental Payments, Tax, Customs, Universal Collection
admin_user	admin1@example.com

Validation Criteria	Entitlements match input; Profile screen displays correct modules
Dependencies	Payment modules configured; Admin portal operational
Notes	Covers default entitlement assignment logic.

## Test Case 12: Field Validation for SWIFT Compliance Characters

Test ID	TC_012
Module	Corporate Module > Data Entry
Category	Functional
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	5 minutes
Description	Ensures only SWIFT-compliant characters are accepted in free format fields.
Objective	Prevent entry of invalid characters in SWIFT-compliant fields to avoid downstream errors.
Preconditions	User is on a form with SWIFT-compliant free format field
Expected Result	Form submission is successful; entered value is accepted and saved without error.

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the relevant data entry form
Step Number	2
Action	Focus on the SWIFT-compliant free format field
Step Number	3
Action	Enter text using allowed SWIFT characters
Step Number	4
Action	Submit the form

### Test Data (Values Only):

free_format_field	/?:() .,+''
user	user2@example.com
Validation Criteria	Only allowed characters accepted; No validation errors
Dependencies	Field configured for SWIFT compliance
Notes	Test negative scenario separately for invalid characters.

# Test Case 13: Auto-Rejection of Pending Transactions after 45 Days

Test ID	TC_013
Module	Corporate Module > Transaction Handling
Category	Functional
Priority	Critical
Test Type	Boundary
Risk Level	High
Estimated Time	8 minutes
Description	Checks that transactions pending for over 45 days are automatically rejected as per business rules.
Objective	Ensure compliance with auto-rejection policy to prevent stale transactions.
Preconditions	Transaction is created and pending approval/release
Expected Result	Transaction is automatically rejected after 45 days; status is updated and user is notified.

## Test Steps (Actions Only):

Step Number	1
Action	Create a new transaction
Step Number	2
Action	Leave transaction in pending state
Step Number	3
Action	Simulate passage of 45 days
Step Number	4
Action	Check transaction status

## Test Data (Values Only):

transaction_type	Governmental Payment
amount	10000.00
creation_date	2024-05-01
current_date	2024-06-15
Validation Criteria	Status changes to 'Rejected'; Notification sent
Dependencies	Scheduler or batch job for auto-rejection enabled
Notes	Simulate date change if system time manipulation is not feasible.

## Test Case 14: Dropdown Default Selection Based on Country Entitlement

Test ID	TC_014
Module	Corporate Module > Customer Profile UI
Category	Usability
Priority	Medium
Test Type	Positive
Risk Level	Medium
Estimated Time	4 minutes
Description	Ensures that dropdowns display correct default options according to country-specific entitlement logic.
Objective	Validate dynamic UI behavior for internationalization and entitlement mapping.
Preconditions	Customer profile exists with Egypt GCIF
Expected Result	Dropdowns for entitlement options display default selections as per Egypt GCIF logic.

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to customer profile screen
Step Number	2
Action	Locate entitlement dropdowns
Step Number	3
Action	Observe default selections

### Test Data (Values Only):

customer_name	Nile Trading
GCIF	EG654321
country	Egypt
Validation Criteria	Defaults match business rules; No manual selection required
Dependencies	Country-specific logic implemented
Notes	Repeat for other countries to verify internationalization.

## Test Case 15: API Integration: Entitlement Assignment via Admin Portal

Test ID	TC_015
Module	Corporate Module > API Integration
Category	Integration
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	7 minutes
Description	Tests the integration between admin portal and backend API for entitlement assignment.
Objective	Ensure entitlements assigned in UI are correctly reflected in backend via API.
Preconditions	Admin portal and backend API are operational
Expected Result	API call is made with correct payload; backend updates entitlement and returns success response; U

**Test Steps (Actions Only):**

Step Number	1
Action	Login to admin portal

Step Number	2
Action	Navigate to entitlement management section

Step Number	3
Action	Assign new entitlement to customer

Step Number	4
Action	Submit entitlement assignment

Step Number	5
Action	Verify backend API call and response

**Test Data (Values Only):**

customer_id	CUST7890
entitlement	Universal Collection
admin_user	admin2@example.com
api_endpoint	/api/entitlements/assign

Validation Criteria	API payload correct; Entitlement updated in backend
Dependencies	API endpoint available; Admin portal connected to backend
Notes	Monitor API logs for request/response validation.



## Test Case 16: Security Validation: Unauthorized Access to Entitlement Management

Test ID	TC_016
Module	Corporate Module > Security
Category	Security
Priority	Critical
Test Type	Negative
Risk Level	Critical
Estimated Time	5 minutes
Description	Checks that users without admin privileges cannot access or modify entitlements.
Objective	Ensure access control is enforced for sensitive entitlement management actions.
Preconditions	Non-admin user is logged in
Expected Result	Access is denied; user cannot view or modify entitlements; appropriate error message is displayed.

### Test Steps (Actions Only):

Step Number	1
Action	Login as non-admin user

Step Number	2
Action	Attempt to navigate to entitlement management section

Step Number	3
Action	Try to assign or modify entitlements

### Test Data (Values Only):

user	user3@example.com
role	Standard User

Validation Criteria	No unauthorized access; Clear error messaging
Dependencies	Role-based access control implemented
Notes	Test with multiple non-admin roles for coverage.

## Test Case 17: File Upload: Beneficiary Document Processing

Test ID	TC_017
Module	Corporate Module > Beneficiary Management
Category	Functional
Priority	High

Test Type	Positive
Risk Level	High
Estimated Time	6 minutes
Description	Validates that beneficiary documents can be uploaded and processed as per requirements.
Objective	Ensure file upload and backend processing work correctly for beneficiary addition.
Preconditions	User is on beneficiary addition screen
Expected Result	Document is uploaded successfully, processed without error, and beneficiary is added with document

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to beneficiary addition section
Step Number	2
Action	Click upload document button
Step Number	3
Action	Select beneficiary document file
Step Number	4
Action	Submit beneficiary addition form
Step Number	5
Action	Verify document processing status

### Test Data (Values Only):

file_name	beneficiary_id.pdf
file_type	PDF
file_size	1.2MB
beneficiary_name	Ali Hassan

Validation Criteria	File accepted; Document linked to beneficiary
Dependencies	File upload and processing services operational
Notes	Repeat with other file types for coverage.

## Test Case 18: Error Handling: Invalid Data Type in Numeric Field

Test ID	TC_018
Module	Corporate Module > Data Entry
Category	Error Handling

Priority	Medium
Test Type	Negative
Risk Level	Medium
Estimated Time	4 minutes
Description	Ensures that non-numeric input is rejected in fields defined as numeric-only.
Objective	Validate strict data type enforcement to prevent invalid data entry.
Preconditions	User is on a form with numeric-only field
Expected Result	Form submission is blocked; validation error message indicates only numeric values are allowed.

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the relevant data entry form
Step Number	2
Action	Focus on numeric-only field
Step Number	3
Action	Enter alphabetic characters in the field
Step Number	4
Action	Attempt to submit the form

### Test Data (Values Only):

numeric_field	ABC123
Validation Criteria	Invalid input rejected; Clear error message displayed
Dependencies	Field validation implemented
Notes	Test with other invalid formats for completeness.

## Test Case 19: Performance: Bulk Entitlement Assignment via API

Test ID	TC_019
Module	Corporate Module > API Integration
Category	Performance
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	8 minutes

Description	Measures system performance and response time when assigning entitlements to a large set of customers.
Objective	Ensure system can handle bulk entitlement assignments efficiently.
Preconditions	API endpoint for bulk assignment is available
Expected Result	API processes all assignments within acceptable response time (<5 seconds); all customers receive their assigned entitlements.

**Test Steps (Actions Only):**

Step Number	1
Action	Prepare bulk entitlement assignment payload

Step Number	2
Action	Send bulk assignment request to API

Step Number	3
Action	Monitor API response time

Step Number	4
Action	Verify entitlements assigned for all customers

**Test Data (Values Only):**

customer_ids	CUST1001, CUST1002, CUST1003, CUST1004, CUST1005
entitlement	Tax
api_endpoint	/api/entitlements/bulk-assign

Validation Criteria	Response time <5s; All entitlements assigned
Dependencies	Bulk API endpoint operational
Notes	Increase batch size for stress testing.

**Test Case 20: Boundary: Maximum Field Length Enforcement**

Test ID	TC_020
Module	Corporate Module > Data Entry
Category	Boundary
Priority	Medium
Test Type	Boundary
Risk Level	Low
Estimated Time	5 minutes
Description	Verifies that the system enforces the maximum length for fixed-length alphanumeric fields.
Objective	Prevent data truncation and ensure compliance with field specifications.

Preconditions	User is on a form with fixed-length alphanumeric field
Expected Result	Field accepts up to 30 characters; additional input is blocked; form submits successfully with full value

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to the relevant data entry form
Step Number	2
Action	Focus on fixed-length alphanumeric field
Step Number	3
Action	Enter maximum allowed number of characters
Step Number	4
Action	Attempt to enter additional characters
Step Number	5
Action	Submit the form

**Test Data (Values Only):**

alphanumeric_field	A1B2C3D4E5F6G7H8I9J0K1L2M3N4O5P6Q7R8S9T0
max_length	30

Validation Criteria	Input limited to max length; No truncation or overflow
Dependencies	Field length validation implemented
Notes	Test with values just below and above the limit.

**Test Case 21: Verify auto-rejection of pending transactions after 45 days**

Test ID	TC_021
Module	Corporate Module > Transaction Handling
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	15 minutes
Description	Ensure that transactions pending approval or release for more than 45 days are automatically rejected
Objective	Validate the business rule for auto-rejection of stale transactions and ensure proper notification.

Preconditions	User has created a transaction pending approval; Transaction remains unapproved for 45 days
Expected Result	Transaction is automatically rejected after 45 days, user receives notification, and audit log records the rejection

**Test Steps (Actions Only):**

Step Number	1
Action	Login to the corporate portal as a business user
Step Number	2
Action	Navigate to the transactions section
Step Number	3
Action	Create a new transaction and submit for approval
Step Number	4
Action	Simulate passage of 45 days without approval or release
Step Number	5
Action	Check the status of the transaction
Step Number	6
Action	Review user notifications and audit logs

**Test Data (Values Only):**

user_role	Business User
transaction_type	Governmental Payment
transaction_amount	10000.00
transaction_date	2024-05-01
approval_status	Pending
days_elapsed	45

Validation Criteria	Transaction status is 'Rejected'; Notification is sent; Audit log entry exists
Dependencies	Transaction creation module; Notification service; Audit logging
Notes	Simulate date change if system time manipulation is required.

**Test Case 22: Validate mandatory field enforcement during customer onboarding**

Test ID	TC_022
Module	Corporate Module > Customer Onboarding
Category	Functional

Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	5 minutes
Description	Check that all mandatory fields are enforced and appropriate error messages are displayed when left blank.
Objective	Ensure system does not allow submission of onboarding form with missing mandatory information.
Preconditions	User is on the customer onboarding screen
Expected Result	System prevents submission, highlights all mandatory fields, and displays clear error messages for each missing field.

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the customer onboarding form
Step Number	2
Action	Leave all mandatory fields blank
Step Number	3
Action	Attempt to submit the onboarding form
Step Number	4
Action	Observe the error messages displayed

### Test Data (Values Only):

mandatory_fields	Customer Name, GCIF, Country, Email
input_values	{"Customer Name": "", "GCIF": "", "Country": "", "Email": ""}
Validation Criteria	Submission is blocked; All missing fields are highlighted; Error messages are user-friendly
Dependencies	Onboarding UI validation
Notes	Test with different browsers for UI consistency.

## Test Case 23: Check data type and SWIFT compliance validation for transaction fields

Test ID	TC_023
Module	Corporate Module > Transaction Entry
Category	Functional
Priority	High
Test Type	Negative

Risk Level	High
Estimated Time	7 minutes
Description	Ensure that fields accept only allowed data types and SWIFT-compliant characters as per specification
Objective	Prevent invalid data entry that could cause downstream processing errors.
Preconditions	User is on the transaction creation screen
Expected Result	System rejects invalid entries, highlights the fields, and displays specific error messages for each data type

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to the transaction creation form
Step Number	2
Action	Enter invalid characters in alphanumeric fields
Step Number	3
Action	Enter alphabetic characters in numeric-only fields
Step Number	4
Action	Enter non-SWIFT compliant characters in free format fields
Step Number	5
Action	Attempt to submit the transaction

**Test Data (Values Only):**

alphanumeric_field	Beneficiary Name
alphanumeric_value	John@Doe!
numeric_field	Amount
numeric_value	10A00
free_format_field	Remarks
free_format_value	Payment for #project

Validation Criteria	Fields are validated as per data type; SWIFT compliance enforced; Error messages are clear
Dependencies	Field validation logic
Notes	Test with both UI and backend validation.

**Test Case 24: Performance test for loading customer profile with multiple entitlements**

Test ID	TC_024
---------	--------



Module	Corporate Module > Customer Profile
Category	Performance
Priority	Medium
Test Type	Positive
Risk Level	Medium
Estimated Time	10 minutes
Description	Measure the time taken to load the customer profile screen when the user has multiple entitlements assigned
Objective	Ensure acceptable performance under heavy entitlement data.
Preconditions	User has multiple entitlements assigned
Expected Result	Customer profile screen loads completely within 3 seconds, with all entitlement options and UI elements visible

### Test Steps (Actions Only):

Step Number	1
Action	Login to the corporate portal as an entitled user
Step Number	2
Action	Navigate to the customer profile screen
Step Number	3
Action	Start a timer when the profile screen is requested
Step Number	4
Action	Stop the timer when the profile screen is fully loaded

### Test Data (Values Only):

user_role	Corporate Admin
entitlements	Governmental Payments, Tax, Custom, Universal Collection, Adhoc Bill
profile_id	CUST123456
expected_load_time	3 seconds
Validation Criteria	Load time <= 3 seconds; All entitlements displayed
Dependencies	Profile data service; Entitlement mapping
Notes	Repeat test during peak and off-peak hours.

## Test Case 25: Verify admin portal can add new governmental payment types post go-live

Test ID	TC_025
---------	--------

Module	Corporate Module > Entitlement Management
Category	Integration
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	8 minutes
Description	Test the extensibility of the admin portal by adding a new governmental payment type and verifying it
Objective	Ensure the system supports post go-live addition of payment types without code changes.
Preconditions	Admin user is logged in; No prior entry for the new payment type
Expected Result	New payment type is added successfully, appears in the entitlement list, and is selectable for new en

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the admin portal entitlement management section
Step Number	2
Action	Select option to add a new governmental payment type
Step Number	3
Action	Enter details for the new payment type
Step Number	4
Action	Save the new payment type
Step Number	5
Action	Verify the new payment type appears in the entitlement list

### Test Data (Values Only):

admin_user	sysadmin@example.com
new_payment_type	Environmental Levy
description	Levy for environmental compliance
country	Egypt
Validation Criteria	Payment type is saved; Visible in entitlement list; Selectable for users
Dependencies	Admin portal UI; Entitlement backend service
Notes	Test extensibility for future updates.

## Test Case 26: Boundary test for fixed length field enforcement

Test ID	TC_026
Module	Corporate Module > Data Entry
Category	Boundary
Priority	Medium
Test Type	Negative
Risk Level	Medium
Estimated Time	6 minutes
Description	Verify that fixed length fields accept only the specified number of characters and reject entries outside the range.
Objective	Prevent data truncation or overflow in fixed length fields.
Preconditions	User is on a form with fixed length fields
Expected Result	System rejects both under-length and over-length entries, displaying appropriate error messages and logging the attempt.

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to the relevant data entry form

Step Number	2
Action	Enter fewer characters than required in the fixed length field

Step Number	3
Action	Attempt to submit the form

Step Number	4
Action	Enter more characters than allowed in the fixed length field

Step Number	5
Action	Attempt to submit the form again

**Test Data (Values Only):**

field_name	GCIF
fixed_length	8
under_length_value	12345
over_length_value	123456789

Validation Criteria	Length validation enforced; Clear error messages
Dependencies	Field validation logic
Notes	Test with both numeric and alphanumeric fixed length fields.

## Test Case 27: Verify entitlement mapping updates in database after admin changes

Test ID	TC_027
Module	Corporate Module > Entitlement Management
Category	Integration
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	8 minutes
Description	Ensure that changes made to user entitlements in the admin portal are correctly reflected in the back
Objective	Validate data integrity and consistency between UI and database.
Preconditions	Admin user is logged in; User has existing entitlements
Expected Result	Database reflects the updated entitlements accurately and matches the changes made in the admin

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the admin portal entitlement management section
Step Number	2
Action	Select a user and modify their entitlements
Step Number	3
Action	Save the changes
Step Number	4
Action	Query the backend database for the user's entitlements

### Test Data (Values Only):

admin_user	admin@example.com
user_id	USR1001
entitlement_changes	Add: Tax, Remove: Custom
database_table	user_entitlements
Validation Criteria	Database values match UI changes; No orphaned or duplicate records
Dependencies	Admin portal; Database access
Notes	Requires database access for validation.

## Test Case 28: Security test: Verify access control for entitlement management

Test ID	TC_028
Module	Corporate Module > Entitlement Management
Category	Security
Priority	Critical
Test Type	Negative
Risk Level	Critical
Estimated Time	5 minutes
Description	Ensure that only authorized admin users can access and modify entitlement management features.
Objective	Prevent unauthorized access to sensitive entitlement management functions.
Preconditions	Non-admin user is logged in
Expected Result	Access is denied, entitlement management options are not visible or actionable, and unauthorized access is blocked.

### Test Steps (Actions Only):

Step Number	1
Action	Login to the portal as a non-admin user

Step Number	2
Action	Attempt to access the entitlement management section

Step Number	3
Action	Attempt to modify entitlements for a user

### Test Data (Values Only):

user_role	Standard User
target_section	Entitlement Management
target_user_id	USR2002

Validation Criteria	Access denied for non-admins; No unauthorized changes possible; Access attempts logged
Dependencies	Role-based access control
Notes	Review audit logs for unauthorized access attempts.

## Test Case 29: Usability test: Verify default selections for entitlement checkboxes and dropdowns

Test ID	TC_029
Module	Corporate Module > Customer Profile

Category	Usability
Priority	Medium
Test Type	Positive
Risk Level	Low
Estimated Time	4 minutes
Description	Check that entitlement checkboxes and dropdowns display correct default selections based on country
Objective	Ensure user experience is consistent and reduces manual input errors.
Preconditions	User is on the customer profile screen; Country and GCIF level are set
Expected Result	Checkboxes and dropdowns display the correct default selections as per country and GCIF level configuration

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the customer profile screen

Step Number	2
Action	Observe the default state of entitlement checkboxes

Step Number	3
Action	Observe the default selections in entitlement dropdowns

### Test Data (Values Only):

country	Egypt
gcif_level	Corporate
expected_checkbox_defaults	{"Governmental Payments": true, "Tax": false, "Custom": false}
expected_dropdown_defaults	{"Adhoc Bill": "No", "Next Authorizer": "None"}

Validation Criteria	Defaults match specification; No manual adjustment needed for defaults
Dependencies	UI configuration; Country-level settings
Notes	Test with different country and GCIF combinations.

## Test Case 30: Error handling: Attempt to assign duplicate entitlement to user

Test ID	TC_030
Module	Corporate Module > Entitlement Management
Category	Error Handling
Priority	Medium
Test Type	Negative

Risk Level	Medium
Estimated Time	5 minutes
Description	Test system behavior when an admin tries to assign an entitlement that the user already possesses.
Objective	Ensure system prevents duplicate entitlements and provides appropriate feedback.
Preconditions	Admin user is logged in; User already has the entitlement
Expected Result	System blocks the duplicate assignment, displays an error message, and prevents any changes to th

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to the entitlement management section in the admin portal
Step Number	2
Action	Select a user who already has a specific entitlement
Step Number	3
Action	Attempt to assign the same entitlement again
Step Number	4
Action	Observe the system response

**Test Data (Values Only):**

admin_user	admin@example.com
user_id	USR3003
entitlement	Universal Collection
Validation Criteria	Duplicate assignments are blocked; Clear error message displayed
Dependencies	Entitlement assignment logic
Notes	Check audit logs for attempted duplicate assignment.

**Test Case 31: Verify Entitlement Assignment During Customer Onboarding**

Test ID	TC_031
Module	Corporate Module > Customer Onboarding
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	Critical

Estimated Time	10 minutes
Description	Test that the system correctly assigns payment module entitlements to a new customer during onboarding
Objective	Ensure entitlement assignment logic and UI defaults work as per requirements during onboarding
Preconditions	User has access to onboarding module; Entitlement mapping is configured; GCIF level is set for Egypt
Expected Result	Customer is onboarded with correct entitlements; default selections and field visibility match country

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to customer onboarding page

Step Number	2
Action	Enter customer profile details

Step Number	3
Action	Select country from dropdown

Step Number	4
Action	Assign user roles

Step Number	5
Action	Select payment module entitlements

Step Number	6
Action	Review and submit onboarding form

**Test Data (Values Only):**

customer_name	ABC Corp
country	Egypt
roles	Initiator, Authorizer
entitlements	Governmental Payments, Tax, Custom, Universal Collection

Validation Criteria	Entitlements assigned as per configuration; UI defaults and visibility correct; Profile reflects entitlements
Dependencies	Entitlement mapping configuration; Country list availability
Notes	Ensure all entitlement checkboxes and dropdowns follow default logic for Egypt.

**Test Case 32: Validate Field Data Types and Lengths in Customer Profile**

Test ID	TC_032
Module	Corporate Module > Customer Profile



Category	Functional
Priority	High
Test Type	Boundary
Risk Level	High
Estimated Time	8 minutes
Description	Test that all profile fields enforce correct data types, lengths, and allowed characters as per specifications
Objective	Ensure strict data validation for all profile fields
Preconditions	User is logged in as admin; Profile fields are visible
Expected Result	All fields accept only valid data as per type and length; invalid data is rejected with appropriate error messages

Test Steps (Actions Only):

Step Number	1
Action	Navigate to customer profile page

Step Number	2
Action	Enter data in amount field

Step Number	3
Action	Enter data in numeric field

Step Number	4
Action	Enter data in alphanumeric field

Step Number	5
Action	Enter data in date field

Step Number	6
Action	Save profile

Test Data (Values Only):

amount	12345.67
numeric_field	987654
alphanumeric_field	ABCD1234
date	15-07-2024

Validation Criteria	Type and length restrictions enforced; Error messages for invalid input
Dependencies	Field validation rules implemented
Notes	Test with both valid and invalid data in separate runs.

# Test Case 33: Check Auto-Rejection of Pending Transactions After 45 Days

Test ID	TC_033
Module	Corporate Module > Transaction Handling
Category	Workflow Automation
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	15 minutes
Description	Test that transactions not approved or released within 45 days are automatically rejected by the system
Objective	Validate workflow automation for transaction expiry
Preconditions	User has pending transaction; System date/time manipulation is possible in test environment
Expected Result	Transaction is automatically rejected after 45 days; user receives notification and audit trail is updated

## Test Steps (Actions Only):

Step Number	1
Action	Initiate a new transaction
Step Number	2
Action	Save transaction without approval or release
Step Number	3
Action	Advance system date by 45 days
Step Number	4
Action	Check transaction status

## Test Data (Values Only):

transaction_type	Governmental Payment
amount	5000.00
initiator	user1
system_date_advance	45 days
Validation Criteria	Transaction status changes to rejected; Notification sent; Audit log updated
Dependencies	System supports date manipulation; Notification system enabled
Notes	Check both UI and backend status.

## Test Case 34: Verify SWIFT Compliance for Free Format Fields

Test ID	TC_034
Module	Corporate Module > Data Entry
Category	Functional
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	7 minutes
Description	Test that free format fields accept only SWIFT-compliant characters and reject others.
Objective	Ensure compliance with SWIFT character requirements
Preconditions	User is on data entry screen with free format field
Expected Result	Form saves successfully with SWIFT-compliant input; error message displayed for non-compliant characters

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to data entry form

Step Number	2
Action	Enter SWIFT-compliant characters in free format field

Step Number	3
Action	Save the form

Step Number	4
Action	Enter non-SWIFT characters in free format field

Step Number	5
Action	Attempt to save the form

### Test Data (Values Only):

swift_compliant	/?:().,+ '
non_swift_compliant	#\$%^&*

Validation Criteria	Only allowed characters accepted; Clear error for invalid input
Dependencies	SWIFT validation implemented
Notes	Test both positive and negative scenarios.

# Test Case 35: Test Admin Portal Entitlement Management for Sub-Types

Test ID	TC_035
Module	Corporate Module > Admin Portal
Category	Functional
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	9 minutes
Description	Test that admin portal allows entitlement management for all sub-types (Taxes, Customs, Bills) under Corporate Module > Admin Portal
Objective	Validate admin capability to manage entitlements for all relevant sub-types
Preconditions	Admin user is logged in; Governmental Payments and sub-types configured
Expected Result	Entitlement changes are saved and reflected in customer profile; audit log records the changes.

## Test Steps (Actions Only):

Step Number	1
Action	Navigate to admin portal
Step Number	2
Action	Access entitlement management section
Step Number	3
Action	Select a customer
Step Number	4
Action	Assign or revoke entitlement for Taxes
Step Number	5
Action	Assign or revoke entitlement for Customs
Step Number	6
Action	Assign or revoke entitlement for Bills
Step Number	7
Action	Save changes

## Test Data (Values Only):

customer_id	CUST1001
entitlements_to_assign	Taxes, Customs
entitlements_to_revoke	Bills

Validation Criteria	Changes reflected in profile; Audit log updated
Dependencies	Admin portal access; Entitlement mapping
Notes	Verify both assignment and revocation.

### Test Case 36: Usability Test for Customer Profile Entitlement UI

Test ID	TC_036
Module	Corporate Module > Customer Profile
Category	Usability
Priority	Medium
Test Type	Positive
Risk Level	Medium
Estimated Time	6 minutes
Description	Test the usability of entitlement options in the customer profile screen, ensuring clear labels, logical g
Objective	Ensure entitlement UI is user-friendly and intuitive
Preconditions	User is on customer profile screen; Entitlement options are visible
Expected Result	Entitlement options are clearly labeled, logically grouped, and default states match configuration; tog

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to customer profile screen
Step Number	2
Action	Review entitlement checkboxes and dropdowns
Step Number	3
Action	Toggle checkboxes
Step Number	4
Action	Select options from dropdowns
Step Number	5
Action	Check default selections

**Test Data (Values Only):**

entitlement_options	Governmental Payments, Tax, Custom, Universal Collection
dropdown_options	Adhoc Bill, Next Authorizer

Validation Criteria	Clear labels; Logical grouping; Correct defaults; Intuitive interaction
Dependencies	UI configuration; Entitlement options loaded
Notes	Gather user feedback if possible.

### Test Case 37: Boundary Test for Fixed-Length Fields

Test ID	TC_037
Module	Corporate Module > Data Entry
Category	Functional
Priority	Medium
Test Type	Boundary
Risk Level	Medium
Estimated Time	7 minutes
Description	Test that fields with fixed length enforce the restriction, accepting only data of exact length.
Objective	Verify fixed-length enforcement for data entry fields
Preconditions	User is on data entry screen; Fixed-length fields are present
Expected Result	Only exact length value is accepted; shorter or longer values are rejected with clear error messages.

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to data entry form
Step Number	2
Action	Enter data with exact required length
Step Number	3
Action	Enter data shorter than required length
Step Number	4
Action	Enter data longer than required length
Step Number	5
Action	Attempt to save form after each entry

**Test Data (Values Only):**

field_name	GCIF Code
required_length	8
exact_length_value	12345678
short_value	1234

long_value	1234567890
------------	------------

Validation Criteria	Exact length accepted; Others rejected; Clear error messages
Dependencies	Fixed-length validation implemented
Notes	Test for multiple fixed-length fields if available.

### Test Case 38: Integration Test: Entitlement Changes Reflected Across Modules

Test ID	TC_038
Module	Corporate Module > Entitlement Management
Category	Integration
Priority	High
Test Type	Positive
Risk Level	High
Estimated Time	12 minutes
Description	Test that changes to user entitlements in the admin portal are immediately reflected in the customer portal
Objective	Ensure entitlement changes propagate across modules
Preconditions	User has entitlements assigned; Admin portal and customer portal are accessible
Expected Result	Entitlement changes made in admin portal are immediately reflected in customer portal and transaction portal

#### Test Steps (Actions Only):

Step Number	1
Action	Log in to admin portal

Step Number	2
Action	Modify user entitlements

Step Number	3
Action	Log in to customer portal as affected user

Step Number	4
Action	Navigate to transaction initiation screen

Step Number	5
Action	Check available payment modules

#### Test Data (Values Only):

user_id	user123
---------	---------

entitlement_to_add	Universal Collection
entitlement_to_remove	Tax

Validation Criteria	Changes reflected in all modules; No access to removed entitlements
Dependencies	Entitlement sync between modules
Notes	Check for caching or propagation delays.

## Test Case 39: Error Handling: Attempt Transaction Without Mandatory Entitlement

Test ID	TC_039
Module	Corporate Module > Transaction Initiation
Category	Error Handling
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	5 minutes
Description	Test that the system prevents transaction initiation if the user lacks the necessary entitlement, and di
Objective	Validate error handling for missing entitlements
Preconditions	User is logged in; User lacks required entitlement
Expected Result	System blocks transaction initiation and displays error indicating missing entitlement.

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to transaction initiation screen
Step Number	2
Action	Select payment module
Step Number	3
Action	Enter transaction details
Step Number	4
Action	Attempt to submit transaction

### Test Data (Values Only):

user_id	user456
payment_module	Tax



transaction_amount	1000.00
--------------------	---------

Validation Criteria	Transaction blocked; Clear error message shown
Dependencies	Entitlement checks implemented
Notes	Test for all payment modules with missing entitlements.

## Test Case 40: Performance Test: Load Customer Profile with Multiple Entitlements

Test ID	TC_040
Module	Corporate Module > Customer Profile
Category	Performance
Priority	Medium
Test Type	Positive
Risk Level	Medium
Estimated Time	4 minutes
Description	Test the system's performance when loading a customer profile with the maximum number of entitlements
Objective	Ensure acceptable load times and UI responsiveness under heavy entitlement configuration
Preconditions	Customer profile has all entitlements assigned; System monitoring tools available
Expected Result	Customer profile loads within acceptable time (e.g., <3 seconds); UI remains responsive and all entitlements are displayed

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to customer profile page
Step Number	2
Action	Load profile with all entitlements
Step Number	3
Action	Monitor page load time and UI responsiveness

### Test Data (Values Only):

customer_id	CUST9999
entitlements	Governmental Payments, Tax, Custom, Universal Collection, Bills, Adhoc Bill

Validation Criteria	Load time within threshold; UI responsive; All entitlements displayed
Dependencies	Performance monitoring tools; Profile with maximum entitlements
Notes	Repeat test at peak system usage.

## Test Case 41: Export Customer Profile Report as CSV

Test ID	TC_041
Module	Reporting > Data Export
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	High
Estimated Time	8 minutes
Description	Validate that users can export the customer profile report in CSV format with all entitlement fields and
Objective	Ensure data export functionality works and exported data matches UI and field specifications.
Preconditions	User is logged into the admin portal; Customer profile data exists with various entitlement combinations
Expected Result	CSV file is generated and downloaded with all customer profile fields, correct entitlement statuses, and

### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the customer profile reporting section

Step Number	2
Action	Select report type

Step Number	3
Action	Choose export format

Step Number	4
Action	Click export button

Step Number	5
Action	Download the exported file

Step Number	6
Action	Open and review the exported file

### Test Data (Values Only):

report_type	Customer Profile
export_format	CSV
user_role	Admin
customer_id	GCIF123456

Validation Criteria	All fields present; Data types and formats match specification; Entitlement statuses accurate
Dependencies	Customer profile data exists; Export functionality enabled
Notes	Verify SWIFT compliance for exported characters.

### Test Case 42: Export Report with Invalid Format

Test ID	TC_042
Module	Reporting > Data Export
Category	Error Handling
Priority	High
Test Type	Negative
Risk Level	Medium
Estimated Time	4 minutes
Description	Test system's response when user selects an unsupported export format for reports.
Objective	Validate error handling for invalid export format selection.
Preconditions	User is logged into the admin portal; At least one report is available for export
Expected Result	System displays an error message indicating the selected format is not supported and prevents export

#### Test Steps (Actions Only):

Step Number	1
Action	Navigate to the reporting section
Step Number	2
Action	Select report type
Step Number	3
Action	Choose unsupported export format
Step Number	4
Action	Click export button

#### Test Data (Values Only):

report_type	Customer Entitlement
export_format	TXT
user_role	Admin
Validation Criteria	Proper error message shown; No file is generated
Dependencies	Export format validation implemented

Notes	Check for localization of error message.
-------	--

## Test Case 43: Backup Customer Entitlement Data

Test ID	TC_043
Module	Backup and Recovery > Data Backup
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	High
Estimated Time	10 minutes
Description	Verify that the system can successfully create a backup of all customer entitlement data.
Objective	Ensure backup process completes and backup file is stored as per policy.
Preconditions	User has backup privileges; Entitlement data exists
Expected Result	Backup completes successfully, file is stored at specified location, and backup log is updated.

### Test Steps (Actions Only):

Step Number	1
Action	Log in to the admin portal
Step Number	2
Action	Navigate to backup management section
Step Number	3
Action	Select data type for backup
Step Number	4
Action	Initiate backup process
Step Number	5
Action	Monitor backup progress
Step Number	6
Action	Verify backup completion

### Test Data (Values Only):

data_type	Customer Entitlement
backup_location	/secure/backups/
user_role	System Admin

Validation Criteria	Backup file exists; Log entry created; No data loss
Dependencies	Backup storage available
Notes	Check backup timestamp and file integrity.

### Test Case 44: Restore Data from Latest Backup

Test ID	TC_044
Module	Backup and Recovery > Data Recovery
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	High
Estimated Time	12 minutes
Description	Validate that entitlement data can be restored from the latest backup without data corruption.
Objective	Ensure recovery process restores all entitlement data accurately.
Preconditions	At least one backup exists; User has restore privileges
Expected Result	All entitlement data is restored accurately, no corruption or data loss, and system logs the restore event.

**Test Steps (Actions Only):**

Step Number	1
Action	Log in to the admin portal
Step Number	2
Action	Navigate to backup management section
Step Number	3
Action	Select latest backup file
Step Number	4
Action	Initiate restore process
Step Number	5
Action	Monitor restore progress
Step Number	6
Action	Verify restored data in customer profile

**Test Data (Values Only):**

backup_file	backup_20240601_120000.zip
-------------	----------------------------

user_role	System Admin
Validation Criteria	Data matches pre-backup state; No errors in logs
Dependencies	Backup file integrity; Restore functionality enabled
Notes	Test with both small and large datasets.

Test Case 45: Cross-Platform Report Export (Windows, Mac, Linux)

Test ID	TC_045
Module	Reporting > Data Export
Category	Integration
Priority	High
Test Type	Positive
Risk Level	Medium
Estimated Time	15 minutes
Description	Verify that report export functionality works consistently across Windows, Mac, and Linux operating systems.
Objective	Ensure cross-platform compatibility for report export.
Preconditions	User has access to all target platforms; Report data is available
Expected Result	Exported report downloads and opens correctly on all platforms, with no formatting or data loss issues.

Test Steps (Actions Only):

Step Number	1
Action	Log in to the portal on the target platform
Step Number	2
Action	Navigate to the reporting section
Step Number	3
Action	Select report type
Step Number	4
Action	Choose export format
Step Number	5
Action	Click export button
Step Number	6
Action	Download and open the exported file

Test Data (Values Only):

report_type	Entitlement Summary
export_format	XLSX
platforms	Windows 11, macOS Ventura, Ubuntu 22.04
user_role	Admin

Validation Criteria	File opens on all platforms; Data and formatting consistent
Dependencies	Platform-specific drivers installed
Notes	Check for encoding and line ending differences.

Test Case 46: Backup Failure Due to Insufficient Storage

Test ID	TC_046
Module	Backup and Recovery > Data Backup
Category	Error Handling
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	6 minutes
Description	Test system behavior when backup is initiated but storage is insufficient.
Objective	Validate error handling and notification for backup failures.
Preconditions	User has backup privileges; Backup storage is nearly full
Expected Result	System aborts backup, displays an error about insufficient storage, and logs the failure event.

Test Steps (Actions Only):

Step Number	1
Action	Log in to the admin portal

Step Number	2
Action	Navigate to backup management section

Step Number	3
Action	Select data type for backup

Step Number	4
Action	Initiate backup process

Test Data (Values Only):

data_type	Customer Entitlement
backup_location	/secure/backups/
available_storage	100MB
required_storage	500MB
user_role	System Admin

Validation Criteria	Error message shown; No partial backup created; Event logged
Dependencies	Storage monitoring enabled
Notes	Simulate low disk space for test.

### Test Case 47: Export Report with SWIFT Non-Compliant Characters

Test ID	TC_047
Module	Reporting > Data Export
Category	Security
Priority	Medium
Test Type	Negative
Risk Level	Medium
Estimated Time	7 minutes
Description	Check that exported reports do not contain characters outside the SWIFT-compliant set.
Objective	Ensure exported data adheres to SWIFT compliance rules.
Preconditions	User is logged in; Report contains fields with special characters
Expected Result	Exported file excludes or sanitizes non-SWIFT-compliant characters, and all fields comply with allowed

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to the reporting section

Step Number	2
Action	Select report type

Step Number	3
Action	Choose export format

Step Number	4
Action	Click export button

Step Number	5
-------------	---



Action	Download and review the exported file
--------	---------------------------------------

**Test Data (Values Only):**

report_type	Customer Profile
export_format	CSV
field_value	Omar@#Ahmed
user_role	Admin

Validation Criteria	No invalid characters in export; Sanitization or error reported
Dependencies	SWIFT compliance validation implemented
Notes	Test with multiple fields and character types.

**Test Case 48: Restore Data with Corrupted Backup File**

Test ID	TC_048
Module	Backup and Recovery > Data Recovery
Category	Error Handling
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	5 minutes
Description	Test system's ability to detect and handle corrupted backup files during restore.
Objective	Ensure corrupted backups are not restored and proper error is shown.
Preconditions	Corrupted backup file is present; User has restore privileges
Expected Result	System detects corruption, aborts restore, displays error message, and logs the incident.

**Test Steps (Actions Only):**

Step Number	1
Action	Log in to the admin portal

Step Number	2
Action	Navigate to backup management section

Step Number	3
Action	Select corrupted backup file

Step Number	4
Action	Initiate restore process

Test Data (Values Only):

backup_file	backup_20240601_120000_corrupted.zip
user_role	System Admin
Validation Criteria	Restore aborted; Error message shown; Incident logged
Dependencies	Corruption detection implemented
Notes	Check system log for detailed error.

Test Case 49: Export Large Report for Performance Benchmark

Test ID	TC_049
Module	Reporting > Data Export
Category	Performance
Priority	Medium
Test Type	Boundary
Risk Level	Medium
Estimated Time	10 minutes
Description	Assess system performance when exporting a report with a large dataset.
Objective	Ensure export completes within acceptable time and system remains responsive.
Preconditions	User is logged in; Large dataset is available for export
Expected Result	Export completes within 2 minutes, file contains all records, and no system errors or timeouts occur.

Test Steps (Actions Only):

Step Number	1
Action	Navigate to the reporting section
Step Number	2
Action	Select report type
Step Number	3
Action	Choose export format
Step Number	4
Action	Click export button
Step Number	5
Action	Measure time taken for export
Step Number	6
Action	Download and verify the exported file

Test Data (Values Only):

report_type	Full Customer Entitlement
export_format	CSV
dataset_size	100,000 records
user_role	Admin

Validation Criteria	Export time under threshold; All records present
Dependencies	Large dataset loaded
Notes	Monitor system resource usage during export.

Test Case 50: Usability: Export Button Accessibility

Test ID	TC_050
Module	Reporting > Data Export
Category	Usability
Priority	Low
Test Type	Positive
Risk Level	Low
Estimated Time	5 minutes
Description	Verify that the export button is accessible via keyboard and screen readers for users with disabilities.
Objective	Ensure export functionality meets accessibility standards.
Preconditions	User is logged in; Report is available for export
Expected Result	Export button is reachable and operable via keyboard, and screen reader announces it correctly.

Test Steps (Actions Only):

Step Number	1
Action	Navigate to the reporting section

Step Number	2
Action	Tab through page elements to reach export button

Step Number	3
Action	Activate export button using keyboard

Step Number	4
Action	Verify screen reader announces export button

Test Data (Values Only):

report_type	Customer Profile
export_format	CSV
assistive_technology	NVDA Screen Reader
user_role	Admin

Validation Criteria	Keyboard navigation works; Screen reader output correct
Dependencies	Accessibility features enabled
Notes	Test with multiple browsers if possible.

### Test Case 51: Successful Customer Onboarding with Entitlement Assignment

Test ID	TC_061
Module	Corporate Module > Customer Onboarding
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	Critical
Estimated Time	10 minutes
Description	Validate that a new customer can be onboarded with correct entitlement assignment for all government
Objective	Ensure onboarding assigns entitlements as per business rules and UI displays correct defaults for Eg
Preconditions	Admin user is logged into the admin portal; No existing customer with the provided GCIF
Expected Result	Customer is onboarded successfully; all selected entitlements are assigned and visible on the profile

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to the customer onboarding section

Step Number	2
Action	Enter customer details

Step Number	3
Action	Select country

Step Number	4
Action	Assign payment module entitlements using checkboxes

Step Number	5
Action	Review default selections for entitlement checkboxes and dropdowns

Step Number	6
Action	Submit the onboarding form

Step Number	7
Action	Verify customer profile screen displays assigned entitlements

Test Data (Values Only):

customer_name	ABC Holdings
gcif	EG123456789
country	Egypt
entitlements	Governmental Payments, Tax, Custom, Universal Collection
admin_username	admin01
admin_password	AdminPass!2024

Validation Criteria	Entitlements are assigned as per selection; UI reflects correct default values; Customer profile displays all assigned entitlements
Dependencies	Admin portal access; Entitlement configuration
Notes	Covers country-specific logic and entitlement mapping during onboarding.

Test Case 52: Field Validation for SWIFT Compliance in Beneficiary Addition

Test ID	TC_062
Module	Corporate Module > Beneficiary Management
Category	Functional
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	5 minutes
Description	Ensure that beneficiary addition enforces SWIFT-compliant characters and field lengths for all mandatory fields.
Objective	Prevent invalid data entry for beneficiary details, enforcing SWIFT and field property rules.
Preconditions	User is logged into the admin portal; Beneficiary addition feature is enabled
Expected Result	System displays validation error for address field, indicating only SWIFT-compliant characters are allowed.

Test Steps (Actions Only):

Step Number	1
Action	Navigate to the beneficiary management section

Step Number	2
-------------	---

Action	Click to add a new beneficiary
Step Number	3
Action	Enter beneficiary name
Step Number	4
Action	Enter beneficiary account number
Step Number	5
Action	Enter beneficiary address with invalid characters
Step Number	6
Action	Attempt to save the beneficiary

**Test Data (Values Only):**

beneficiary_name	John Doe
account_number	1234567890123456
address	Main St. #42!@
admin_username	admin01
admin_password	AdminPass!2024

Validation Criteria	Validation error is shown for invalid characters; Beneficiary is not added to the system
Dependencies	Beneficiary management UI; Field validation logic
Notes	Tests strict data validation for SWIFT compliance and field properties.

**Test Case 53: Auto-Rejection of Stale Transactions after 45 Days**

Test ID	TC_063
Module	Corporate Module > Transaction Processing
Category	Functional
Priority	Medium
Test Type	Boundary
Risk Level	Medium
Estimated Time	2 minutes (excluding wait period)
Description	Check that transactions not approved or released within 45 days are automatically rejected and that
Objective	Ensure business rule for auto-rejection is enforced and users are informed.
Preconditions	User is logged in with transaction initiation rights; Transaction approval workflow is active
Expected Result	Transaction is automatically rejected after 45 days of inactivity, and user receives a notification regar

Test Steps (Actions Only):

Step Number	1
Action	Initiate a new payment transaction
Step Number	2
Action	Submit transaction for approval
Step Number	3
Action	Do not take any approval or release action for 45 days
Step Number	4
Action	Check transaction status after 45 days
Step Number	5
Action	Verify user receives notification of auto-rejection

Test Data (Values Only):

initiator_username	user01
initiator_password	UserPass@2024
transaction_amount	10000.00
transaction_type	Governmental Payment
transaction_date	2024-05-01

Validation Criteria	Transaction status changes to 'Auto-Rejected' after 45 days; User receives clear notification about rejection
Dependencies	Transaction workflow; Notification system
Notes	Simulate time passage using system date manipulation or test hooks.

Test Case 54: API Security Validation for Unauthorized Entitlement Modification

Test ID	TC_064
Module	Corporate Module > Entitlement Management API
Category	Security
Priority	Critical
Test Type	Negative
Risk Level	Critical
Estimated Time	3 minutes
Description	Test that the entitlement management API rejects requests from users lacking sufficient privileges.
Objective	Ensure access control is enforced at the API level for sensitive entitlement changes.
Preconditions	API endpoint for entitlement management is accessible; Test user exists with read-only privileges

Expected Result	API request is rejected with an authorization error; no changes are made to customer entitlements.
-----------------	--

**Test Steps (Actions Only):**

Step Number	1
Action	Authenticate to the API using read-only user credentials
Step Number	2
Action	Send a request to modify customer entitlements
Step Number	3
Action	Capture API response

**Test Data (Values Only):**

api_url	https://api.example.com/entitlements/modify
customer_id	CUST1001
entitlement_changes	Add: Universal Collection
username	readonlyuser
password	Readonly@2024

Validation Criteria	API returns 403 Forbidden or equivalent error; No entitlement changes are reflected in the system
Dependencies	API gateway; Access control configuration
Notes	Covers security risk of privilege escalation via API.

**Test Case 55: Performance Test for Customer Profile Screen with Multiple Entitlements**

Test ID	TC_065
Module	Corporate Module > Customer Profile UI
Category	Performance
Priority	Medium
Test Type	Positive
Risk Level	Medium
Estimated Time	4 minutes
Description	Assess the loading time and responsiveness of the customer profile screen when displaying a customer's profile with multiple entitlements.
Objective	Verify UI performance under heavy entitlement data load.
Preconditions	Customer exists with all entitlements assigned; User is logged into the admin portal
Expected Result	Customer profile screen loads within 3 seconds, all entitlement options are visible and responsive, and no errors are displayed.



Test Steps (Actions Only):

Step Number	1
Action	Navigate to the customer management section
Step Number	2
Action	Search for the customer with maximum entitlements
Step Number	3
Action	Open the customer profile screen
Step Number	4
Action	Measure the time taken for the profile to fully load
Step Number	5
Action	Interact with entitlement checkboxes and dropdowns

Test Data (Values Only):

customer_id	CUST9999
assigned_entitlements	Governmental Payments, Tax, Custom, Universal Collection, Adhoc Bill, Next Authorizer, Verifier/Release
admin_username	admin01
admin_password	AdminPass!2024
browser	Chrome
max_acceptable_load_time	3 seconds

Validation Criteria	Profile loads within acceptable time; UI remains responsive with all entitlements displayed
Dependencies	Customer profile UI; Entitlement data
Notes	Validates UI scalability and responsiveness under heavy data load.

Test Case 56: Fallback Functional Test 1

Test ID	TC_001
Module	System > Module_1
Category	Functional
Priority	Critical
Test Type	Positive
Risk Level	High
Estimated Time	20 minutes
Description	Comprehensive functional validation test with clearly separated action steps and data values
Objective	Validate functional requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available

Expected Result	User successfully authenticates and accesses functional module, positive test workflow executes correctly
-----------------	---

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to application login page
Step Number	2
Action	Enter user credentials
Step Number	3
Action	Submit login form
Step Number	4
Action	Access functional module
Step Number	5
Action	Execute positive test workflow
Step Number	6
Action	Validate system response and capture results

**Test Data (Values Only):**

username	fallback_user_001@test.com
password	FallbackPass1@2024
login_url	https://testapp1.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Functional > Test Area
scenario_type	Positive
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Functional_1
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Critical

Validation Criteria	Functional requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved;
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

### Test Case 57: Fallback Security Test 2

Test ID	TC_002
Module	System > Module_1
Category	Security
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	21 minutes
Description	Comprehensive security validation test with clearly separated action steps and data values
Objective	Validate security requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses security module, negative test workflow executes correctly

#### Test Steps (Actions Only):

Step Number	1
Action	Navigate to application login page
Step Number	2
Action	Enter user credentials
Step Number	3
Action	Submit login form
Step Number	4
Action	Access security module
Step Number	5
Action	Execute negative test workflow
Step Number	6
Action	Validate system response and capture results

#### Test Data (Values Only):

username	fallback_user_002@test.com
----------	----------------------------

password	FallbackPass2@2024
login_url	https://testapp2.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Security > Test Area
scenario_type	Negative
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Security_2
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	High

Validation Criteria	Security requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved; Performance metrics within acceptable range
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

### Test Case 58: Fallback Performance Test 3

Test ID	TC_003
Module	System > Module_1
Category	Performance
Priority	Medium
Test Type	Boundary
Risk Level	Medium
Estimated Time	22 minutes
Description	Comprehensive performance validation test with clearly separated action steps and data values
Objective	Validate performance requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses performance module, boundary test workflow executes without errors

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to application login page
Step Number	2
Action	Enter user credentials
Step Number	3
Action	Submit login form
Step Number	4
Action	Access performance module
Step Number	5
Action	Execute boundary test workflow
Step Number	6
Action	Validate system response and capture results

Test Data (Values Only):

username	fallback_user_003@test.com
password	FallbackPass3@2024
login_url	https://testapp3.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Performance > Test Area
scenario_type	Boundary
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Performance_3
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Medium

Validation Criteria	Performance requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

### Test Case 59: Fallback Integration Test 4

Test ID	TC_004
Module	System > Module_1
Category	Integration
Priority	Low
Test Type	Error
Risk Level	Low
Estimated Time	23 minutes
Description	Comprehensive integration validation test with clearly separated action steps and data values
Objective	Validate integration requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses integration module, error test workflow executes completion

#### Test Steps (Actions Only):

Step Number	1
Action	Navigate to application login page

Step Number	2
Action	Enter user credentials

Step Number	3
Action	Submit login form

Step Number	4
Action	Access integration module

Step Number	5
Action	Execute error test workflow

Step Number	6
Action	Validate system response and capture results

#### Test Data (Values Only):

username	fallback_user_004@test.com
password	FallbackPass4@2024
login_url	https://testapp4.example.com/login
browser_type	Chrome

browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Integration > Test Area
scenario_type	Error
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Integration_4
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Low

Validation Criteria	Integration requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved;
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

### Test Case 60: Fallback Usability Test 5

Test ID	TC_005
Module	System > Module_1
Category	Usability
Priority	Critical
Test Type	Positive
Risk Level	High
Estimated Time	24 minutes
Description	Comprehensive usability validation test with clearly separated action steps and data values
Objective	Validate usability requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses usability module, positive test workflow executes completely

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to application login page
Step Number	2

Action	Enter user credentials
Step Number	3
Action	Submit login form
Step Number	4
Action	Access usability module
Step Number	5
Action	Execute positive test workflow
Step Number	6
Action	Validate system response and capture results

**Test Data (Values Only):**

username	fallback_user_005@test.com
password	FallbackPass5@2024
login_url	https://testapp5.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Usability > Test Area
scenario_type	Positive
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Usability_5
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Critical

Validation Criteria	Usability requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved; P
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

**Test Case 61: Fallback Functional Test 6**



Test ID	TC_006
Module	System > Module_1
Category	Functional
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	25 minutes
Description	Comprehensive functional validation test with clearly separated action steps and data values
Objective	Validate functional requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses functional module, negative test workflow executes correctly

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to application login page

Step Number	2
Action	Enter user credentials

Step Number	3
Action	Submit login form

Step Number	4
Action	Access functional module

Step Number	5
Action	Execute negative test workflow

Step Number	6
Action	Validate system response and capture results

**Test Data (Values Only):**

username	fallback_user_006@test.com
password	FallbackPass6@2024
login_url	https://testapp6.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes

module_path	Main > Functional > Test Area
scenario_type	Negative
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Functional_6
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	High

Validation Criteria	Functional requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved;
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

Test Case 62: Fallback Security Test 7

Test ID	TC_007
Module	System > Module_1
Category	Security
Priority	Medium
Test Type	Boundary
Risk Level	Medium
Estimated Time	26 minutes
Description	Comprehensive security validation test with clearly separated action steps and data values
Objective	Validate security requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data
Expected Result	User successfully authenticates and accesses security module, boundary test workflow executes cor

Test Steps (Actions Only):

Step Number	1
Action	Navigate to application login page
Step Number	2
Action	Enter user credentials
Step Number	3
Action	Submit login form

Step Number	4
Action	Access security module

Step Number	5
Action	Execute boundary test workflow

Step Number	6
Action	Validate system response and capture results

Test Data (Values Only):

username	fallback_user_007@test.com
password	FallbackPass7@2024
login_url	https://testapp7.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Security > Test Area
scenario_type	Boundary
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Security_7
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Medium

Validation Criteria	Security requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved; P
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

Test Case 63: Fallback Performance Test 8

Test ID	TC_008
Module	System > Module_1
Category	Performance
Priority	Low

Test Type	Error
Risk Level	Low
Estimated Time	27 minutes
Description	Comprehensive performance validation test with clearly separated action steps and data values
Objective	Validate performance requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses performance module, error test workflow executes correctly

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to application login page

Step Number	2
Action	Enter user credentials

Step Number	3
Action	Submit login form

Step Number	4
Action	Access performance module

Step Number	5
Action	Execute error test workflow

Step Number	6
Action	Validate system response and capture results

**Test Data (Values Only):**

username	fallback_user_008@test.com
password	FallbackPass8@2024
login_url	https://testapp8.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Performance > Test Area
scenario_type	Error
expected_response_time	< 5 seconds
validation_timeout	60 seconds

evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Performance_8
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Low

Validation Criteria	Performance requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

### Test Case 64: Fallback Integration Test 9

Test ID	TC_009
Module	System > Module_1
Category	Integration
Priority	Critical
Test Type	Positive
Risk Level	High
Estimated Time	28 minutes
Description	Comprehensive integration validation test with clearly separated action steps and data values
Objective	Validate integration requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses integration module, positive test workflow executes correctly

#### Test Steps (Actions Only):

Step Number	1
Action	Navigate to application login page
Step Number	2
Action	Enter user credentials
Step Number	3
Action	Submit login form
Step Number	4
Action	Access integration module
Step Number	5

Action	Execute positive test workflow
Step Number	6
Action	Validate system response and capture results

Test Data (Values Only):

username	fallback_user_009@test.com
password	FallbackPass9@2024
login_url	https://testapp9.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Integration > Test Area
scenario_type	Positive
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Integration_9
execution_mode	Automated
cleanup_required	True
retry_count	3
priority_level	Critical

Validation Criteria	Integration requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved;
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases

Test Case 65: Fallback Usability Test 10

Test ID	TC_010
Module	System > Module_1
Category	Usability
Priority	High
Test Type	Negative
Risk Level	High
Estimated Time	29 minutes

Description	Comprehensive usability validation test with clearly separated action steps and data values
Objective	Validate usability requirements through structured workflow with separated concerns
Preconditions	Test environment is deployed and accessible; Test user credentials are configured; Required test data is available
Expected Result	User successfully authenticates and accesses usability module, negative test workflow executes correctly

**Test Steps (Actions Only):**

Step Number	1
Action	Navigate to application login page
Step Number	2
Action	Enter user credentials
Step Number	3
Action	Submit login form
Step Number	4
Action	Access usability module
Step Number	5
Action	Execute negative test workflow
Step Number	6
Action	Validate system response and capture results

**Test Data (Values Only):**

username	fallback_user_010@test.com
password	FallbackPass10@2024
login_url	https://testapp10.example.com/login
browser_type	Chrome
browser_version	118.0
screen_resolution	1920x1080
environment	Test Environment
session_timeout	30 minutes
module_path	Main > Usability > Test Area
scenario_type	Negative
expected_response_time	< 5 seconds
validation_timeout	60 seconds
evidence_types	Screenshot, Log file, Response data
test_dataset	FallbackData_Usability_10
execution_mode	Automated

cleanup_required	True
retry_count	3
priority_level	High

Validation Criteria	Usability requirements fully satisfied; System maintains stability throughout execution; Data integrity preserved; P
Dependencies	System deployment; Test data configuration; User account setup
Notes	Fallback test case with separated steps and data due to parsing error: Insufficient parsed cases