REVIEWING TEST PLANS & CREATING TDD/BDD TEST CASES

Core Digital Transformation for Retail Banking — ABC Bank Digital Core Modernization Platform

Date: Feb 18, 2026 | **Version: 1.0** | **Prepared By: Solutions Architect** | **Status: Completed**

Task 1: Review and Annotate the Sample Test Plan

The test plan for ABC Bank's Digital Core Modernization Platform was reviewed against the modules defined in scope: Customer Onboarding, Account Creation, Loan Eligibility, Transaction Processing, Fraud Detection, and AML Monitoring. The following issues were identified:

#	Issue Found	Severity	Remediation
1	**No test case for multi-factor authentication (MFA) failure** The test plan references MFA under Security Testing (Section 5.1) but contains no explicit test case for an MFA failure scenario — e.g., incorrect OTP, expired OTP, or MFA bypass attempt. This is a critical security gap.	**Critical**	Immediate
2	**Missing negative test for an invalid KYC document** KYC validation is in scope (Section 2.1) but no negative test case exists for submitting an expired, blurry, or mismatched identity document. The system's rejection logic and error-response codes are untested.	**Critical**	Immediate
3	**Duplicate test cases for account balance retrieval** Two test scenarios for account balance retrieval appear to be functionally identical (same preconditions, same inputs, same expected output) with only different test IDs, indicating a documentation error rather than intentional parameterized testing.	**High**	Sprint 1
4	**No test cases validating fraud scoring thresholds** Fraud detection is listed as in-scope (Section 2.1) but the plan contains no test cases for boundary values on fraud scoring — e.g., transactions at exactly the threshold, just below, or just above. Fraud rules engine boundary conditions are entirely unverified.	**High**	Sprint 1
5	**Acceptance criteria unclear for loan eligibility tests** The loan eligibility test scenarios do not specify quantified acceptance criteria. Terms such as 'valid income' and 'acceptable credit score' are undefined — no numeric thresholds are given, making test results subjective and non-repeatable.	**High**	Sprint 1
6	**Missing integration test: Kafka event bus message validation** The test plan lists Kafka as part of the test environment (Section 6) but contains no integration test cases validating that events are correctly published and consumed between microservices (e.g., onboarding → account creation trigger).	**Medium**	Sprint 2
7	**No AML flag escalation test case** AML monitoring is listed in scope but no test verifies the escalation path when an AML flag is raised — specifically whether the transaction is correctly suspended, a compliance alert is generated, and the audit log is updated.	**Medium**	Sprint 2

8	**Test schedule lacks specific dates and milestone owners** Section 9 (Test Schedule) lists phases without dates, durations, or assigned owners. Without concrete timelines, it is impossible to track test progress, identify blockers, or ensure regulatory deadlines are met.	**Low**	Sprint 2

Task 2: Requirements Traceability Matrix (RTM)

The RTM below maps each functional requirement to its corresponding test cases, confirming coverage status and highlighting gaps identified during the test plan review.

Req. ID	Requirement Description	Test Case IDs	Coverage	Notes
RQ-01	Customer can register and log in using valid credentials with MFA	TC-101, TC-102, TC-103	**Yes**	*TC-101: valid login; TC-102: MFA success; TC-103: MFA failure — gap identified (MFA failure case was missing, now added)*
RQ-02	KYC documents must be validated against government database and AML sanctions list	TC-201, TC-202, TC-203	**Yes**	*TC-201: valid docs; TC-202: expired document rejection; TC-203: AML flag trigger — negative case TC-202 was missing, now added*
RQ-03	Loan eligibility engine must evaluate income threshold ≥ ₹25,000/month and credit score ≥ 650	TC-301, TC-302, TC-303	**Yes**	*TC-301: eligible applicant; TC-302: income below threshold; TC-303: credit score boundary — acceptance criteria thresholds now explicitly defined*
RQ-04	Fraud detection engine must flag transactions exceeding ₹1,00,000 within 10 minutes as high-risk	TC-401	**No**	*COVERAGE GAP — No boundary-value test cases exist for the ₹1,00,000 fraud threshold. TC-401 tests only well-above threshold; TC-402 (at threshold) and TC-403 (just below) required*
RQ-05	Transaction events must be published to Kafka and consumed by the Ledger microservice within 500ms	TC-501	**No**	*COVERAGE GAP — TC-501 tests only happy-path event publishing; no consumer validation, timeout, or retry test cases exist*
RQ-06	AML-flagged transactions must be suspended and an alert raised to the Compliance Officer within 60 seconds	TC-601, TC-602	**Yes**	*TC-601: AML flag detection; TC-602: alert escalation and audit log entry — previously missing escalation path now covered*
RQ-07	Account balance retrieval must return correct balance for authenticated user within 200ms	TC-701	**Yes**	*Duplicate test case removed; single parameterized test retained. Performance SLA (200ms) added as explicit acceptance criterion*

Task 3: TDD-Style Test Cases

Three TDD-style test cases are written below for the Loan Eligibility Engine, KYC Validation Service, and Fraud Detection Rules Engine — covering positive, negative, and boundary-value scenarios.

TDD Test Case 1 — Loan Eligibility Engine

Test Case ID	TC-301
Module	Loan Eligibility Engine
Purpose	Validate that an applicant with a monthly income of exactly ₹25,000 (minimum threshold) and a credit score of exactly 650 (minimum threshold) is correctly classified as 'Eligible' — boundary-value positive test.
Preconditions	Loan Eligibility Engine microservice is running. Credit Bureau API stub returns credit score 650 for test applicant ID 'APP-BV-001'. Income verification service stub returns ₹25,000/month.
Inputs	Applicant ID: APP-BV-001 Monthly Income: ₹25,000 Credit Score: 650 Loan Amount Requested: ₹2,00,000 Loan Tenure: 24 months
Execution Steps	1. POST /api/loans/eligibility with payload {applicantId: 'APP-BV-001', income: 25000, creditScore: 650, amount: 200000, tenure: 24} 2. Assert HTTP 200 OK response. 3. Assert response body contains { eligible: true, reason: 'Meets minimum income and credit score thresholds' }. 4. Assert response time < 300ms.
Expected Output	HTTP 200 OK { eligible: true, loanRef: "LN-XXXX", reason: "Meets minimum income and credit score thresholds", emi: ₹9,167/month }
Edge Cases	Income = ₹24,999 → Expected: Not Eligible (below threshold) Credit Score = 649 → Expected: Not Eligible Income = ₹25,000 + Credit Score = 649 → Expected: Not Eligible (both criteria must pass) Null applicant ID → Expected: HTTP 400 Bad Request

TDD Test Case 2 — KYC Validation Service

Test Case ID	TC-202
Module	KYC Validation Service
Purpose	Validate that an expired identity document is correctly rejected by the KYC validation service with an appropriate error code and message — critical negative test case identified as missing in the test plan review.
Preconditions	KYC Validation Service is running. Document store contains a test passport image (test-kyc-expired.pdf) with an expiry date of 2023-01-01. Government database stub is active.
Inputs	Customer ID: CUST-NEG-007 Document Type: Passport Document File: test-kyc-expired.pdf (expired 2023-01-01) Submission Date: 2026-02-18

Execution Steps	1. POST /api/kyc/validate with multipart form: {customerId: 'CUST-NEG-007', docType: 'passport', file: test-kyc-expired.pdf}. 2. Assert HTTP 422 Unprocessable Entity response. 3. Assert response body: { valid: false, errorCode: 'KYC-003', reason: 'Document expired', expiredOn: '2023-01-01' }. 4. Assert no account creation event is published to Kafka. 5. Assert audit log entry is created with status REJECTED and reason KYC-003.
Expected Output	HTTP 422 { valid: false, errorCode: 'KYC-003', reason: 'Document expired', expiredOn: '2023-01-01', nextStep: 'Please upload a valid, non-expired document' }
Edge Cases	Blurry / illegible document → Expected: HTTP 422, errorCode KYC-004 (Image Quality) Document name mismatch vs. application name → Expected: HTTP 422, errorCode KYC-005 Document from unsupported country → Expected: HTTP 422, errorCode KYC-006 Valid document, AML watchlist hit → Expected: HTTP 200 valid but flagged, route to compliance review

TDD Test Case 3 — Fraud Detection Rules Engine

Test Case ID	TC-402
Module	Fraud Detection Rules Engine
Purpose	Validate that a transaction of exactly ₹1,00,000 within a 10-minute window is correctly classified as 'High Risk' and the transaction is suspended — boundary-value test at the fraud threshold identified as missing in the test plan review.
Preconditions	Fraud Detection Rules Engine is running. Account ACC-FR-099 has transaction history with ₹0 spent in the prior 10-minute window. Fraud threshold is configured as ₹1,00,000 / 10 minutes.
Inputs	Account ID: ACC-FR-099 Transaction Amount: ₹1,00,000 (exactly at threshold) Transaction Type: Online Transfer Time Window: Single transaction within 10 minutes Recipient: External account
Execution Steps	1. POST /api/transactions/initiate with payload {accountId: 'ACC-FR-099', amount: 100000, type: 'transfer', recipient: 'EXT-ACC-001'}. 2. Assert HTTP 202 Accepted with status: PENDING_FRAUD_REVIEW. 3. Assert fraud engine publishes FRAUD_ALERT event to Kafka topic fraud-alerts. 4. Assert transaction status in DB is SUSPENDED. 5. Assert compliance officer alert generated within 60 seconds. 6. Assert audit log entry: { event: FRAUD_FLAG, accountId: ACC-FR-099, amount: 100000, riskLevel: HIGH }.
Expected Output	HTTP 202 { status: 'SUSPENDED', riskLevel: 'HIGH', fraudRef: 'FR-XXXX', message: 'Transaction suspended pending fraud review', estimatedReviewTime: '2 hours' }
Edge Cases	Amount = ₹99,999 → Expected: Approved (below threshold, no fraud flag) Amount = ₹1,00,001 → Expected: SUSPENDED (above threshold) Two transactions of ₹60,000 within 10 min → Expected: Second triggers fraud flag (cumulative) Fraud engine timeout → Expected: Fail-safe SUSPEND, not approve

Task 4: BDD Scenarios in Gherkin Syntax

Two BDD scenarios are written below in Gherkin syntax, reflecting real customer journeys aligned with the retail banking platform's core modules.

BDD Scenario 1 — Successful Customer Onboarding with Valid KYC

Feature: Customer Onboarding & KYC Verification

```
Given  a new customer 'Jane Doe' has registered on the ABC Bank digital portal
with email jane.doe@email.com and selected a Savings account at Mumbai Branch
  And  the AML screening service confirms Jane is not on any sanctions
watchlist
When  Jane uploads a valid, unexpired passport (ID: P1234567) and a utility
bill as proof of address, and submits the onboarding form
Then  the KYC validation service verifies the documents against the government
database within 30 seconds, and the system creates account number ACC-20260218-
001, sends a confirmation email and SMS to Jane, and displays 'Account Created
Successfully' on the portal
```

BDD Scenario 2 — Loan Rejection Due to Insufficient Income

Feature: Loan Eligibility Engine

```
Given  an existing ABC Bank customer 'Raj Kumar' (Customer ID: CUST-00456) is
logged into the digital portal and has a verified KYC status
When  Raj applies for a Personal Loan of ₹5,00,000 and the loan eligibility
engine retrieves his monthly income of ₹18,000 and credit score of 620 from the
Credit Bureau API
Then  the loan eligibility engine returns 'Not Eligible' because the monthly
income ₹18,000 is below the minimum threshold of ₹25,000, the portal displays
an error message 'Loan application unsuccessful — income does not meet minimum
eligibility criteria', and no loan record is created in the database
```

BDD Scenario 3 (Bonus) — Fraud Alert on High-Value Transaction

Feature: Fraud Detection & AML Monitoring

```
Given  customer 'Priya Singh' (Account: ACC-20260101-099) has a zero
transaction history in the past 10 minutes and initiates an online transfer of
₹1,00,000 to an external account
When  the fraud detection rules engine evaluates the transaction against the
high-risk threshold of ₹1,00,000 per 10-minute window
Then  the transaction is suspended with status 'PENDING_FRAUD_REVIEW', a fraud
alert is published to the Kafka topic 'fraud-alerts', the compliance officer
receives an automated notification within 60 seconds, and an audit log entry is
created with riskLevel: HIGH
```

Task 5: Validation Checklist

Validation Item	Yes / No	Evidence / Notes
All requirements mapped in RTM	Yes	*7 requirements mapped across RQ-01 to RQ-07 with explicit gap notes for RQ-04 & RQ-05*
Edge cases considered	Yes	*Boundary values tested in TC-301 (income threshold), TC-402 (fraud threshold), TC-202 (expired docs)*
Negative scenarios included	Yes	*TC-202 (invalid KYC), TC-302 (income below threshold), TC-103 (MFA failure) all included*
TDD tests aligned with the architecture	Yes	*All three TDD cases map to microservices defined in the test plan: Loan Engine, KYC Service, Fraud Engine*
BDD flows reflect user journeys	Yes	*BDD Scenarios 1–3 cover: onboarding, loan rejection, and fraud detection — real customer-facing journeys*
Regulatory (KYC/AML) tests included	Yes	*TC-202 (KYC doc rejection), TC-601 (AML flag), TC-602 (AML escalation & audit log), BDD Scenario 1 (AML screening)*
Full traceability achieved	No	*RQ-04 (fraud boundary) and RQ-05 (Kafka consumer) have coverage gaps flagged — additional test cases TC-402, TC-403, TC-501 must be created*