ADX Credit

07.09.2025

Johnpaul Emmanuel (“Consultant”)
ISEDA Solutions LTD (“Client”)

Table of Contents

1. **Executive Summary & Background**
    - Overview
    - Problem Background
    - Project Scope
    - Stakeholders & Roles
    - Goals & Success Metrics
    - Assumptions & Constraints
    - High-Level Business Rules
    - Feasibility Summary
    - Expected Value
    - Risks & Mitigations
    - Phase-1 Deliverables
2. **System Models & Rules (Phase 2)**
    - Use Case Diagram (Roles & Interactions)
    - Use Case Descriptions (Table)

- Rules Matrix (Initial Draft)
- Sequence Diagram: Auto-Approval Flow
- Sequence Diagram: Auditor Override Flow
- Phase-2 Deliverables

3. **Functional Requirements (Phase 3)**
   - Onboarding & Registration
   - KYC & Identity Verification
   - Credit Scoring & Assessment
   - Decision Engine (Automated Loan Decisioning)
   - Loan Disbursement
   - Audit & Overrides
   - Notifications & Communication
   - Admin & Operations Portal
   - Audit Logging & Compliance
   - System Management (Back-office)
   - Phase-3 Deliverables
4. **Non-Functional Requirements (Phase 4)**
   - Performance & Scalability
   - Availability & Reliability
   - Security

- Compliance & Legal
- Usability & Accessibility
- Maintainability & Extensibility
- Monitoring & Logging
- Phase-4 Deliverables

5. **UML Models (Phase 5)**
    - Class Diagram (Core Entities & Relationships)
    - Activity Diagram (Loan Approval Workflow)
    - State Diagram (Loan/Application Lifecycle)
    - Sequence Diagrams (referenced from Phase 2)
    - Developer Notes & Conventions
    - Phase-5 Deliverables
6. **Implementation Roadmap & Release Plan (Phase 6)**
    - Implementation Phases (Requirements → Go-Live)
    - Release Milestones
    - Fallback & Contingency Plans
    - Maintenance & Future Enhancements
    - Phase-6 Deliverables

Executive Summary

The ADX Credit System is a **loan web application** for microfinance banks, designed to fully automate the loan lifecycle — from **KYC/ID validation** (including NIN, BVN, and other IDs) through **credit scoring**, **automated decision-making**, and **loan disbursement** — with a secure **manual fallback** when required.

The system introduces three key roles: **Applicant, Admin, and Auditor/Compliance**, ensuring both operational efficiency and regulatory compliance. Automated rules handle most decisions, while auditors retain the authority to override with a full audit trail.

Key objectives include:

- **Speed & Efficiency**: Reduce approval times to near real-time.
- **Fraud Reduction**: Minimize duplicate or fraudulent applications.
- **Compliance**: Maintain NDPR, KYC/AML alignment with full audit logs.
- **Scalability & Resilience**: Vendor-agnostic adapters and fallback flows for continuity.

This document outlines the **requirements, models, rules, and roadmap** across six phases. By following this phased approach, ADX Credit ensures faster loan processing, lower operational costs, improved customer experience, and stronger compliance posture.

Phase 1 — Requirements & Background

Overview

We're building a **Loan Web Application** for a microfinance bank that automates the full loan lifecycle: **KYC/ID validation (NIN, BVN, others) → credit scoring via credit bureau APIs → automated decisioning using business rules → disbursement** (via payment/banking APIs) with a **manual fallback**. Three roles interact with the system:

Applicant, **Admin**, and **Auditor/Compliance** (the auditor can override automated decisions with a full audit trail). The outcome is a faster, compliant, auditable pipeline that reduces fraud and operational cost while improving user experience.

Problem Background

- Manual reviews are slow, error-prone, and expensive.
- Fragmented checks (ID, credit, income, history) lead to inconsistent outcomes.
- Compliance pressure (KYC, AML, NDPR/data protection) demands traceability and least-privilege access.
- Customers expect real-time decisions and instant disbursement; microfinance competition is rising.

Project Scope

In Scope (MVP+)

- Applicant onboarding: account creation, profile, consent capture, and document upload.
- KYC/ID validation: primarily **NIN (via NIMC)** and **BVN (via bank/CBN channels)**; extensible to Driver's License, Voter's Card, Passport.
- Credit assessment: integrate a **pluggable credit-bureau adapter** (vendor list to follow).
- Automated decision engine: rule-based approvals/rejections with explainable outputs.
- Disbursement: automated via bank/payment APIs and manual fallback with maker–checker.

- Audit & overrides: Auditor role can override denials with justification; all actions are logged.
- Notifications: status updates (email/SMS/in-app).
- Compliance & security controls: consent logs, encryption, RBAC, data retention policy, audit trails.

Out of Scope (MVP)

- Native mobile apps (web-only initially).
- Collections/repayment engine (only minimal hooks/placeholders).
- Advanced ML underwriting (start with rules; ML optional in roadmap).

Stakeholders & Roles

- **Applicant (Customer):** applies, uploads data, tracks status, receives funds.
- **Admin (Bank Staff/Operations):** monitors queues, handles manual disbursement, and manages products/rules (under governance).
- **Auditor/Compliance Officer:** view-only analytics + override authority with forced reason codes; sees immutable logs.
- **System Owner (Bank/IT):** approves configuration changes, vendor onboarding, and release management.
- **Developers/Analysts:** build/maintain integrations, rules, and UI.

Goals & Success Metrics

- **Straight-Through Processing (STP):** ≥70% auto-approved or auto-rejected without manual touch in MVP; target ≥85% by Phase 3.
- **Fraud Reduction:** measurable drop in identity/duplicate fraud (baseline to be captured during UAT).
- **Compliance:** 100% actions traceable; zero critical audit findings on KYC/consent/logging.
- **Reliability:** ≥99.5% service uptime (web tier + decision engine) in MVP.

Assumptions & Constraints

- **APIs:** Credit bureau + NIMC + bank/payment providers expose stable REST/HTTP endpoints with test sandboxes.
- **Data residency / NDPR:** PII stored within approved regions; encryption at rest and in transit is mandatory.
- **Manual fallback:** Required for both decisioning (auditor override) and disbursement (ops flow) when third-party APIs degrade.
- **Vendor variability:** We must abstract providers behind a pluggable adapter to swap credit/ID vendors without codebase surgery.
- **Tech stack:** Web-only; dev may choose stack (front-end currently TS/React, but documentation remains tech-agnostic).

High-Level Business Rules (MVP)

These drive the automated decision engine. Formalize into a rules table in Phase 3.

1. **Identity must pass (hard rule):** If BVN/NIN mismatch, inactive, or unverifiable → **Reject** (auditor may override).

2. **Credit score threshold:** Approve if score ≥ configurable threshold for the product; borderline → **Refer/Review**.

3. **Income vs Loan Amount (Affordability / DTI):** Debt-to-Income must be ≤ configured cap; verify declared income via payslip/bank-statement parsing (manual upload MVP; API in roadmap).

4. **Existing Obligations:** Reject if active delinquent loans exist above threshold.

5. **Product Limits:** Enforce min/max loan amount, tenor, interest, and fees.

6. **Risk Flags:** Device anomalies, geo-mismatch, duplicate identities, blacklist hit → **Escalate or Reject**.

7. **Auditor Override:** Allowed with reason code and evidence; dual-control confirmation may be required based on risk.

Feasibility Summary

Technical

- Straightforward web app with external integrations: NIMC (NIN), BVN/bank APIs, credit bureau(s), payment/bank disbursement.

- Core complexity: decision engine + provider abstraction + non-repudiable audit logs.

- Scales horizontally at the API layer; stateless services + durable storage for PII, documents, and logs.

Economic

- Costs: vendor onboarding (per-call fees), KYC checks, infra/hosting, SMS/email.

- Benefits: reduced headcount in ops, faster throughput, lower fraud losses, better customer retention.

Operational

- Maker–checker controls for risky operations (manual disbursement, overrides).
- Runbooks for API downtime (retry/backoff, queueing, switch to manual).
- Clear SLAs with vendors; monitoring and alerting from day one.

Legal / Compliance

- KYC & AML: Capture consent, purpose, and data lineage; keep immutable logs.
- NDPR/Data Protection: Data minimization, retention schedules, breach response plan.
- Access Control: RBAC, least privilege, periodic access reviews.

Expected Value

- **Speed:** near-instant decisions; happier customers.
- **Consistency:** codified rules remove reviewer bias.
- **Compliance posture:** every action is traceable and reconstructable.
- **Agility:** vendor-agnostic adapters let us switch credit/ID providers quickly.
- **Resilience:** manual fallback keeps the bank operating during outages.

Risks & Mitigations (Early View)

- **Vendor outages / slow APIs** → circuit breakers, retries with exponential backoff, graceful degradation to manual flow.
- **False negatives/positives** → auditor override + post-decision tuning loop.
- **Data breach risk** → TLS 1.2+ in transit, AES-256 at rest, secrets management, audit logging, regular pen-tests.
- **Regulatory drift** → compliance reviews each release; configurable rules, not hard-coded thresholds.
- **Scope creep** → phase gates with signed acceptance criteria.

Phase-1 Deliverables

1. This **Phase-1 Requirements & Background** document (signed off by Product/Compliance/IT).
2. **Candidate Provider List (annex)** — e.g., CRC, First Central, CreditRegistry (credit bureaus); NIMC (NIN); BVN via banks/CBN channels; Flutterwave/Paystack/Moniepoint (disbursement options).
3. **Initial Rules Matrix (draft)** — thresholds and decision outcomes per product (to finalize in Phase 3).
4. **Glossary & Data Inventory (draft)** — PII fields, required consents, retention windows.

Phase 2 — Use Cases, Rules Matrix & Sequence Diagram

1. Use Case Diagram (Roles & Interactions)

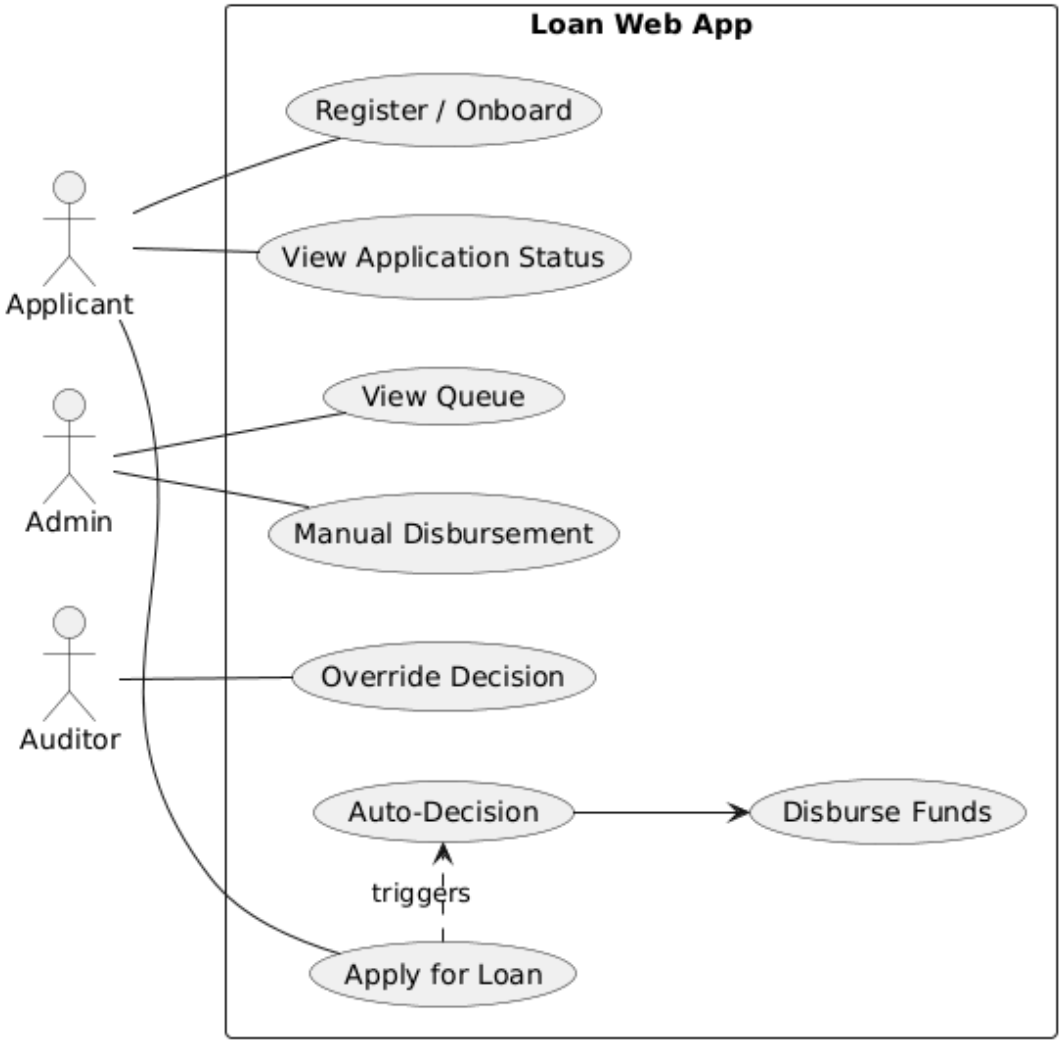
Actors:

- **Applicant** (Customer)
- **Admin** (Bank Staff/Operations)
- **Auditor/Compliance Officer**

System: **Loan Web Application**

Key Use Cases:

- Applicant: Register / Onboard, Apply for Loan, View Application Status
- Admin: View Loan Queue, Manual Disbursement
- Auditor: Override Decision
- System: Auto-Decision, Disburse Funds



Use Case Descriptions (Table)

Actor/User	***Use case***	***Description***	***Preconditions***	***Postconditions***
Applicant	*Register/ Onboard*	*Create account, upload ID, give consent*	*Applicant has valid email/phone*	*User profile created; consent captured*
Applicant	*Apply for loan*	*Submit loan amount, tenor, and documents*	*User onboarded*	*Loan application saved; enters*

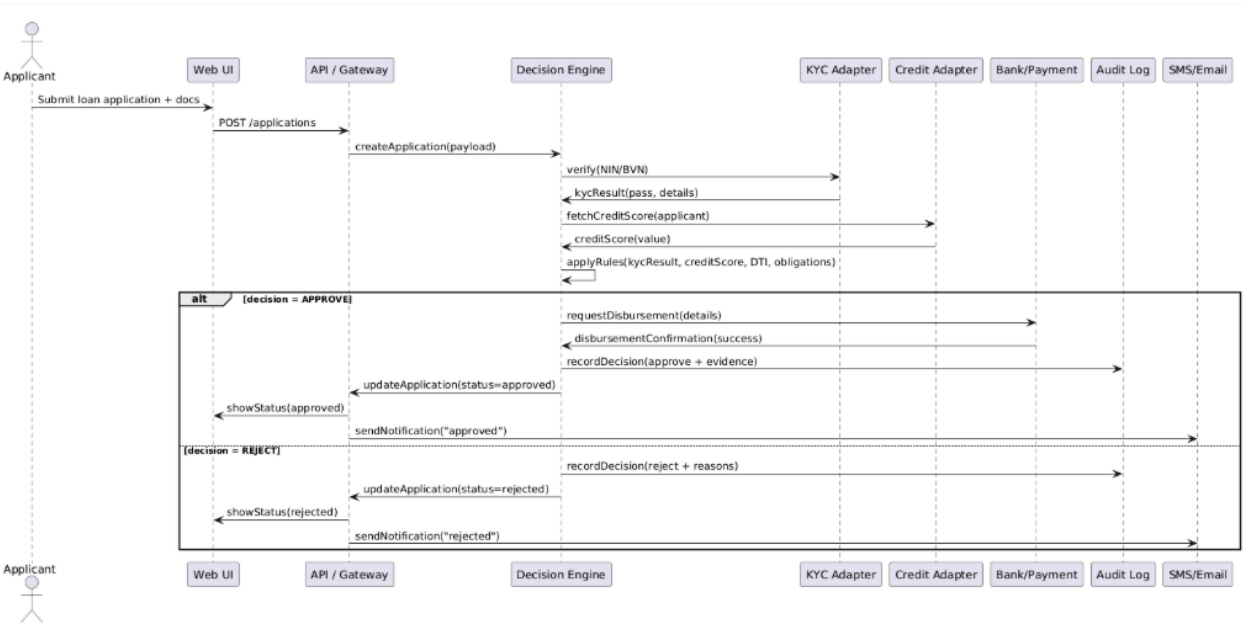
				processing queue
Applicant	*View Application status*	*View current status of loan application*	*Loan application exists*	*Status displayed; notifications logged*
Admin	*View loan Queue*	*View and manage pending/review loan applications*	*Admin authenticated*	*Applications displayed with filters/actions*
Admin	*Manual Disbursement*	*Trigger manual loan disbursement*	*Admin authenticated & permitted*	*Disbursement executed; audit log updated*
Auditor	*Override decision*	*Review and override an auto-rejected or auto-approved loan*	*Auditor authenticated; application exists*	*Decision changed; reason and evidence logged*
System	*Auto decision*	*Automatically verify ID, check credit score, and apply loan rules*	*Loan application submitted*	*Loan status = Approved / Rejected / Review; audit trail*
System	*Disburse funds*	*Send approved loan amount via bank/payment API*	*Loan status = Approved*	*Loan funds disbursed; applicant notified*

Sequence Diagram (Loan Approval Flow)

Narrative Description

This diagram shows the **happy path** of a loan application:

1. Applicant submits loan request via Web UI.
2. System sends request to API gateway and decision engine.
3. Decision engine verifies ID (NIN, BVN) and credit score through external APIs.
4. Decision engine applies business rules.
5. If approved → funds are disbursed, decision recorded, applicant notified.
6. If rejected → rejection reason recorded, applicant notified.



Notes for Stakeholder/Dev

- This is the **baseline automated flow**.
- We will add an **Auditor Override sequence diagram** in later phases to show manual override steps.

- External APIs (KYC, Credit, Bank) are **abstracted via adapters** so vendors can change without redesign.

Initial Rules Matrix (Draft)

Rule Id	***Rule name***	***Type***	***Inputs***	***Condition/threshold***
R001	*Identity Validation*	*HARD*	*NIN,BVN*	*Must match NIMC/Bank records*
R002	*Credit Score Threshold*	*SOFT*	*Credit score*	*≥600 → Approve; 500–599 → Refer; <500 → Reject*
R003	*Affordability (DTI)*	*SOFT*	*Declared income, Loan*	*Debt-to-Income ≤ 40%*
R004	*Delinquency Check*	*HARD*	*Loan history*	*Any delinquent loan above threshold*
R005	*Product Limits*	*HARD*	*Loan amount, tenor*	*Within min/max loan product parameters*
R006	*Risk Flags*	*SOFT*	*Device, GeoIP, blacklist*	*Suspicious score ≥ threshold*
R007	*Duplicate Applications*	*HARD*	*Identity fields*	*Duplicate application within 30 days*

Auditor Override Sequence Diagram

Narrative Description

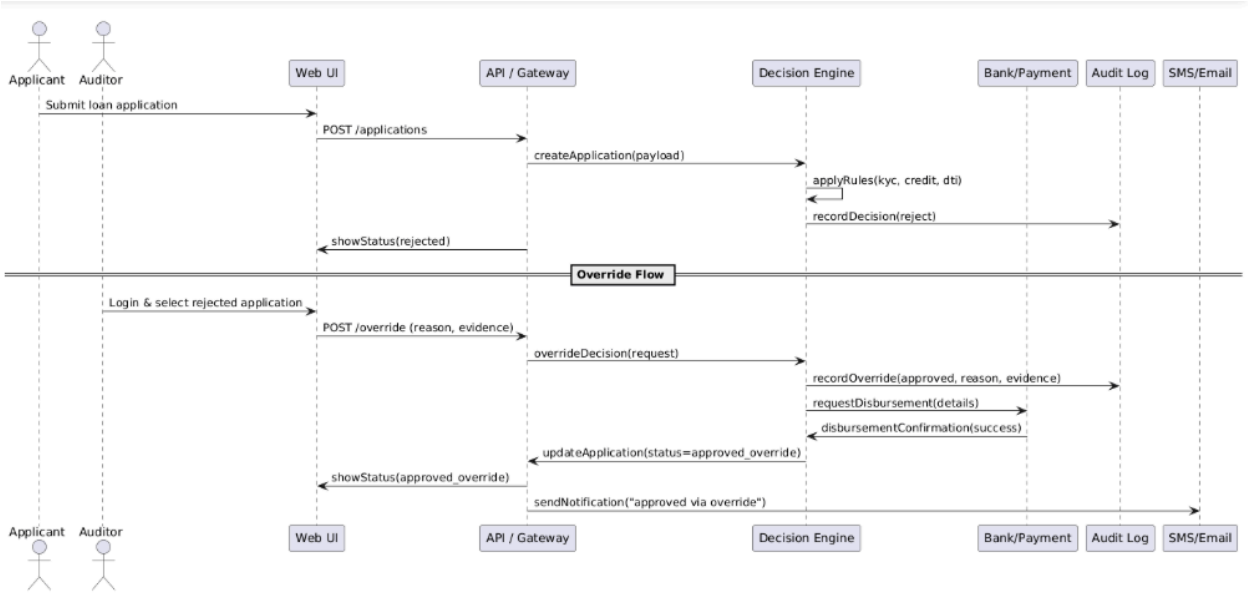
This diagram shows how an **Auditor or Compliance Officer** can override a loan decision that was auto-rejected (or flagged for review):

1. Applicant applies for a loan.
2. System processes application and rejects it automatically (due to credit score, ID mismatch, etc.).
3. Applicant physically visits or appeals, and case is escalated.
4. Auditor logs into the Loan Web App and requests override with evidence (e.g., supporting documents).
5. Decision Engine updates the application status to **Approved (Override)**.
6. Disbursement process is triggered (manual or automated).
7. Action is logged in Audit Log with **reason code + evidence**.
8. Applicant is notified of updated decision.

Notes for Stakeholder/Dev

- Overrides **must** include:
    - Reason code
    - Evidence (file/document)
    - Auditor identity logged

- Overrides are always logged as **Approved (Override)** to distinguish from auto-approval.
- Disbursement can still be **manual fallback** if APIs are down.
- Optional: Dual-authorization (second auditor must confirm override for large loans).



Phase 2 Final Deliverables (Complete)

1. **Use Case Diagram** (roles & interactions)
2. **Use Case Table** (descriptions, pre/post conditions)
3. **Rules Matrix (Initial Draft)** (loan logic)
4. **Sequence Diagram (Auto-Approval Flow)**
5. **Sequence Diagram (Auditor Override Flow)**

Phase 3 — Functional Requirements

1. Onboarding & Registration

- The system shall allow applicants to **create an account** using email/phone.
- The system shall require applicants to **set up a profile** (full name, date of birth, address, next of kin, etc.).
- The system shall require applicants to **provide consent** before processing their data (KYC, credit checks).
- The system shall allow applicants to **upload documents** (ID cards, proof of income, selfies for face match).
- The system shall validate **unique identifiers** (NIN/BVN) to prevent duplicate accounts.

---

2. KYC & Identity Verification

- The system shall integrate with **NIMC API** for NIN validation.
- The system shall integrate with **Bank/CBN API** for BVN validation.
- The system shall allow extension to validate other IDs (Driver's License, Passport, Voter's Card).
- The system shall **reject applications** if ID checks fail, with an option for auditor override.
- The system shall log all ID verification requests and results in the **Audit Log**.

---

3. Credit Scoring & Assessment

- The system shall connect to **credit bureau APIs** through a pluggable adapter (CRC, First Central, CreditRegistry).
- The system shall retrieve the **credit score and loan history** of applicants.
- The system shall calculate **Debt-to-Income (DTI)** using declared income and obligations.
- The system shall store bureau responses securely for audit purposes.
- The system shall mark applications for **auto-approve, auto-reject, or refer** based on rules.

---

4. Decision Engine (Automated Loan Decisioning)

- The system shall apply **business rules** (identity validation, credit score, DTI, product limits, risk checks).
- The system shall allow **configuration** of thresholds (credit score cutoffs, DTI percentages, product limits).
- The system shall **reject applications** failing hard rules (e.g., ID mismatch, delinquent loans).
- The system shall **approve applications** passing all required rules.
- The system shall **escalate/refer applications** that fall into borderline thresholds.
- The system shall record all outcomes in the **Decision History** with timestamps.

---

5. Loan Disbursement

- The system shall support **automated disbursement** via bank APIs or payment gateways (Flutterwave, Paystack, Moniepoint).
- The system shall support **manual disbursement** (maker–checker workflow) if APIs fail or override is required.
- The system shall update the applicant's loan status to **Disbursed** once payment confirmation is received.
- The system shall log all disbursement actions (automated or manual) in the **Audit Log**.

---

6. Audit & Overrides

- The system shall allow **Auditors** to override rejected or referred applications.
- Overrides shall require a **reason code** and **evidence upload**.
- The system shall clearly mark overridden applications as **Approved (Override)**.
- The system shall allow **dual authorization** for overrides above a configurable loan amount.
- The system shall log all override actions with **Auditor ID, reason, timestamp, and evidence**.

---

7. Notifications & Communication

- The system shall send **real-time notifications** (SMS, email, in-app) for each application status change.

- Notifications shall cover: application received, under review, approved, rejected, disbursed.
- The system shall provide **clear rejection reasons** to applicants.
- The system shall retry failed notifications and log delivery status.

---

8. Admin & Operations Portal

- The system shall provide **Admin users** with a dashboard to view loan queues.
- Admins shall be able to **filter applications** by status (pending, approved, rejected, disbursed).
- Admins shall be able to **initiate manual disbursements** with approval workflow.
- Admins shall view **transaction logs and reports** (daily/weekly/monthly).
- Admins shall not be able to override decisions (only Auditors).

---

9. Audit Logging & Compliance

- The system shall generate an **immutable audit log** of all actions (auto-decisions, overrides, disbursements).
- The audit log shall capture: **actor ID, action, timestamp, affected loan ID, evidence**.
- The system shall enforce **role-based access control (RBAC)** for Applicants, Admins, and Auditors.
- The system shall allow compliance teams to **export logs** for external review.

---

10. System Management (Back-office)

- The system shall allow **configuration of business rules** without code changes.
- The system shall allow **addition/removal of credit bureau providers** through configuration.
- The system shall provide **error handling** for API failures with fallback to manual processes.
- The system shall include **reporting tools** for monitoring throughput, approval rates, and fraud flags.

Phase-3 Deliverables

1. **Functional requirements list** (this section).
2. **Module breakdown** (onboarding, KYC, credit, decisioning, disbursement, audit, notifications).
3. **Acceptance criteria draft** — stakeholder can review and approve before implementation begins.

Phase 4 — Non-Functional Requirements

1. Performance & Scalability

- The system shall handle **at least 500 concurrent users** in MVP without performance degradation.
- Average response time for loan application submission shall be **≤ 3 seconds** under normal load.

- External API calls (KYC, credit bureau, bank) shall implement **timeouts and retries** to avoid blocking.
- The system shall scale **horizontally** at the API layer (stateless services).
- Database and storage shall support **indexing and partitioning** for growth up to at least **1M loan records**.

---

2. Availability & Reliability

- The system shall maintain **≥ 99.5% uptime** during MVP.
- Scheduled maintenance shall be limited to **off-peak hours** with proper notification to users.
- System shall implement **graceful degradation** (manual fallback) if external APIs are unavailable.
- Automated monitoring and alerting shall detect and report failures within **5 minutes**.
- Critical components (decision engine, audit log, payment integration) shall be deployed with **redundancy**.

---

3. Security

- All sensitive data shall be encrypted **in transit (TLS 1.2+)** and **at rest (AES-256 or equivalent)**.
- User authentication shall require **multi-factor authentication (MFA)** for Admin and Auditor roles.
- Passwords shall be hashed using **bcrypt or Argon2** with salt.

- The system shall enforce **role-based access control (RBAC)** with least privilege.
- Audit logs shall be **immutable** and tamper-evident.
- API keys and secrets shall be managed via a **secure vault**.
- Regular **penetration testing and vulnerability scanning** shall be performed.

---

4. Compliance & Legal

- The system shall comply with **NDPR (Nigeria Data Protection Regulation)** and any applicable **CBN microfinance directives**.
- All KYC checks shall be logged with **consent records**.
- Data retention policies shall align with **regulatory requirements** (e.g., retain for 5–7 years, then purge).
- The system shall include a **data breach response plan** with escalation steps.
- Access rights shall be reviewed **quarterly** to ensure compliance.

---

5. Usability & Accessibility

- The web app shall support **desktop and mobile browsers** (responsive design).
- The user interface shall be **simple, clear, and multi-language ready** (English MVP, extensible to local languages).
- Status updates (approved/rejected/disbursed) shall be **prominently displayed**.
- Error messages shall be **clear and actionable** (not just codes).

- Accessibility standards: WCAG 2.1 compliance where possible.

---

6. Maintainability & Extensibility

- The system shall be **modular**, with adapters for KYC, credit, and payment providers.
- Business rules shall be **configurable without code changes**.
- Code shall follow **standard naming, linting, and testing conventions**.
- System shall maintain **≥ 80% unit test coverage**.
- Documentation (API, system architecture, configs) shall be kept up to date.

---

7. Monitoring & Logging

- The system shall log **all transactions** (loan apps, decisions, overrides, disbursements).
- Logs shall be stored in a **centralized logging system** with search and reporting.
- Metrics to monitor: approval rate, rejection rate, API response times, system uptime.
- Alerts shall be configured for SLA breaches (e.g., API downtime, failed disbursements).

Phase-4 Deliverables

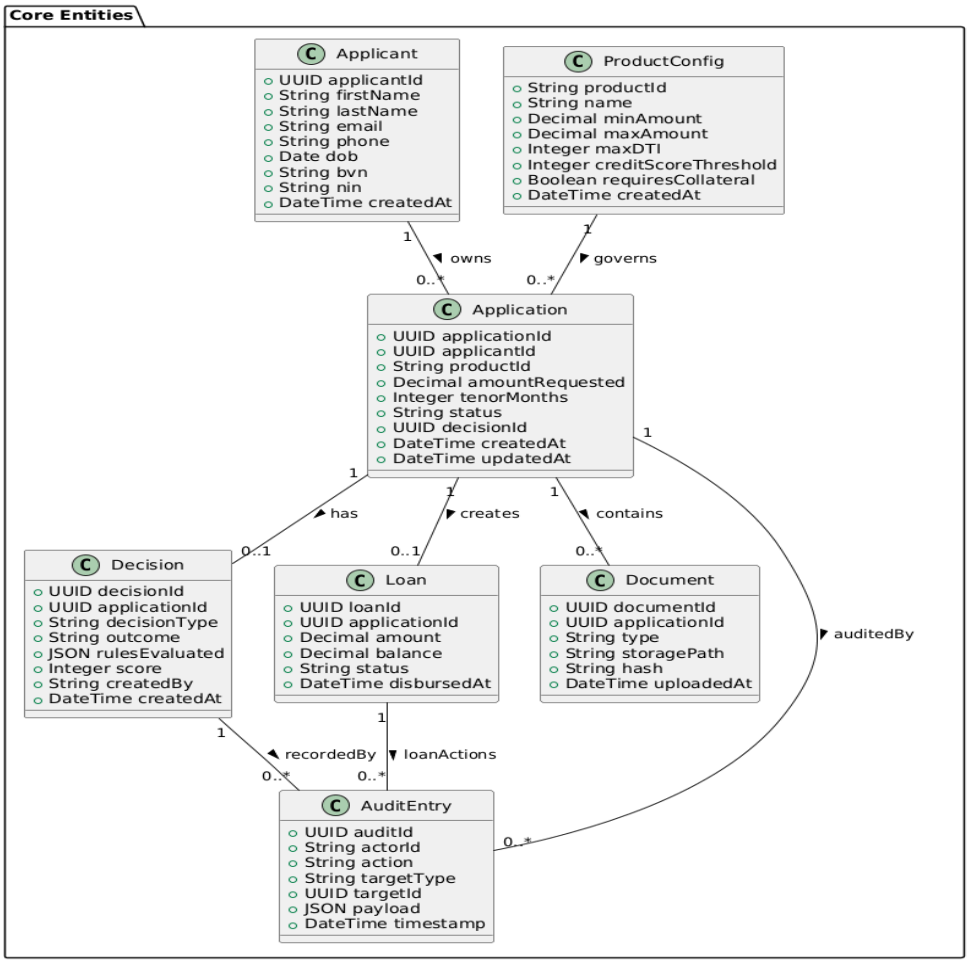
1. **Non-Functional Requirements list** (performance, security, compliance, usability, etc.).

2. **SLAs & Targets** (uptime %, response time, retention periods).

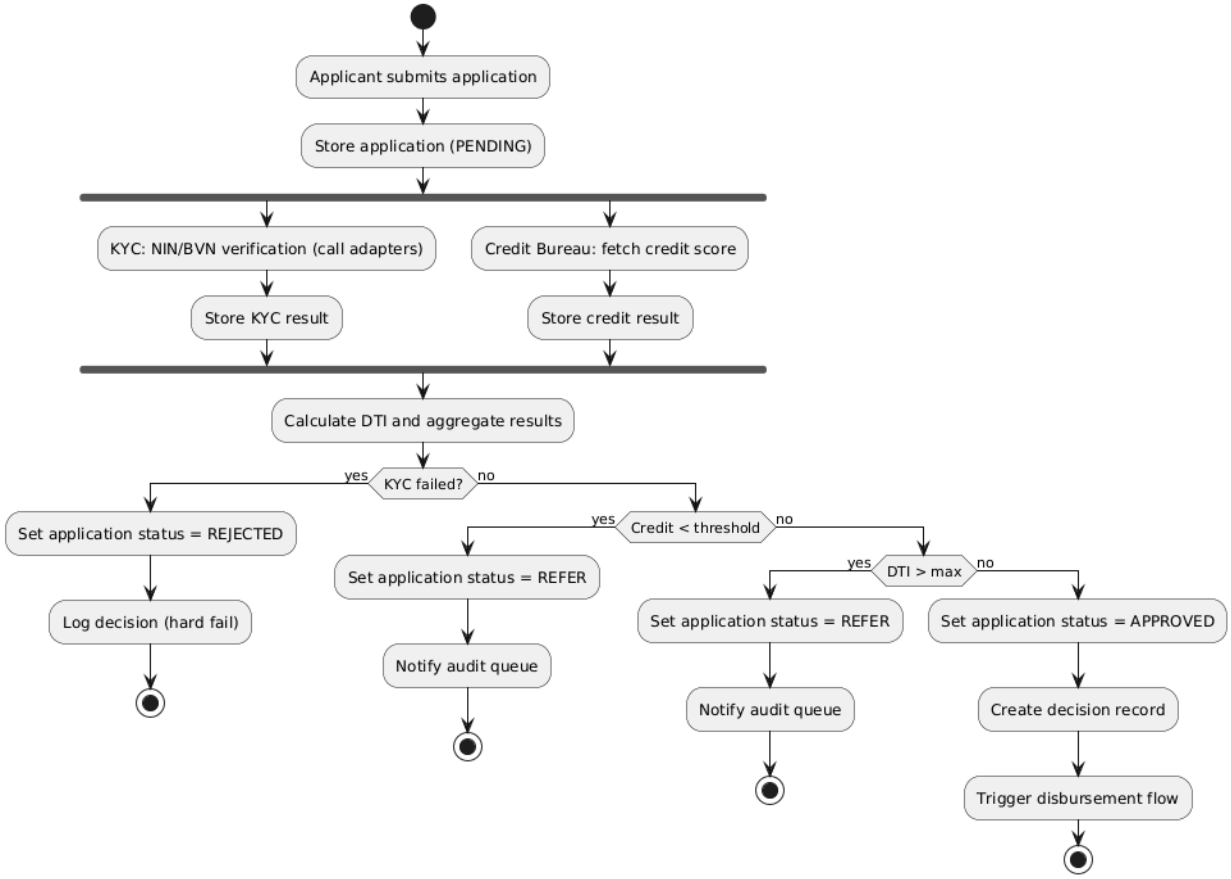
3. **Compliance & Security Checklist** (encryption, RBAC, audit log, NDPR alignment).

Phase 5 — UML Models

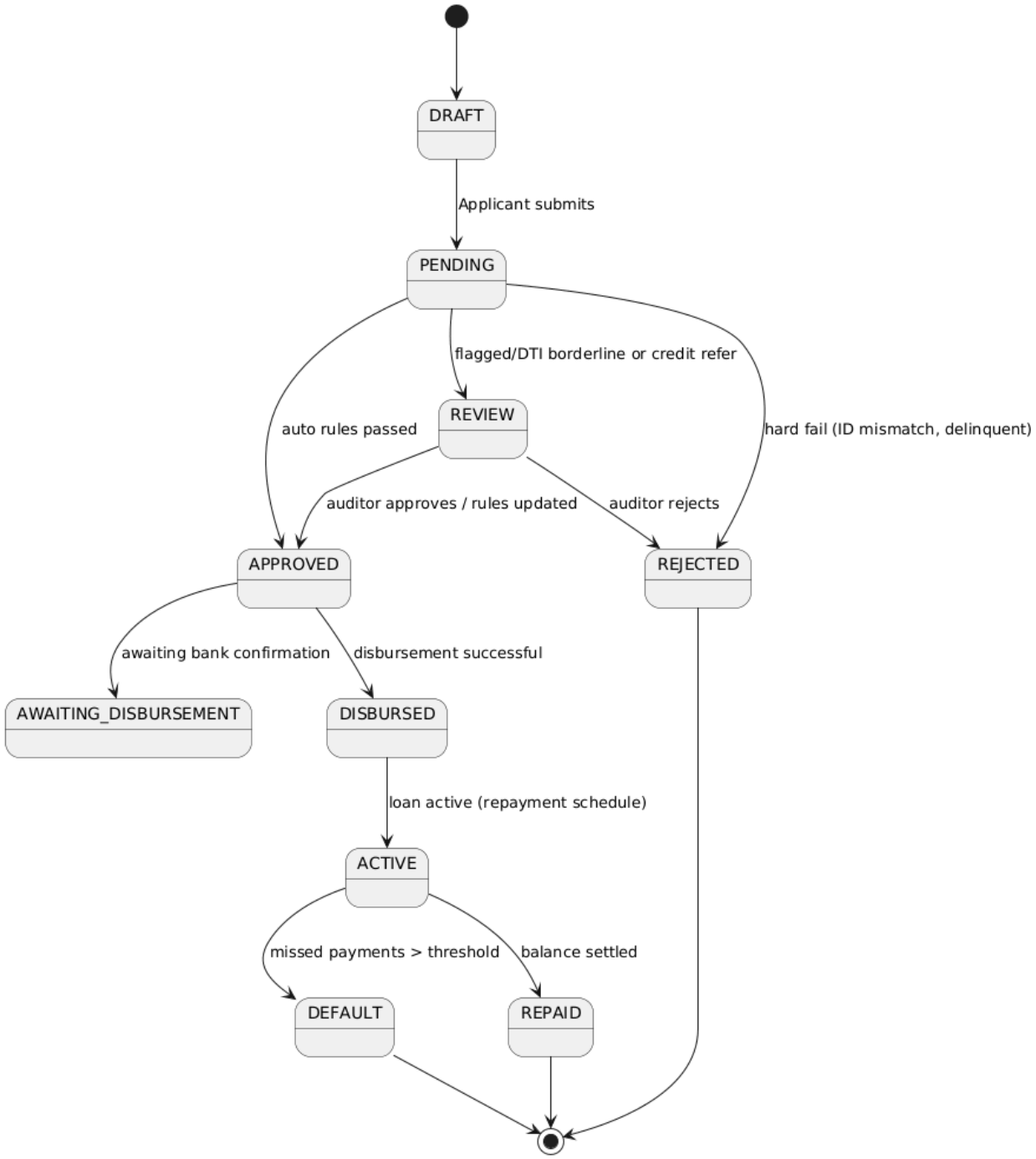
A — Class Diagram



B — Activity Diagram (Loan Approval Workflow)



C — State Diagram (Loan/Application States)


DRAFT
Applicant submits
PENDING
flagged/DTI borderline or credit refer
auto rules passed
REVIEW
hard fail (ID mismatch, delinquent)
auditor approves / rules updated
auditor rejects
APPROVED
REJECTED
awaiting bank confirmation
disbursement successful
AWAITING_DISBURSEMENT
DISBURSED
loan active (repayment schedule)
ACTIVE
missed payments > threshold
balance settled
DEFAULT
REPAID

D — Sequence Diagrams (Reminder)

- **Auto-Approval Flow (Phase 2)**
- **Auditor Override Flow (Phase 2)**

(Keep them referenced here so all UML diagrams are tracked together.)

E — Developer Notes

- **IDs: use UUIDs (no sequential IDs).**
- **PII: mask BVN/NIN in UI, encrypt at rest.**
- **Decisions: persist full JSON snapshot of rules + inputs.**
- **Audit: logs are append-only, immutable.**
- **Product Config: thresholds and limits must be configurable (not hard-coded).**
- **Idempotency: ensure API calls (createApplication, disbursement) are idempotent.**

Phase-5 Deliverables

1. **Class Diagram (core entities + relationships).**
2. **Activity Diagram (loan approval workflow).**
3. **State Diagram (application/loan lifecycle).**
4. **Sequence Diagrams (Auto-Approval + Auditor Override).**

5. Developer Notes (standards & conventions).

Phase 6 — Implementation Roadmap & Release Plan

A — Implementation Phases

Phase 1: Requirements & Design

- Finalize requirements document (this doc).
- Approve business rules and initial provider list.
- Sign off on roles & access model.
  Deliverable: Requirements Spec (signed by Product, Compliance, IT).

Phase 2: System Models & Architecture

- Produce use case diagrams, rules matrix, sequence diagrams.
- Define system architecture (web app, API gateway, adapters).
- Set up environments (dev, test, staging).
  Deliverable: Approved system design pack.

Phase 3: Core Development (MVP Scope)

- Build Applicant onboarding (account, profile, consent, document upload).
- Integrate KYC adapters (NIN via NIMC, BVN via bank API).

- **Integrate credit bureau adapter (generic, placeholder provider).**
- **Implement decision engine (rules only, no ML).**
- **Implement audit log (append-only, immutable).**
- **Enable basic notifications (email/SMS).**
  Deliverable: MVP backend + frontend; UAT with test APIs.

Phase 4: Disbursement & Admin Functions

- **Integrate bank/payment APIs for automated disbursement.**
- **Implement manual disbursement flow (maker–checker).**
- **Build Admin dashboard (loan queue, filters).**
- **Build Auditor portal (override functionality).**
  Deliverable: End-to-end loan processing including disbursement.

Phase 5: Compliance & Security Hardening

- **Enforce RBAC with MFA for Admin/Auditor.**
- **Apply data encryption (in transit + at rest).**
- **Implement data retention & purging rules.**
- **Conduct penetration testing & vulnerability scan.**
- **Audit logs export feature for compliance.**
  Deliverable: Security & compliance report.

Phase 6: Optimization & Monitoring

- Set up monitoring & alerting (uptime, API response times, failure rates).
- Add retry/circuit breaker logic for external APIs.
- Add scalability features (horizontal scaling, DB indexing).
- Improve notification reliability (delivery tracking, retry).
  Deliverable: Ops playbook + monitoring dashboard.

Phase 7: Go-Live & Post-Launch

- Deploy to production environment.
- Run smoke tests and pilot rollout with real users.
- Provide training to Admins and Auditors.
- Establish support & incident management process.
  Deliverable: Production system live + support handover.

B — Release Milestones

Milestone	Target Timeline	Key Outputs
Requirements Sign-off	Week 1–2	Approved requirements doc
MVP Build Complete	Week 6–8	Onboarding + KYC + Credit + Decision engine

Disbursement Integration	Week 10	Automated & manual disbursement flows
Compliance Hardening	Week 12	RBAC, encryption, penetration testing
Monitoring Setup	Week 13	Metrics, alerts, fallback runbooks
Production Go-Live	Week 14–16	Live deployment, support handover

---

C — Fallback & Contingency Plans

- API downtime: switch to manual KYC/credit checks, queue requests for replay.
- Disbursement failure: route to manual maker–checker until bank API recovers.
- Override disputes: require dual authorization for sensitive overrides.
- System outage: activate DR plan; recovery within SLA (≤ 4 hours).

D — Maintenance & Future Enhancements

- Regular updates to rules matrix (credit thresholds, DTI caps).
- Add new credit bureau / KYC providers via adapters.
- Roadmap for mobile app (Phase 2+ after web stabilizes).
- Explore machine learning underwriting once baseline rules stabilize.

Phase-6 Deliverables

1. **Implementation Roadmap (phases, milestones, timelines).**
2. **Release Plan (MVP → Go-Live).**
3. **Fallback & Contingency Playbook.**
4. **Maintenance & Enhancement Roadmap.**

The ADX Credit System documentation provides a comprehensive blueprint for building and deploying a loan web application tailored to microfinance operations. By breaking the project into clear phases — from requirements gathering through system modeling, functional and non-functional requirements, UML models, and an implementation roadmap — the project team can move forward with clarity, accountability, and reduced risk.

The solution delivers:

- **Automation of loan approvals through rule-based decisioning and integrations with ID/credit bureau APIs.**
- **Flexibility via manual override and fallback options when automated flows are unavailable.**
- **Compliance & Security through strict audit trails, encryption, and NDPR/KYC alignment.**
- **Scalability by using modular, vendor-agnostic adapters and performance targets from the outset.**

With the roadmap defined and deliverables mapped at each stage, ADX Credit is positioned to achieve its goals of faster loan processing, reduced fraud, improved customer trust, and operational efficiency.

This document should serve as both a reference for developers during implementation and a control artifact for stakeholders to track progress and ensure compliance at every step.