

Functional Requirements Document (FRD)

1. Document Control

- **Version:** 1.0
 - **Author:** Richard Baah, Jose Lamela, Saksham Mehta
 - **Date:** 2025-09-14
 - **Reviewers:** Development Team, Product Owner, Stakeholders
-

2. Project Overview

This project aims to build a **microlending platform** that connects borrowers seeking small loans with lenders willing to fund them. The system provides a **secure, data-driven** environment for loan application, credit evaluation, funding, repayment tracking, and compliance oversight. It solves the problem of **financial accessibility** for individuals and small businesses who lack traditional credit history, while giving lenders transparent risk insights and investment tools.

3. User Roles

Role	Description	Key Permissions
Borrower	Individual requesting loans	Create loan applications, view repayment status, communicate with lenders
Lender	Individual/institution funding loans	View borrower risk profiles, invest in loans, track repayments
Administrator	System manager	Approve/suspend users, manage transactions, generate reports
Compliance Officer	Auditor/regulator	View audit logs, export compliance reports

4. Core Functionality

Data Entry and Modification

- Borrowers can create loan applications (amount, purpose, term, income, documents).
- Lenders can pledge funds to open loan requests.
- Admins can modify or suspend user accounts.

Query and Search

- Borrowers can view application history and status.
- Lenders can search loans by borrower score, interest rate, and term.
- Admins can filter reports by user, loan status, or compliance flag.

Reporting and Visualization

- Dashboards for lenders showing portfolio performance.
- Admins and compliance officers can export audit and repayment data to CSV/PDF.
- Visual analytics (graphs/charts) for delinquency rates and repayment trends.

Authentication and Authorization

- JWT-based session tokens with user roles encoded.
- Passwords hashed with SHA-256 or bcrypt.
- Role-based access: borrowers/lenders/admins see distinct dashboards.

5. Functional Requirements

ID	Requirement	Acceptance Criteria
FR-1	Borrowers can submit loan applications.	Loan data stored with borrower ID, amount, purpose, and term.
FR-2	System performs credit/risk scoring.	Risk score computed and persisted in borrower profile.
FR-3	Lenders can manage loan portfolios.	Portfolio dashboard lists funded, active, and completed loans.
FR-4	Borrowers can make repayments.	Payment updates reflected in account and loan balance.
FR-5	System tracks delinquencies.	Late loans auto-flagged; alerts generated for admins.
FR-6	Secure communication channel between borrower and lender.	Messages stored with timestamps and encrypted.
FR-7	Admins can manage users.	Create, update, suspend users; actions logged.
FR-8	Compliance reports generated.	Reports downloadable as CSV or PDF.

6. Business Rules

- A borrower **cannot have more than 3 active unpaid loans**.
- A loan **cannot exceed \$5,000**.
- Loan requests are **automatically rejected** if the borrower’s risk score is below the system threshold.
- Loan status transitions: **PENDING → APPROVED → FUNDED → REPAID / DELINQUENT**.
- Repayments must be made **before due date** to avoid delinquency.
- Admins must **approve all new lender accounts** before first transaction.

7. Use Cases

Use Case 1: Borrower Applies for Loan

Actor: Borrower **Precondition:** Borrower account verified. **Steps:**

1. User logs in and selects "Apply for Loan."
 2. Fills out loan form with amount, purpose, and term.
 3. Submits application. **Postcondition:** Application stored in DB with status **PENDING**.
-

Use Case 2: Lender Funds a Loan

Actor: Lender **Precondition:** Loan is approved and lender has sufficient balance. **Steps:**

1. Lender views available loan list.
 2. Selects a borrower's loan and commits funds.
 3. Platform updates loan to **FUNDED**. **Postcondition:** Loan is funded and repayment schedule activated.
-

Use Case 3: Borrower Repays Loan

Actor: Borrower **Precondition:** Loan is in **FUNDED** status. **Steps:**

1. Borrower initiates repayment.
 2. Payment gateway confirms transaction.
 3. Balance updates in database; loan marked **REPAID** if complete. **Postcondition:** Repayment recorded; transaction added to audit trail.
-

Use Case 4: Admin Reviews Delinquencies

Actor: Administrator **Precondition:** Loan marked as **DELINQUENT**. **Steps:**

1. Admin views delinquency dashboard.
 2. System lists overdue loans with borrower info.
 3. Admin can contact borrower or suspend account. **Postcondition:** Delinquent accounts flagged for compliance review.
-

Use Case 5: Compliance Officer Exports Report

Actor: Compliance Officer **Precondition:** System data available. **Steps:**

1. Officer navigates to compliance tab.
 2. Selects report type (e.g., audit, KYC).
 3. Clicks "Export." **Postcondition:** PDF/CSV generated for audit purposes.
-

8. Non-Functional Requirements

Category	Requirement
Performance	Loan application and risk scoring complete within 3 seconds.
Security	JWT-based authentication, TLS encryption, password hashing, and audit logging.
Scalability	Supports horizontal scaling (Railway hosting or equivalent).
Availability	≥ 99.5% uptime.
Usability	Responsive web UI and mobile compatibility.
Data Integrity	Transactional consistency for payments and loan updates.

9. Assumptions & Dependencies

- Stripe or PayPal integration for payments.
- Cloud-hosted SQL database (e.g., PostgreSQL).
- Document verification API (e.g., Persona or Onfido) for KYC.
- SMS/email APIs for borrower-lender notifications.

10. Success Metrics

- ≥ 90% of loans processed automatically.
- ≥ 95% repayment success with automated reminders.
- 99.5% uptime and < 3s response time.
- 80% user satisfaction across borrowers and lenders.