Bank Loan Eligibility and Credit Limits Prediction Using a Machine Learning Approach

Project ID: 24-25J-268

User Acceptance Testing Report

IT21191060 - P.R.I.Pravean

IT21164644 - Pathirathna N

IT21142178 - Hilma M.I.F

B.Sc. (Hons) in Information Technology

Specializing in Information Systems Engineering

Department of Department of Computer Systems Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

APIRL 2025

Bank Loan Eligibility and Credit Limits Prediction Using a Machine Learning Approach

Project ID: 24-25J-268

IT21191060 - P.R.I.Pravean

IT21164644 - Pathirathna N

IT21142178 - Hilma M.I.F

B.Sc. (Hons) in Information Technology

Specializing in Information Systems Engineering

Department of Department of Computer Systems Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

APIRL 2025

Table of Contents

Contents

1.0 Introduction

The purpose of this document is to outline the User Acceptance Testing (UAT) process for the Loan Eligibility Prediction System Using Machine Learning and AI Chatbot. Project Sponsors from all participating departments are intending to review this document. Approval of this document implies that reviewers are confident that following the execution of the test plan, the resulting system will be considered fully tested and eligible for implementation.

UAT is to be completed by the Business Departments (UAT Team) that will be utilizing the software and/or support departments. The testing is conducted to enable a user to validate that the software meets the agreed upon acceptance criteria

This project delivers a web-based, AI-enhanced system designed to automate and improve the accuracy and efficiency of loan eligibility predictions for financial institutions, specifically targeting the banking sector in Sri Lanka. The system incorporates machine learning algorithms for predicting loan eligibility and credit limits, -based document validation and IDs, a dynamic AI chatbot for customer engagement, and a financial literacy module that provides personalized guidance for rejected applicants, including collateral-based loan alternatives.

The system is hosted on Google Cloud Platform and consists of three core modules:

- An AI chatbot for interactive support and loan-related queries,
- A machine learning model for eligibility and loan amount prediction,
- A financial guidance module for rejected applicants including asset-based suggestions.

1.0.1 In Scope

Testing of functional requirements including:

- Loan eligibility prediction workflow
- Document validation (NIC, pays lips)
- Chatbot performance for both registered and non-registered users
- Alternative financing guidance suggestions

Use case validations for:

- Existing customers, new applicants, and bank staff users
- Various rejection scenarios (e.g., low income, incomplete documentation)

User Interface (UI) consistency and user experience flow.

Testing of user registration, login, and real-time application status tracking.

1.0.2 Out of Scope

- Real-time integration with external banking systems (e.g., live CRIB or savings data APIs)
- Disaster recovery and backup plan validations
- Business continuity testing
- Advanced fraud detection or anomaly monitoring modules (planned for future enhancement)

1.1 Objective

1.1.1 Primary Objective

User Acceptance Testing is conducted to ensure that the system satisfies the needs of the business as specified in the functional requirements and provides confidence in its use. Modifications to the requirements will be captured and tested to the highest level of quality allowed within the project timeline.

.

1.1.2 Secondary Objective

To identify and expose defects and associated risks, communicate all known issues to the project team, and ensure that all issues are addressed in an appropriate manner prior to implementation. It will also assess the accuracy of AI-driven decision-making, the relevance of personalized financial suggestions, and the reliability of the chatbot across various user scenarios.

2.0 UA Test Methodology

User Acceptance Testing will be conducted primarily by the end users (i.e., Subject Matter Experts, including bank staff and business users). Users will execute all Loan Eligibility Prediction System test scripts referenced in section 6.1. Users may also perform additional tests not detailed in the plan but remain relevant and within the scope of the project. UAT progress will be tracked and reported based on the percentage of test cases executed and results from any exploratory or scenario-based testing.

Issues and defects encountered during testing will be documented by the business analysts and escalated to the development team. These incidents will be described, prioritized, and tracked using screen captures, exact user input/output, and steps necessary to reproduce the defect. Additional information regarding defect tracking and prioritization is detailed in section 7.2..

2.1 Test Phases

Phase 1: Preparation and Environment Setup

- Validate that the test environment mimics production settings, including access controls, dummy banking databases, and chatbot knowledge base.
- Load required test data including pays lips, NIC scans, and credit histories.
- Confirm user access for UAT testers (bank staff, SMEs, admin users).

Phase 2: Functional and Usability Testing
Execute core functional test cases for:

- Loan eligibility and amount prediction
- Document upload and validation
- AI chatbot interactions (registered and guest)
- Application status tracking
- Validate UI/UX flows for accessibility, responsiveness, and clarity.

Phase 3: Scenario-Based Testing

- Test diverse real-world cases:
- Approved and rejected applicants
- Re-application via collateral (land, FD, EPF)
- Error handling (e.g., incomplete uploads, invalid NIC)
- Evaluate chatbot personalization for authenticated users and general guidance for anonymous users.

Phase 4: Regression and Retesting

- Re-test all failed or blocked test cases after defect fixes.
- Validate no new issues have emerged due to code or model changes.

Phase 5: Feedback and Sign-Off

- Collect tester feedback through structured forms and debrief sessions.
- Business users sign off based on satisfactory resolution of defects and successful completion of critical test cases.

3.0 UA Test Environment

User Acceptance Testing will be conducted in a secure, cloud-hosted environment that replicates the expected production environment of the Loan Eligibility Prediction System. The application is deployed via Google Cloud Platform (GCP) and consists of three key modules: the AI chatbot interface, loan eligibility and amount prediction engine, and the financial literacy guidance module.

Testing will be conducted using a web-based interface accessible via dedicated URLs shared with UAT participants. All necessary test credentials, documentation, and environment access details will be distributed prior to testing. Applicable IP addresses, secure login credentials, and environmental URLs will be provided to the UAT Team, and all workstations will be configured to support full system functionality.

Environment configuration includes:

- Web URL (UAT): https://uat.loanpredict.ai-platform.com (example only)
- Server Environment: Google Cloud Platform (GCP) with Python (Flask) backend, MySQL database, and AI models deployed via TensorFlow and Hugging Face Transformers.
- Frontend Stack: React.js with Tailwind CSS
- Database: MySQL (test instance with anonymized synthetic financial data)

3.1 UA Test Desktops

Each test participant will be provided with a checklist to verify access to all applications within the defined
Each UAT participant will receive a pre-configured test workstation or browser access checklist to verify:

- Access to UAT environment via provided URL
- Successful login with test user credentials (New customer, Returning customer, Bank officer, Admin)
- Availability of test features such as:
    - AI chatbot interface
    - Loan application form
    - Document upload functionality
    - Application status tracker

- Correct rendering of menus and permissions based on user roles.
- Any application components missing or inaccessible must be reported to IT support for immediate resolution prior to testing.

3.2 UA Test Data

- Access to relevant test data is essential for conducting a thorough and realistic validation of the system. The following datasets and resources are required before testing begins:

- Test user accounts: Multiple profiles including salaried, self-employed, low-income, and credit-deficient applicants

- test documents: Sample NICs, pays lips, and credit reports in scanned PDF format for testing document analysis

- CRIB data simulation: Mock credit scores for approved and rejected loan applicants

- Collateral details: Asset data such as fixed deposits, EPF balance, land valuations, and vehicle registration information for collateral-based loan suggestions

- Admin and bank officer roles: Permissions to view and manage user loan histories and escalate unresolved chatbot queries

- All test users will be assigned appropriate roles that replicate production-level access, and permission requests will be coordinated through the project's UAT coordinator if additional access is required.

4.0 Roles and Responsibilities

Keys to a successful UAT process involve open channels of communication, detailed documentation, and, above all, clearly defined roles and responsibilities. Each team member must function fluidly in a group setting as well as work independently for extended periods of time. UAT is largely a collaborative process and test results must be analyzed from different perspectives and by team members with various levels of expertise across the business to ensure success.

4.1 UAT Team

The test team is composed of Subject Matter Experts (SMEs) and business users who have in-depth knowledge of banking workflows, loan processing, and customer service policies. These individuals are well-positioned to test real-world use cases, including document submissions, eligibility criteria, loan approval/rejection workflows, and chatbot interactions.
All members will be briefed on the UAT objectives, the testing process, and the tools they will use. The Business Analyst (BA) will coordinate the overall test cycle, distribute test scripts, offer guidance, and escalate critical issues to the development team.

Name	Project Role	Phone	E-mail	Location
Dilini Pathirathna	Business Analyst	076 110 75 01	dilinikanethmini@gmail.com	COLOMBO
Pravean Ravi	SME – Loan Processing	076 110 75 05	praveanravi02@gmail.com	COLOMBO
Hilma Iliyas	SME – Chatbot UX	076 110 75 07	hilmailliyas2@gmail.com	COLOMBO

4.2 Test Partners

Third-party involvement includes simulated banking interfaces and services that are essential for document validation. These services have been integrated into the UAT environment but are not managed by the internal team. The test strategy for third-party components is as follows:

- Use mock services and controlled test data to simulate responses from external engines (e.g., Tesseract) and credit score APIs.
- Validate that system behavior remains consistent across both successful and failed integrations.
- Escalate integration errors to the third-party service provider as needed for debugging or service adjustments.

The UAT team will not test production APIs or involve actual financial institutions during UAT, but a plan is in place for full integration testing during system rollout.

5.0 UAT Deliverables

The following sections detail milestones crucial to the completion of the UAT phase of the project. Once all dependent milestones have been completed, UAT will formally sign off on the system's functionality and distribute an e-mail to all project stakeholders.

5.1 UAT Activities

All core UAT activities for the **Loan Eligibility Prediction System** are defined below::

- **Identify UAT Team** – The Business Analyst compiled a list of SMEs and testers, including banking operations staff, loan officers, and chatbot interaction testers. Details are available in Section 4.1.
- **UAT Plan** – A strategy document outlining test methodology, functional areas in scope (loan eligibility prediction, chatbot, validation, collateral suggestions), and success criteria was prepared and distributed to all team members.
- **UAT Plan Team Review** – A review session was conducted with stakeholders from business, technical, and academic teams to validate the test approach, schedule, and risk mitigation plan.
- **UA Test Cases** – Detailed test cases were designed to reflect real-world banking scenarios, including:
    - Submitting applications with varying financial profiles
    - Uploading valid/invalid pays lips and NICs
    - Querying the chatbot for loan-related information

- **Test Data Acquisition** – A dataset of approximately 100 anonymized loan applications was prepared in advance. Synthetic documents were generated for pays lip testing, CRIB credit score simulation, and collateral use cases. At least one full week was allocated to prepare and load data into the UAT environment.
- **UA Test Case Review** – All scripts were reviewed and approved by SMEs and academic supervisors. Feedback from reviewers helped refine scenarios and ensure full coverage.
- **Desktop Validation** – UAT participants validated browser access to the GCP-hosted web application. All required configurations (e.g., access credentials, document upload capability) were verified.
- **UAT Environment Validation** – Each user successfully logged into the UAT environment and performed baseline tests, including chatbot login, document uploads, and accessing loan status dashboards
- **Test Case Execution** – Testers executed test scripts covering functional modules such as:
    - AI chatbot (general and personalized queries)
    - Eligibility prediction with -based inputs
    - Rejection of handling and alternative financing suggestions
    - Application progress and user notification tracking

- **Defect Tracking** – Defects were documented using a shared defect tracker (Excel spreadsheet). Each entry included:
    - Description
    - Severity (High/Medium/Low)
    - Steps to reproduce
    - Screenshots or logs
    - Assigned owner for resolution

- **UAT Touch Point** – Weekly virtual meetings were held with the test team to discuss:
    - Test progress
    - Blockers or defects
    - Schedule adjustments
    - High-priority issues requiring escalation

6.0 UA Test Cases

Test cases contain a detailed step-by-step breakdown of each scenario to be executed by the User Acceptance (UA) tester. Each script includes the following attributes:

- **Test Case Number**
- **Product**: Loan Eligibility Prediction System
- **Test Description**
- **Requirement Number** (mapped to functional specifications)
- **Requestor** (typically Business Analyst or SME)
- **Tester** (assigned SME or business user)
- **Action to be Performed**
- **Test Data to be Utilized**
- **Expected Results**
- **Error Descriptions** (if applicable)
- **Pass/Fail Result**
- **Date Tested**
- **Comments** (including any recommendations or observed issues)

The detailed UA test scripts are stored in the **UAT Test Case Spreadsheet**, with each test case linked via a hyperlink for easy access

.

Test Case ID	Product	Test Description	Tester	Action	Test Data	Expected Result	Pass/Fail
TC-001	Loan Eligibility Prediction	Validate loan approval for user with valid income and high CRIB score	Pravean Ravi	Submit application with valid NIC, pay slip, and credit score	Applicant A (Monthly Salary: LKR 100,000, CRIB Score: 720)	Application marked as eligible and approved amount displayed	Pass
TC-002	AI Chatbot	Query chatbot for general eligibility requirements (guest user)	Pravean Ravi	Ask: "What documents are needed for a loan?"	N/A	Chatbot responds with required documents	Pass
TC-003	Document Analysis	Upload low-resolution NIC scan	Hilma Iliyas	Upload blurred NIC image	NIC with 70 DPI resolution	Error message: "Unable to extract data – please upload a clearer image"	Pass
TC-004	Financial Literacy Module	Test collateral suggestion after rejection due to low salary	Dilini Pathirathna	Submit low-income application, review suggestion	Applicant B (Salary: LKR 20,000, no FD)	Rejection shown with message suggesting fixed deposit as collateral	Pass

TC-005	Application Tracking	Verify real-time status updates after submission	Hilma Iliyas	Submit application and track status	Applicant A	Status updated from "Pending Review" → "Under CRIB Check" → "Approved"	Pass

These test cases cover core system modules:

- AI chatbot responses for both registered and unregistered users
- Eligibility decisions using, CRIB score, and income validation
- Rejection management and collateral recommendations
- Application tracking and status notification flow

7.0 UAT Defects

Defects discovered during User Acceptance Testing will be logged and tracked using a shared UAT Defect Tracker spreadsheet maintained by the Business Analyst. Each defect entry will include relevant metadata such as:

- Test Case ID
- Module (e.g., Chatbot, , Eligibility Prediction)
- Description of the issue
- Steps to reproduce
- Screen captures or supporting evidence
- Severity level
- Assigned owner (developer or QA)
- Status (New, In Progress, Resolved, Retest, Closed)

These defects will be reviewed regularly to ensure timely escalation and resolution. Findings will be summarized in weekly touchpoint meetings with business and development teams.

7.1 UAT Defect Tracking

All UAT participants will be trained to execute test scripts systematically and to identify and report defects using standardized procedures. Key tools include:

- **Microsoft Excel** for logging and tracking defects in a central defect tracker.
- **SnagIt** or built-in screen capture utilities for visual documentation.
- **Descriptive reporting format** includes:
    - Date of defect occurrence
    - User role and action performed
    - Actual vs. expected result
    - Environmental conditions (e.g., browser used)

Team members are responsible for reporting issues during scheduled touchpoint meetings. Additional support may be requested by development or QA to replicate or understand complex issues.

Examples of common track issues from the project:

- AI Chatbot providing generic answers to specific customer queries (Low severity)
- engine failing to extract text from low-quality NIC scans (High severity)
- Financial guidance module not triggering after collateral-based rejection (Critical severity)

7.2 UAT Defect Prioritization

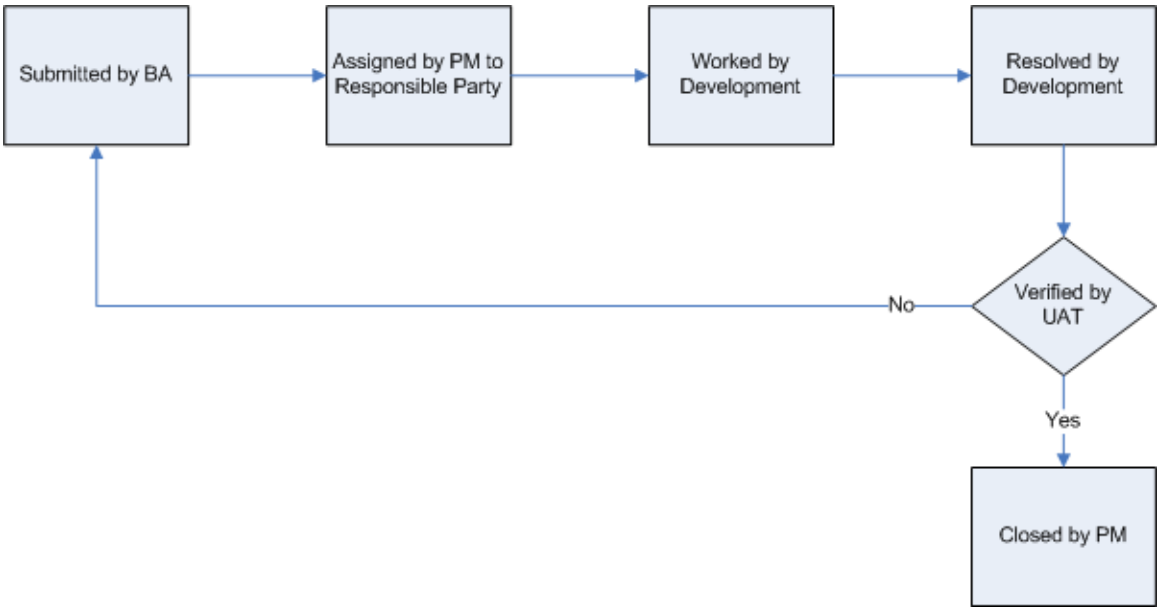
The Business Analyst (BA) will coordinate with development and stakeholders to prioritize and classify defects. The following severity levels will be used:

- **Regulatory** – Required due to compliance with government or institutional policy (not applicable in this version but noted for future integration with banking regulations).
- **Critical** – Blocking functionality, such as eligibility decisions not processing or application crashes during document upload. No workaround is possible.
- **High** – Important features, like incorrect loan amount prediction or delayed chatbot response for authenticated users. Workaround may exist but is not ideal.
- **Low** – UI inconsistencies, minor formatting issues in chatbot replies, or text misalignment that do not impact functionality. It can be resolved post-launch if needed.

7.3 UAT Defect Lifecycle

The defective lifecycle will follow the below sequence to ensure effective resolution and verification:

1. **Capture** – Defect identified and logged with full details and visual evidence.
2. **Review** – Business Analyst reviews and assigns severity and ownership.
3. **Escalation** – High/Critical defects escalated immediately to development.
4. **Resolution** – Developer fixes the issue in the UAT environment.
5. **Retesting** – UAT tester re-executes the related test case to verify fix.
6. **Closure** – If resolved successfully and meets acceptance criteria, the defect is marked as Closed.



8.0 Assumptions

The following are key assumptions made by the UAT team prior to the commencement of the acceptance test phase for the Loan Eligibility Prediction System:

- QA testing has been completed with all critical and high-severity defects resolved or mitigated prior to UAT initiation.
- The UAT environment is stable and accessible, reflecting the final production configuration, including AI chatbot, ML models, integration, and the financial literacy module.
- All necessary configuration information and synthetic test data (loan applications, pays lips, NICs, CRIB scores, and collateral records) have been loaded into the UAT environment and validated.
- Desktops and browsers used for UAT (Google Chrome, Firefox) have been pre-configured with required permissions and tools such as screen capture utilities (e.g., SnagIt) and internet access to the hosted system on Google Cloud Platform.
- All Subject Matter Experts (SMEs) from relevant areas, loan processing, banking operations, and chatbot evaluation—are available and have been trained to execute test scripts and report findings.
- The AI chatbot model's knowledge base is up to date with current loan policies, eligibility rules, and user-specific data to ensure realistic testing.
- All third-party integrations (engine, CRIB data simulator) function as expected within the UAT scope using test stubs or mocked responses.

9.0 Risks

Below are risks that could potentially impact the UAT process and prevent its successful and timely completion:

- Unstable UAT Environment

  The web-based system is hosted on Google Cloud Platform and integrates multiple services (ML models, and chatbot). Downtime or latency in these components could interrupt test execution or affect result accuracy.

- Inadequate Test Data

  The success of eligibility prediction and collateral recommendation features depends on realistic financial and credit data. If test data does not cover a wide range of scenarios (e.g., rejected applications, low CRIB scores, various income levels), system behavior may not be properly validated.

- Incorrect Software Version(s)

  Deploying an outdated or development version of the platform may result in the absence of critical features such as real-time status tracking, chatbot responses, or eligibility thresholds, leading to invalid test results.

- Failure to Emulate Production Environment

  The absence of API integrations with real-time CRIB or banking systems in UAT may limit validation of actual response behaviors, particularly for CRIB score validation and financial verification.

- Lack of Human Resources

  If SMEs or assigned testers are unavailable due to parallel academic or operational duties, test execution may be delayed, affecting the overall UAT schedule.

- AI Chatbot Limitations

  The chatbot uses retrieval-augmented generation (RAG) and may provide generic responses if prompts are ambiguous or fall outside the trained scope. Miscommunication during testing can impact user confidence in the system.

- Defect Resolution Turnaround

  Critical issues uncovered during testing may require architectural changes or retraining ML models, which may not be feasible within the UAT window, impacting timelines.

10.0 References

[1] M. A. Sheikh, A. K. Goel, and T. Kumar, "An Approach for Prediction of Loan Approval using Machine Learning Algorithm," in *IEEE International Conference on Computing*, 2020.

[2] N. Uddin, M. K. U. Ahamed, M. A. Uddin, M. M. Islam, M. A. Talukder, and S. Aryal, "An Ensemble Machine Learning Based Bank Loan Approval Prediction System with a Smart Application," *Int. J. of Cognitive Computing in Engineering*, vol. 4, pp. 327–339, Jun. 2023.

[3] Q. Lu et al., "Developing Responsible Chatbots for Financial Services: A Pattern-Oriented Responsible Artificial Intelligence Engineering Approach," *IEEE Intelligent Systems*, vol. 38, no. 6, pp. 42–51, Nov. 2023.

[4] A. Lusardi and O. S. Mitchell, "The Economic Importance of Financial Literacy: Theory and Evidence," *National Bureau of Economic Research*, Working Paper No. 18952, 2013. [Online]. Available: http://www.nber.org/papers/w18952