User Acceptance Testing (UAT) Template

Date	04/02/2026
Team ID	LTVIP2026TMIDS81483
Project Name	Prosperity Prognosticator: Machine Learning for Startup Success Prediction
Maximum Marks	4 Marks

Project Overview:

Project Name: Prosperity Prognosticator: Machine Learning for Startup Success Prediction

Project Description: Prosperity Prognosticator is a machine learning–driven web application that predicts startup success using funding, industry, and investment features. Built with a Random Forest model and deployed using Flask, it delivers real-time predictions with confidence scores through an interactive interface. The system supports data-driven investment insights and policy decisions while complementing—not replacing—human judgment.

Project Version: v4.0

Testing Period: 04/02/206 to 12/02/2026

Testing Scope:

Features and Functionalities to be Tested

1. Homepage Loading – Verify index.html loads correctly when accessing /.
2. Prediction Form Display – Ensure home.html loads properly via GET /predict.
3. Form Submission Handling – Check that user input is correctly captured and processed.
4. Feature Mapping Accuracy – Validate that input features match the required model feature order.
5. Model Prediction Execution – Confirm the Random Forest model generates predictions without errors.
6. Confidence Score Calculation – Verify success and failure probabilities are calculated correctly.
7. Success Level Classification – Ensure correct success category is assigned based on probability thresholds.
8. Results Display – Confirm prediction results render properly in results.html.
9. API Endpoint Functionality – Test /api/predict returns valid JSON responses.
10. Error Handling – Validate system handles missing or invalid inputs gracefully**.**

User Stories / Requirements to be Tested

1. As a user, I want to enter startup details and receive a success prediction.
2. As a user, I want to see confidence scores to understand prediction reliability.
3. As a user, I want a clear success level indicator (High, Medium, Low).
4. As a developer, I want the API endpoint to accept JSON input and return prediction results.
5. As a stakeholder, I want the system to provide accurate and consistent predictions.
6. As a tester, I want the application to handle empty or incorrect inputs without crashing.

Test Cases:

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Pass/Fail
TC-001	Valid startup data submission	Step 1: Open application. Step 2: Enter valid numeric inputs in all fields Step 3 : Click Predict	Prediction result displayed with success %, failure %, and success level	Prediction displayed correctly with confidence scores	Pass
TC-002	Empty input field handling	Step 1: Open prediction form Step 2: Leave some fields blank Step 3: Click Predict	System replaces blank values with 0 OR shows validation message	Blank values converted to 0 and prediction generated	Pass
TC-003	Invalid data type in numeric field	Step 1: Enter "abc" in funding field Step 2: Submit form	Validation error displayed	Application crashed with 500 error	Fail
TC-004	Success level threshold check (75%)	Step 1: Submit input generating 75% probability	Success Level = "High Success"	Displayed as "Low Success"	Fail
TC-005	Large funding value input	Step 1: Enter extremely large funding amount Step 2: Submit	System processes without crash	Prediction generated successfully	Pass
TC-006	Mobile responsiveness test	Step 1: Open results page on mobile device	Proper responsive layout	Layout overlaps and misaligned	Fail
TC-007	Missing required JSON key in API	Step 1: Send JSON without one feature Step 2: Check response	Clear error message returned	Generic 400 error without message	Fail

Bug Tracking:

Bug ID	Bug Description	Steps to reproduce	Severity	Status	Additional feedback
BG-001	Server crash on text input in numeric field	Step 1: Enter text in numeric field Step 2: Submit	High	Open	Add backend validation and try-except handling
BG-002	Incorrect success level logic	Step 1: Submit data with 75% probability Step 2: Check result	High	In Progress	Threshold condition needs correction
BG-003	API error message not descriptive	Step 1: Send incomplete JSON Step 2: Observe response	Medium	Open	Improve API error handling response

Sign-off:

Tester Name: Yaswanth Chowdary Nallapati

Date: 12/02/2026

Signature: N. Yaswanth Chowdary