1. INTRODUCTION

1.1. Purpose

This test plan describes the testing approach and overall framework that will serve as a guide to test Paylater and related apps.

The document introduces:

Test Strategy: rules the test will be based on, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).

Execution Strategy: describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.

Test Management: process to handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation, team roster)

1.2. Project Overview

PayLater is a simple, entirely online lending platform that provides short-term loans to help cover unexpected expenses or urgent cash needs. Users can sign up and apply for a PayLater loan 24 hours a day, 7 days a week by simply downloading the app from the Google Playstore. The quick application process lets a user know his status within minutes and repayments can be made as at when due.

1.3. Audience

Project team members perform tasks specified in this document, and provide input and recommendations on this document.

Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.

The stakeholders’ representatives and participants (individuals as identified by the Project Management Office (PMO) Leads) may take part in the UAT test to ensure the business is aligned with the results of the test.

Technical Team ensures that the test plan and deliverables are in line with the design, provides the environment for testing and follows the procedures related to the fixes of defects.

Business analysts will provide their inputs on functional changes.

2. TEST STRATEGY

2.1. Test Objectives

The objective of the test is to verify that user can apply for loan, pay back and that the functionality of Paylater works according to the specifications. The test will execute and verify manual and automated test scripts, identify, fix and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing.

The final product of the test is twofold:

A production-ready software;

A set of stable test scripts that can be reused for Functional and UAT test execution

2.2. Test Assumptions

Key Assumptions

Production like data required will be available in the system prior to start of Functional Testing. This shall include but not limited to Paylater and non-Paylater client users, paylater, payroll and non-payroll loans, notifications, payments, referrals, discounts, debit cards.

General

Exploratory Testing would be carried out once the build is ready for testing

Performance testing is not considered for this estimation, but will be considered when huge amount of traffic is expected.

All the defects would come along with a snapshot

The Test Team will be provided with access to Test environment, including Admin and Database

The Test Team assumes all necessary inputs required during Test design and execution will be supported by Development/Business Analysts appropriately.

Test case design activities will be performed by QA team

Test environment and preparation activities will be owned by Dev Team

Business Analyst will review and sign-off all Test cases prepared by Test Team prior to start of Test execution

The defects will be tracked through JIRA. Any defect fixes planned will be shared with Test Team prior to applying the fixes on the Test environment

Project Manager/Business Analyst will review and sign-off all test deliverables

The project will provide test planning, test design and test execution support

Test team will manage the testing effort with close coordination with Project Manager/Business Analyst

Project team has the knowledge and experience necessary, or has received adequate training in the system, the project and the testing processes.

There is no environment downtime during test due to outages or defect fixes.

The system will be treated as a black box; if the information shows correctly online and in the reports, it will be assumed that the database is working properly.

Functional Testing

During Functional testing, testing team will use preloaded data which is available on the system at the time of execution

The Test Team will perform Functional testing only on release version

UAT

UAT test execution will be performed by selected end users and QA Group will provide their support on creating UAT script.

2.3. Test Principles

Testing will be focused on meeting the business objectives, cost efficiency, and quality.

There will be common, consistent procedures for all teams supporting testing activities.

Testing processes will be well defined, yet flexible, with the ability to change as needed.

Testing activities will build upon previous stages to avoid redundancy or duplication of effort.

Testing environment and data will emulate a production environment as much as possible.

Testing will be a repeatable, quantifiable, and measurable activity.

Testing will be divided into distinct phases, each with clearly defined objectives and goals.

There will be entrance and exit criteria.

2.4. Scope and Level of Testing

Exploratory

Purpose: the purpose of this test is to make sure critical defects are removed before the next levels of testing can start.

Scope: Client login, loan application, admin module, loan disbursement, loan repayment

Testers: Testing team.

Method: this exploratory testing is carried out in the application without any test scripts and documentation

Timing: at the beginning of each cycle.

Functional Test

Purpose: Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application.

Scope: The scope of Functional test is maintained on Zephyr. The scope is high level due to changes in the requirement.

Testers: Testing Team.

Method: The test will be performed according to Functional scripts, which are stored in Zephyr.

Timing: after Exploratory test is completed.

Test Acceptance Criteria

Approved Functional Specification document, Use case documents must be available prior to start of Test design phase.

Test cases approved and signed-off prior to start of Test execution

Development completed, unit tested with pass status and results shared to Testing team to avoid duplicate defects

Test environment configured and application in ready to use state.

TEST DELIVERABLES

S.No.

Deliverable Name

Author

Reviewer

1.

Test Plan

Test Lead

Project Manager/ Business Analyst

2.

Functional Test Cases

Test Team

Business Analyst Sign off

3.

Logging Defects in JIRA

Test Team

Test Lead/ Programming Lead

4.

Daily/weekly status report

Test Team/ Test Lead

Test Lead/ Project Manager

5.

Test Closure report

Test Lead

Project Manager

User Acceptance Test

Purpose: This test focuses on validating the business logic. It allows the end users to complete one final review of the system prior to deployment.

TESTERS: the UAT is performed by the end users.

METHOD: Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts.

TIMING: After all other levels of testing (Exploratory and Functional) are done. Only after this test is completed the product can be released to production.

3. EXECUTION STRATEGY

3.1. Entry and Exit Criteria

The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.

The exit criteria are the desirable conditions that need to be met in order to proceed with the implementation.

Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final “go-no go” decision.

Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.

Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

3.2. Test Cycles

There will be two cycles for functional testing. Each cycle will execute all the scripts .

The objective of the first cycle is to identify any blocking, critical defects, and most of the high defects. It is expected to use some work-around in order to get to all the scripts.

The objective of the second cycle is to identify remaining high and medium defects, remove the work-around from the first cycle, correct gaps in the scripts and obtain performance results.

UAT test will consist of one cycle.

3.3. Validation and Defect Management

It is expected that the testers execute all the scripts in each of the cycles. However it is recognized that the testers could also do additional testing if they identify a possible gap in the scripts. This is especially relevant in the second cycle, when the business analysts join in the execution of the test, since the business analysts have a deeper knowledge of the business processes. If a gap is identified, the scripts and traceability matrix will be updated and then a defect logged against the scripts.

The defects will be tracked through bugs logged in JIRA and test cases in Zephyr while the technical team will work on fixes.

It is the responsibility of the tester to open the defects, link them to the corresponding script, assign an initial severity and status, retest and close the defect; it is the responsibility of the Defect Manager to review the severity of the defects and facilitate with the technical team the fix and its implementation, communicate with testers when the test can continue or should be halted, request the tester to retest, and modify status as the defect progresses through the cycle; it is the responsibility of the technical team to review bug tickets on a daily basis, ask for details if necessary, fix the defect, communicate to the Defect Manager when the fix is done, implement the solution per the Defect Manager request.

Severity

Impact

1 (Critical)

This bug is critical enough to prevent loan application or disbursement

Crash the application or system, cause file corruption, or cause potential data loss

2 (High)

It causes a lack of vital program functionality with workaround.

3 (Medium)

This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired

This bug prevents other areas of the product from being tested. However other areas can be independently tested.

4 (Low)

There is an insufficient or unclear error message, which has minimum impact on product use.

5(Cosmetic)

There is an insufficient or unclear error message that has no impact on product use.

3.4. Test Metrics

Test metrics to measure the progress and level of success of the test will be developed and shared with the project manager for approval. The below are some of the metrics

Report

Description

Frequency

Test preparation & Execution Status

To report on % complete, %WIP, % Pass, % Fail

Defects severity wise Status – Open, closed, any other Status

Weekly (optional)

Daily execution

status

To report on Pass, Fail, Total defects, highlight Showstopper/ Critical defects

Daily Standups

Project Weekly Status report

Project driven reporting (As requested by PM)

Weekly – If project team needs weekly update apart from daily and there is template available with project team to use.

4. TEST MANAGEMENT

Test management are the activities invloved in managing the computer software testing process. A test management is also performed using tools to manage both types of tests, automated and manual, that have been previously specified by a test procedure.

4.1. Manual Test Execution Management

The tester performs all the test steps manually and inform the system of the result. The test cases are associated with other test artifacts such as test plans, test scripts, test environments, test case execution records, and test suites. The test scrpits and test cases are managed in Zephyr and Jira repectivley. After the manual execution is completed, a UAT test script is created and shared with selected end users

4.2 Automated Test Execution Management

The Paylater app automated test execution is managed by the Appium fraimework - an open-source tool for automating native, mobile web, and hybrid applications on iOS and Android platforms. It is built with Java programming language and reporting managed with TestNG and Maven for build management.