CS631G – Team 3



SYSTEM TEST PLAN

Xero

(Cloud-based Accounting System)

Authors: Tarjanee Desai, Tanzil Bilal Mohammed, Kalpana Komatineni, Shloka Gupta, Rahul Anand Nayanegali, Harsh Moradiya, Sakshi Singh, Sai Sandeep Mandava, Madhu Kiran Thalluri

QA Manager: Tarjanee Desai

Date: 04/02/2024

**Table of Contents**

[Introduction (Tarjanee Desai) 2](#_Toc1359782971)

[Test Items (Tanzil Bilal Mohammed) 3](#_Toc1990408433)

[Features to be tested (Kalpana Komatineni) 3](#_Toc1087384923)

[Features not to be tested (Kalpana Komatineni) 4](#_Toc1383279373)

[Approach (Shloka Gupta) 5](#_Toc1606679596)

[Item Pass/Fail Criteria (Rahul Anand Nayanegali) 5](#_Toc1413392572)

[Suspension Criteria and resumption requirements (Rahul Anand Nayanegali) 6](#_Toc1468753271)

[Test deliverables (Harsh Moradiya) 6](#_Toc1208267734)

[Testing tasks (Harsh Moradiya) 7](#_Toc621014314)

[Enviornmental needs (Sakshi Singh) 8](#_Toc998664915)

[Responsibilities (Sakshi Singh) 8](#_Toc516079959)

[Staffing and training needs (Sai Sandeep Mandava) 9](#_Toc1144665694)

[Test Cycles and Schedule (Tarjanee Desai) 10](#_Toc1290244192)

[Risks and contingencies (Madhu Kiran Thalluri) 11](#_Toc1499737959)

# Introduction (Tarjanee Desai)

This document outlines the Test Plan for the Xero Cloud-Based Accounting Application, aiming to provide stakeholders with a clear understanding of testing objectives and methodologies. It covers test items, features to be tested, approach, pass/fail criteria, suspension criteria, test deliverables, testing tasks, environmental needs, responsibilities, staffing/training needs, schedule, and risks/contingencies. By adhering to this plan, we aim to validate the Xero application effectively, ensuring its reliability and functionality for users and stakeholders.

# Test Items (Tanzil Bilal Mohammed)

The Test items includes the following modules in the Xero application:

1. Purchase Orders
2. Expense Claims
3. Accounts Payable

# Features to be tested (Kalpana Komatineni)

**1. Purchase Order Module:**

Features to be tested:

* Purchase order creation: Verify that users can create purchase orders accurately and efficiently.
* Vendor management: Test functionality related to adding, editing, and managing vendor information.
* Purchase order approval workflows: Test the functionality of approval workflows for purchase orders.
* Integration with other modules: Verify integration with inventory management, accounts payable, and financial reporting modules.
* Error handling: Ensure appropriate error messages are displayed when users input incorrect data.
* Purchase order tracking: Verify that users can track the status of purchase orders throughout the procurement process.

**2. Expense Claim Module:**

Features to be tested:

* Expense claim submission: Verify that users can submit expense claims accurately.
* Approval workflows: Test the functionality of approval workflows for expense claims.
* Expense tracking: Ensure that expenses are accurately tracked and categorized.
* Integration with other modules: Verify integration with accounts payable and general ledger modules.
* Error handling: Ensure appropriate error messages are displayed when users input incorrect data.
* Reimbursement processing: Test the functionality related to processing reimbursements for approved expense claims.

**3. Accounts Payable Module:**

Features to be tested:

* Invoice processing: Verify that users can process invoices accurately and efficiently.
* Payment processing: Test the functionality related to recording and processing payments to vendors.
* Vendor management: Ensure that vendor information is accurately recorded and maintained.
* Integration with other modules: Verify integration with purchase order module, expense claim module, and general ledger module.
* Error handling: Ensure appropriate error messages are displayed when users input incorrect data.
* Aging analysis: Test functionality related to analyzing outstanding vendor balances.

# Features not to be tested (Kalpana Komatineni)

**1. Purchase Order Module:**

Features that might not need testing or require minimal testing:

* Sales-related features: Since the focus is on procurement, features related to sales orders or customer invoicing may not require extensive testing within the purchase order module.
* Non-purchase related financial features: Features such as general ledger management or financial reporting, which are not directly related to purchase orders, may not need testing within this module.

**2. Expense Claim Module:**

Features that might not need testing or require minimal testing:

* Sales-related features: Like the purchase order module, features related to sales orders or customer invoicing may not require extensive testing within the expense claim module.
* Non-expense related financial features: Features such as payroll management or invoicing may not need testing within this module.

**3. Accounts Payable Module:**

Features that might not need testing or require minimal testing:

* Sales-related features: As with the other modules, features related to sales orders or customer invoicing may not require extensive testing within the account's payable module.
* Non-accounts payable related financial features: Features such as payroll management or financial reporting may not need testing within this module.

# Approach (Shloka Gupta)

**Methodology**: Utilize a combination of manual and automated testing strategies to evaluate the specified features. This includes:

* **Functional Testing**: To verify each feature works according to requirements.
* **Integration Testing**: To ensure different parts of the application work together seamlessly.
* **Regression Testing**: To confirm that new changes haven't adversely affected existing functionalities.
* **User Acceptance Testing (UAT)**: To validate the application with end-users for real-world scenario effectiveness.

# Item Pass/Fail Criteria (Rahul Anand Nayanegali)

The pass/fail criteria for the Xero application will be determined by the following exit criteria, which must be met before testing can be considered complete:

* **Test Design Completeness**: All functional requirements must be covered by test cases. This ensures that every aspect of the application has been considered and that there are specific tests designed to verify each feature's functionality.
* **Test Execution Completeness**: All test cases must have been executed. This criterion confirms that the testing team has carried out all planned tests and that no part of the test plan has been overlooked.
* **Product Stability Assessment**: No defects of Critical & High severity remain open. This means that any defects that could cause severe disruption or pose significant risks to the functionality of the application must be resolved. Lower severity defects may be documented for future resolution, but they should not impede the release of the software.

To align with these exit criteria, the specific pass/fail criteria for the core features of the Xero application are as follows:

* **Purchase Order Features (08.01 to 08.17)**: All test cases associated with purchase order features must be executed, and the features must perform as expected without critical or high-severity defects. This includes creation, editing, deletion, viewing, searching, filtering, sorting, printing, sending, copying, uploading files, approving, billing, adding notes, and tracking history.
* **Expense Claims Features (09.01 to 09.10)**: All test cases for expense claim processes must be executed and pass without any critical or high-severity defects. This encompasses creation, editing, deletion, review, approval, decline, payment, and exploration of expense claims.
* **Vendor Bills and Contacts Features (11.01 to 11.11)**: All test cases for vendor bills and contacts must be executed and pass without any critical or high-severity defects. This includes accurate and timely recording, reviewing, scheduling, disbursement, updating, and adding to accounts payable, as well as functional and error-free supplier and contact management.

# Suspension Criteria and resumption requirements (Rahul Anand Nayanegali)

Testing activities may be suspended under the following conditions:

* **Critical Bugs**: Discovery of critical bugs that prevent further testing of the application, such as system crashes or data corruption.
* **Environment Downtime**: Unavailability of the test environment due to maintenance or unexpected outages.
* **Resource Unavailability**: Lack of necessary resources, such as personnel or tools, to continue testing effectively.

Resumption of testing will occur when:

* **Critical Bugs Resolved**: The critical bugs have been fixed, and the application is stable enough to resume testing.
* **Environment Restored**: The test environment is back online and fully operational.
* **Resource Availability**: Required resources become available to continue the testing process.

# Test deliverables (Harsh Moradiya)

Test Plan: This document outlines the overall testing approach, objectives, scope, schedule, and resources required for testing the Xero application.

Test Design Specifications: Detailed specifications outlining the design of test cases, including test scenarios, inputs, expected outputs, and test conditions for testing various functionalities of the Xero application.

Test Case Specifications: Specific test cases derived from the test design specifications, providing step-by-step instructions on how to execute tests to validate different aspects of the Xero application.

Test Procedure Specifications: Procedures detailing the steps to be followed for executing test cases, including prerequisites, setup instructions, and expected outcomes.

Test Item Transmittal Reports: Reports documenting the transfer of test items, such as test cases, procedures, and other testing artifacts, between different testing phases or teams.

Test Logs: Logs recording the execution of test cases, including details such as test case ID, description, actual results, and any deviations or anomalies encountered during testing.

Test Incident Reports: Reports documenting any incidents or defects identified during testing, including detailed descriptions of the issue, steps to reproduce, severity, and priority.

Test Summary Reports: Comprehensive reports summarizing the overall testing activities, including test coverage, test results, defect metrics, and recommendations for further improvement.

Test Data: Data sets used for testing various scenarios and functionalities of the Xero application, including sample transactions, user profiles, and configurations.

Test Tools: Any specialized tools or software used to aid in the testing process, such as test management tools, automation frameworks, performance testing tools, etc.

# Testing tasks (Harsh Moradiya)

Planning: This involves creating a comprehensive test plan that outlines the testing approach, objectives, scope, schedule, and resources required for the project.

Designing Test Cases: Harsh is responsible for developing detailed test cases based on the requirements and specifications of the software being tested. These test cases include step-by-step instructions, inputs, expected outputs, and any preconditions.

Setting up the Test Environment: Harsh ensures that the test environment is properly configured to mimic the production environment as closely as possible. This may involve installing software, configuring hardware, and preparing test data.

Executing Tests: Harsh executes the test cases according to the test plan and records the results. This involves following the steps outlined in the test cases, entering inputs, comparing actual results with expected results, and documenting any deviations or issues encountered.

Logging Defects: When Harsh identifies any issues or defects during testing, he logs them into a defect tracking system. This includes providing detailed descriptions of the issues, steps to reproduce them, and any relevant screenshots or logs.

Retesting Fixes: After developers address and fix the reported defects, Harsh performs retesting to verify that the fixes have been implemented correctly and that the issues have been resolved.

Reporting: Harsh prepares test reports summarizing the testing activities, results, and findings. These reports are typically shared with project stakeholders, including developers, managers, and clients, to provide visibility into the quality of the software and any remaining risks or issues.

By effectively managing these testing tasks to ensure the quality, reliability, and functionality of the software being developed.

# Enviornmental needs (Sakshi Singh)

The test environment for the Xero application must replicate the production environment as closely as possible and include:

* **Hardware**: Adequate servers, workstations, and networking equipment to support the application and testing tools.
* **Software**: All necessary software, including the Xero application, operating systems, databases, and any third-party integrations.
* **Tools**: Test management tools, defect tracking systems (e.g., JIRA), and any specialized tools for performance or security testing.

# Responsibilities (Sakshi Singh)

QA Manager

* Oversee the entire quality of the product development.
* Ensure that the testing process is carried out according to the test plan.
* Coordinate with other team leads to align the testing schedule with the development schedule.
* Manage the QA team, assign tasks, and ensure that each team member understands their responsibilities.
* Monitor the testing process and provide regular progress updates to stakeholders.
* Address any issues that arise during the testing process and ensure they are resolved in a timely manner.

QA Analyst

* Conduct thorough testing of the software to ensure it meets quality standards and requirements.
* Identify, report, and track bugs, defects, and issues within the software.
* Collaborate with the development team to ensure defects are understood and resolved.
* Execute test cases and report on test execution results.
* Participate in test planning and strategy development.
* Assist in the creation and maintenance of test documentation.

Business Analyst

* Work closely with stakeholders to understand business requirements and ensure they are accurately reflected in the test plan.
* Translate business requirements into detailed functional specifications for the testing team.
* Review test cases to ensure they align with business requirements and objectives.
* Provide input on the prioritization of testing activities based on business impact.
* Facilitate communication between the business stakeholders and the testing team.

Software Tester

* Execute test cases as per the test plan and document the outcomes.
* Report defects and issues found during testing in a clear and concise manner.
* Collaborate with QA analysts to understand the testing requirements and scope.
* Assist in the maintenance of the testing environment and setup.
* Participate in regular team meetings to discuss testing progress and challenges.
* Contribute to the continuous improvement of testing processes and practices.

# Staffing and training needs (Sai Sandeep Mandava)

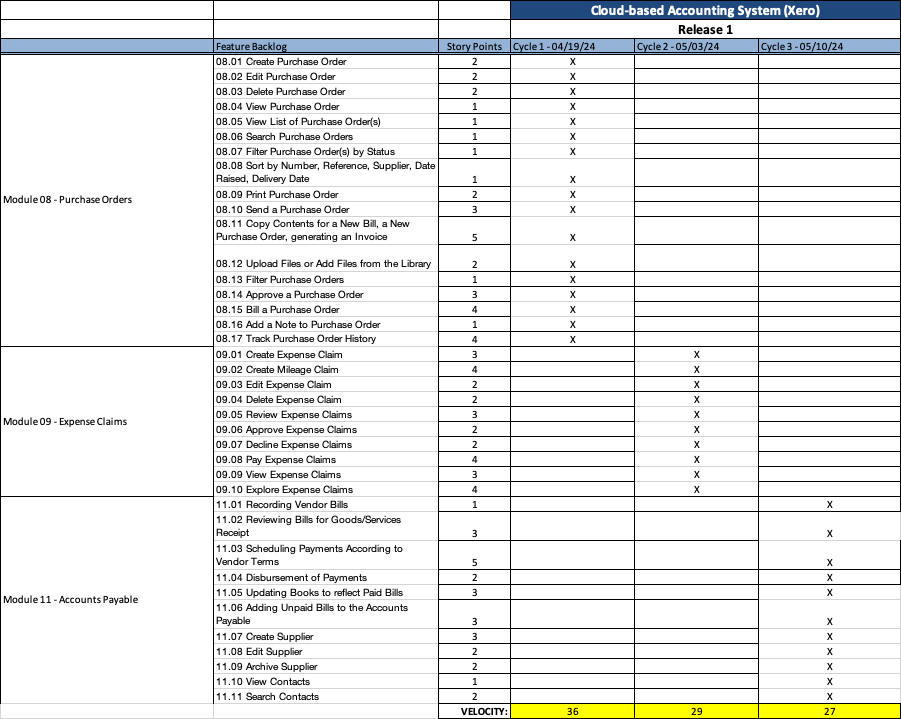
The testing team will consist of QA analysts with varying levels of expertise to cover all entitlements and features. Training will be provided as follows:

* **New Tools and Technologies**: Training on new testing tools or technologies that will be used during the testing process.
* **Xero Application Features**: Training on the Xero application's features, especially for new team members or when new features are introduced.
* **Testing Best Practices**: Ongoing training on best practices in software testing to ensure high-quality results.

# Test Cycles and Schedule (Tarjanee Desai)

System testing will be executed in three cycles:

* Cycle 1 – Focuses on testing all the core features of the Purchase Order Module
* Cycle 2 – Focuses on testing all the core features of the Expense Claims Module
* Cycle 3 – Focuses on testing all the core features of the Accounts Payable Module



# Risks and contingencies (Madhu Kiran Thalluri)

**Contingency:** To mitigate this risk, a robust defect management process will be implemented, focusing specifically on the accounts payable module. This includes prioritizing defects based on severity and impact, establishing efficient communication channels between testing and development teams, and implementing proactive measures to prevent the recurrence of similar defects.

**Purchase Orders Module:**

**Risk 1: Limited Testing Resources:**

**Risk:** Limited testing resources may result in delays in testing the purchase orders module. Insufficient staffing or inadequate testing tools can hinder the progress of testing activities, leading to potential project delays.

**Contingency:** To mitigate this risk, a proactive approach to resource management will be adopted. This may include reallocating resources from non-critical tasks, outsourcing specific testing activities related to purchase orders, or leveraging automation tools to improve testing efficiency.

**Risk 2: Changes in Scope Objectives:**

**Risk:** Any changes to the scope objectives related to purchase orders can cause delays or additional work in the testing process. Scope changes, whether due to evolving business requirements or stakeholder feedback, may necessitate adjustments to testing plans and timelines.

**Contingency:** To address this risk, change management procedures will be implemented specifically for the purchase orders module. Clear communication channels will be established with stakeholders to manage expectations and prioritize testing efforts accordingly.

**Risk 3: High Volume of Defects:**

**Risk:** Many defects in the purchase orders module may require longer time to fix and complete testing. The discovery of numerous defects during testing can overwhelm the testing team, leading to extended defect resolution cycles and potential delays in project delivery.

**Contingency:** To mitigate this risk, a robust defect management process will be implemented, focusing specifically on the purchase orders module. This includes prioritizing defects based on severity and impact, establishing efficient communication channels between testing and development teams, and implementing proactive measures to prevent the recurrence of similar defects.

**Expense Claims Module:**

**Risks and Contingencies:**

**Risk 1: Limited Testing Resources:**

**Risk:** Limited testing resources may result in delays in testing the expense claims module. Insufficient staffing or inadequate testing tools can hinder the progress of testing activities, leading to potential project delays.

**Contingency:** To mitigate this risk, a proactive approach to resource management will be adopted specifically for testing the expense claims module. This may include reallocating resources from non-critical tasks, outsourcing specific testing activities related to expense claims, or leveraging automation tools to improve testing efficiency.

**Risk 2: Changes in Scope Objectives:**

**Risk:** Any changes to the scope objectives related to expense claims can cause delays or additional work in the testing process. Scope changes, whether due to evolving business requirements or stakeholder feedback, may necessitate adjustments to testing plans and timelines.

**Contingency:** To address this risk, change management procedures will be implemented specifically for the expense claims module. Clear communication channels will be established with stakeholders to manage expectations and prioritize testing efforts accordingly.

**Risk 3: High Volume of Defects:**

**Risk:** Many defects in the expense claims module may require longer time to fix and complete testing. The discovery of numerous defects during testing can overwhelm the testing team, leading to extended defect resolution cycles and potential delays in project delivery.

**Contingency:** To mitigate this risk, a robust defect management process will be implemented, focusing specifically on the expense claims module. This includes prioritizing defects based on severity and impact, establishing efficient communication channels between testing and development teams, and implementing proactive measures to prevent the recurrence of similar defects.

**Accounts Payable Module:**

**Risks and Contingencies:**

**Risk 1: Limited Testing Resources:**

**Risk:** Limited testing resources may result in delays in testing the accounts payable module. Insufficient staffing or inadequate testing tools can hinder the progress of testing activities, leading to potential project delays.

**Contingency:** To mitigate this risk, a proactive approach to resource management will be adopted specifically for testing the accounts payable module. This may include reallocating resources from non-critical tasks, outsourcing specific testing activities related to accounts payable, or leveraging automation tools to improve testing efficiency.

**Risk 2: Changes in Scope Objectives:**

**Risk:** Any changes to the scope of objectives related to accounts payable can cause delays or additional work in the testing process. Scope changes, whether due to evolving business requirements or stakeholder feedback, may necessitate adjustments to testing plans and timelines.

**Contingency:** To address this risk, change management procedures will be implemented specifically for the accounts payable module. Clear communication channels will be established with stakeholders to manage expectations and prioritize testing efforts accordingly.

**Risk 3: High Volume of Defects:**

**Risk:** Many defects in the accounts payable module may require longer time to fix and complete testing. The discovery of numerous defects during testing can overwhelm the testing team, leading to extended defect resolution cycles and potential delays in project delivery.