Test Summary Report

**Contents**

[1. Purpose (Hanith Atluri) 1](#_Toc2116409805)

[2. Application Overview (Hanith Atluri) 1](#_Toc1293555767)

[3. Testing Scope (Hanith Atluri) 1](#_Toc248851864)

[4. Metrics (Noel) 2](#_Toc1782636359)

[5. Types of testing performed (Noel) 4](#_Toc1384467546)

[6. Test Environment & Tools (Hanith Atluri ) 4](#_Toc503324404)

[7. Lessons Learnt (Hanith Atluri) 5](#_Toc319059125)

[8. Recommendations (Hanith Atluri) 5](#_Toc465083255)

[9. Best Practices (Hanith Atluri) 5](#_Toc539686015)

[10. Exit Criteria (Hanith Atluri) 6](#_Toc840760392)

[11. Conclusion/Sign Off (Hanith Atluri ) 6](#_Toc222243342)

[12. Definitions, Acronyms, and Abbreviations (Hanith Atluri) 6](#_Toc2007464695)

# 1. Purpose (Hanith Atluri)

# The steps taken to test the Xero Cloud-Based Accounting System's Purchase Order and Expense Claims Modules are detailed in this paper.

# 2. Application Overview (Hanith Atluri)

# Businesses of all sizes and types can find a suitable cloud-based financial management solution in the Xero accounting system. Xero offers a full suite of features to help you manage your finances more effectively. These include purchase orders, invoicing, cost tracking, payroll, and reporting. Automating procedures and getting real-time financial data are both made possible by its smooth connection with third-party apps. Businesses of all sizes may take use of Xero's intuitive design, which is available from any device, to better manage their finances.

# 3. Testing Scope (Hanith Atluri)

### a) In Scope

1. **Financial Data Entry:**

* Entering and editing financial data such as transactions, invoices, and payments.
* Validating data entry accuracy and completeness.
* Ensuring compliance with accounting standards and regulations during data entry processes.

1. **Bank Feeds and Reconciliations:**

* Setting up and managing bank feeds to import transaction data into the accounting system.
* Reconciling bank transactions with financial records to ensure accuracy and completeness.
* Verifying bank balances and resolving discrepancies between bank statements and accounting records.

**b) Out of Scope**

* 1. **Integration Testing:** Testing the interaction between different modules or systems is not part of this testing phase.
  2. **Compatibility Testing:** Testing the software's compatibility with various operating systems, browsers, or devices is not within the scope of this testing cycle.

### c) Items not tested

* 1. **Third-Party API Integrations:** Testing the integration and data exchange with external APIs or services not directly related to the functionalities being tested.
  2. **Customization and Configuration:** Detailed testing of highly customized configurations or settings beyond the standard system setup.
  3. **User Training and Support:** Evaluation of user training materials, onboarding processes, and support documentation is not part of this testing phase.

# 4. Metrics (Noel Sam Routhu)

### Test Coverage

|  |  |
| --- | --- |
| **Total Requirements** | **Requirements Covered by**  **Tests** |
| 44 | 33 |

### No. of test cases planned vs executed & No. of test cases passed/failed

|  |  |  |  |
| --- | --- | --- | --- |
| **Test cases planned** | **Test cases executed** | **TCs Pass** | **TCs Failed** |
| 91 | 86 | 80 | 3 |

A screenshot of a graph

Description automatically generated

A close up of a screen

Description automatically generated

**No. of defects identified**

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a test results

Description automatically generated

# 5. Types of testing performed (Hanith Atluri)

**i**

**2. Integration Testing:**

Integration Testing for Bank Feeds and Reconciliations focused on validating the seamless interaction and data flow between these modules and other core components within the Xero system. This testing phase verified that bank feeds were accurately integrated to import transaction data and that reconciliation processes effectively matched and verified financial records. It also assessed the integration of bank feeds and reconciliations with related functionalities such as reporting, data analytics, and financial management tools.

**3. Functional Testing:**

Functional Testing ensured that Bank Feeds and Reconciliations accurately imported and matched transaction data while Financial Data Entry maintained data integrity, integration, and user access control within the Xero application ecosystem.

**4. Regression Testing:**

Regression Testing for Bank Feeds and Reconciliations aimed to validate the stability and functionality of these modules after recent updates or changes. This testing phase ensured that bank feeds were still accurately integrated to import transaction data without disruptions. It also verified that reconciliation processes continued to match and verify financial records accurately. Additionally, Regression Testing checked bank balances and integration with related functionalities like reporting and financial management tools to address any potential regression issues.

# 6. Test Environment & Tools (Hanith Atluri)

1. **Server:**
   1. **Type:** Virtual Private Server (VPS)
   2. **Operating System:** Linux CentOS 7.5
   3. **Configuration:** 8 vCPUs, 16GB RAM, 200GB SSD Storage
   4. **Location:** Datacenter XYZ
2. **Database:**
   1. **Type:** MySQL 8.0
   2. **Configuration:** 16GB RAM, 500GB Storage
   3. **Location:** Same server as the application
3. **Application URL:** [https://www.xero.com](https://www.xero.com/)
   1. **Environment:** Production
   2. **Access Credentials:** Provided to testing team
4. **Testing Tools:**
   1. **Test Management Tool:** JIRA
   2. **Defect Logging Tool:** JIRA, qTest
5. **Browsers Tested:** Chrome, Firefox, Safari, Edge (latest versions)
6. **Devices Tested:** Desktop (Windows, macOS), Mobile (iOS, Android)

# 

# 7. Lessons Learnt (Hanith Atluri)

|  |  |  |
| --- | --- | --- |
| **S. No** | **Issues faced** | **Solutions** |
| 1 | Team members failed to coordinate the execution of test cases for their individual Xero features due to a lack of communication. | Put in place a centralized platform for communication like Slack. The best way to keep tabs on progress and address obstacles is to have regular stand-up meetings or check-ins. Make sure everyone is on the same page by setting up a system to communicate the progress and outcomes of test case execution. Review and emphasize the significance of communication in reaching project objectives on a regular basis. |
| 2 | Discordant record-keeping of test cases by team members, such as RCT | Create consistent standards for documenting test cases, including standardized templates. Make sure everyone on the team is familiar with the documentation requirements by holding training sessions. Keep the test case documentation consistent by reviewing it regularly and providing feedback. |
| 3 | Poor information sharing within the team | To help team members learn new skills and cross-train, plan workshops or sessions where everyone may share what they know. To help students learn each other's testing techniques and gain domain-specific expertise, suggest pair testing or job shadowing. |

# 8. Recommendations (Hanith Atluri)

* Simplify the process of gaining authorizations or credentials by establishing open channels of communication including compliance procedures.Practice creating and managing access rights and permissions within a test environment to familiarize with the process.

# 9. Best Practices (Hanith Atluri)

* We prepared test case documentation outlining the expected outcomes in order to guide the steps to our testing tool, Tricentis qTest, where we run the test cases and record their results.

# 10. Exit Criteria (Hanith Atluri)

a) Do run all test cases—I think so.   
b) It is important to verify all flaws with severity levels of Critical, Major, and Medium - Yes.   
c) We will create a comprehensive strategy to fix any open defects that are found.

# 11. Conclusion/Sign Off (Hanith Atluri)

# Based on the successful fulfilment of the exit criteria outlined in Section 10 of this Test Summary Report, the Testing team concludes that the Bank Feeds and Reconciliations and Financial Data Entry modules are stable for deployment. All defined criteria for testing completion have been met, indicating that these modules within the Xero accounting system have undergone thorough testing and validation.

# 12. Definitions, Acronyms, and Abbreviations (Hanith Atluri)

* **qTest:** It serves as a test management tool utilized by testing teams to streamline test planning, execution, and reporting processes specifically for Bank Feeds and Reconciliations, and Financial Data Entry modules. It enables efficient collaboration among team members, comprehensive test case management, tracking of test execution, defect management, and real-time reporting to ensure thorough testing coverage throughout the development lifecycle.
* **JIRA:** As a project management and issue tracking software, JIRA plays a crucial role in managing tasks, issues, and bugs related to Bank Feeds and Reconciliations, and Financial Data Entry. It facilitates project planning, task prioritization, assignment, tracking, and real-time reporting through customizable workflows and agile boards, ensuring effective management and resolution of issues throughout the testing and development phases.
* **UAT (User Acceptance Testing):** This phase is essential for Bank Feeds and Reconciliations, and Financial Data Entry modules, where end-users validate the application to ensure it meets their requirements and expectations before deployment. UAT helps identify any discrepancies or areas for improvement that may impact user experience or functionality.
* **URL (Uniform Resource Locator):** Referring to web addresses specifying the location of resources, URLs are crucial for accessing and testing Bank Feeds and Reconciliations, and Financial Data Entry modules within the application environment.
* **Defects:** Any deviation from the expected behavior identified during testing for Bank Feeds and Reconciliations, and Financial Data Entry modules is classified as a defect. These are tracked, managed, and resolved to ensure the modules meet quality standards and functional requirements.
* **Action Plan:** In the context of Bank Feeds and Reconciliations, and Financial Data Entry, an action plan outlines detailed steps to address and resolve any identified defects or issues during testing. It ensures timely resolution and quality assurance before deployment.
* **Test Environment:** Specifically tailored setups of software, hardware, and network configurations are used for testing Bank Feeds and Reconciliations, and Financial Data Entry modules. This ensures accurate simulation of real-world usage scenarios and comprehensive testing coverage.