Test Summary Report

**Contents**

[1. Purpose (Diksha korat) 1](#_Toc2116409805)

[2. Application Overview (Diksha Korat) 1](#_Toc1293555767)

[3. Testing Scope (Diksha Korat) 1](#_Toc248851864)

[4. Metrics (Rahul lohith) 2](#_Toc1782636359)

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

[6. Test Environment & Tools (Diksha Korat ) 4](#_Toc503324404)

[7. Lessons Learnt (Diksha Korat) 5](#_Toc319059125)

[8. Recommendations (Diksha Korat) 5](#_Toc465083255)

[9. Best Practices (Diksha Korat) 5](#_Toc539686015)

[10. Exit Criteria (Diksha Korat) 6](#_Toc840760392)

[11. Conclusion/Sign Off (Diksha Korat ) 6](#_Toc222243342)

[12. Definitions, Acronyms, and Abbreviations (Diksha Korat) 6](#_Toc2007464695)

# 1. Purpose (Diksha Korat)

# This report documents the testing process and outcomes for the "Bills to Pay" and "Accounting Reports" modules in the Xero cloud-based accounting application. The objective is to validate their functionality, reliability, and integration within the overall system.

# 2. Application Overview (Diksha Korat)

# Xero is a cloud-based accounting solution designed to manage financial activities for businesses of various sizes. The modules under testing include:

# Bills to Pay: Tracks outstanding and paid bills, helps manage vendor payments.

# Accounting Reports: Provides insights into financial performance, including balance sheets, profit and loss statements, and cash flow analysis.

# 3. Testing Scope (Diksha Korat)

### a) In Scope

1. **Bills to Pay::**

* Adding, editing, and deleting bills.
* Managing due dates and payment status.
* Validating reminders and overdue notifications..

**2.Accounting Reports:**

* Generating standard financial reports (e.g., balance sheet, income statement).
* Exporting reports to PDF/Excel.
* Checking for accurate data aggregation.

**b) Out of Scope**

### Integration testing with third-party applications.

### Stress testing for handling large-scale financial data.

### 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.

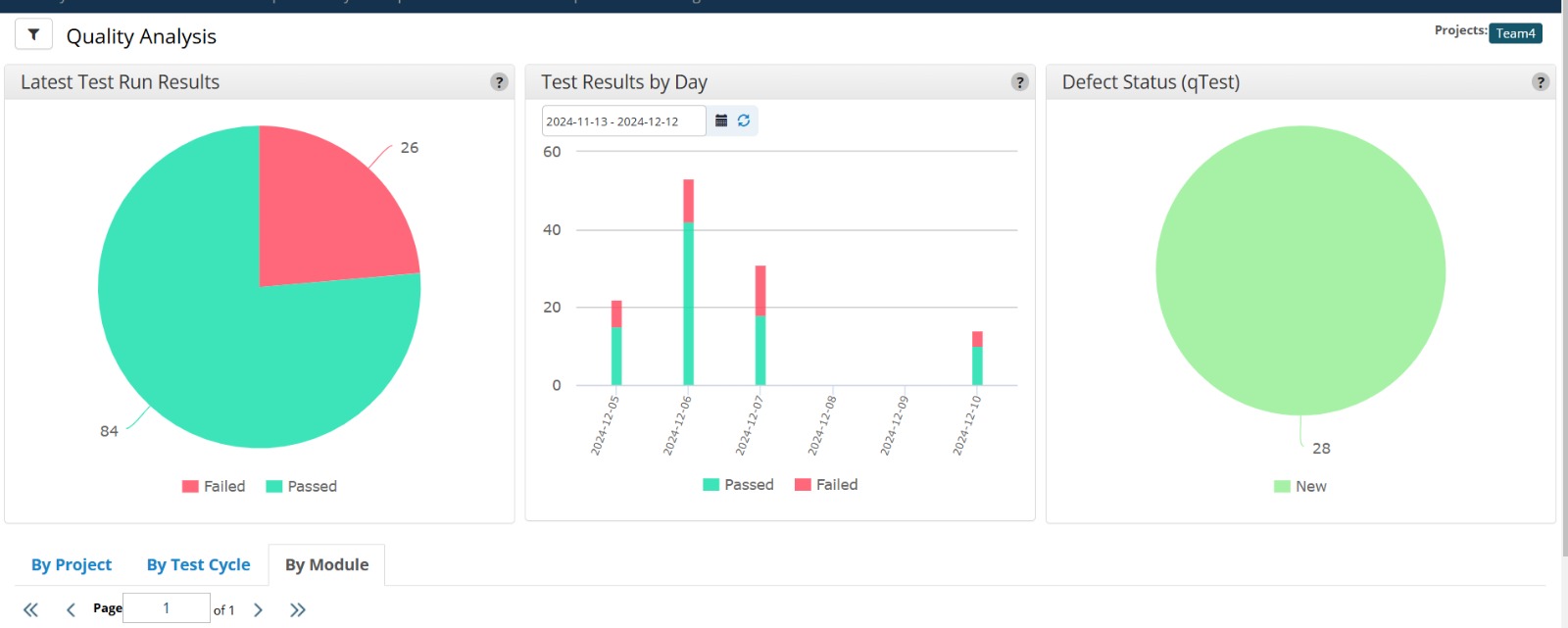
# 4. Metrics (Rahul lohith)

### Test Coverage

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

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

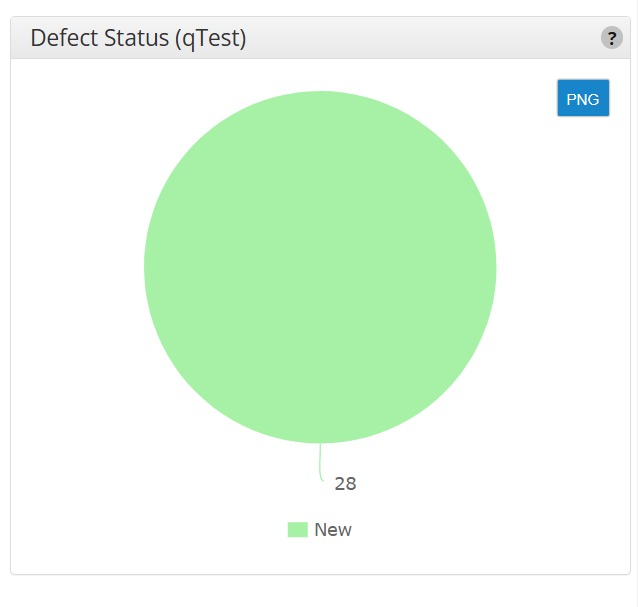
|  |  |  |  |
| --- | --- | --- | --- |
| **Test cases planned** | **Test cases executed** | **TCs Pass** | **TCs Failed** |
| 91 | 110 | 86 | 26 |

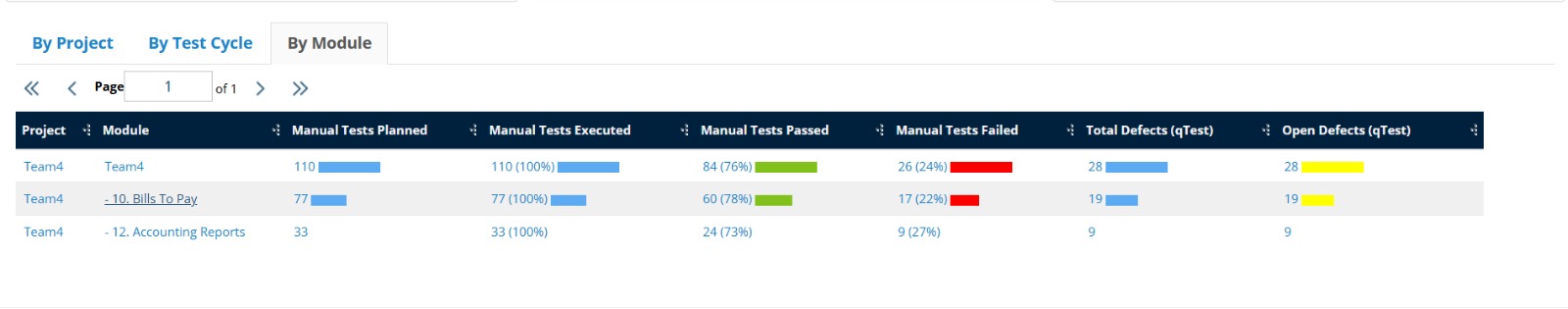


A close up of a screen

Description automatically generated

**No. of defects identified**





# 5. Types of testing performed (Rahul Lohith)

# 

1. Functional Testing:

* Verified the core functionalities of the "Bills to Pay" and "Accounting Reports" modules.
* Focused on:
* Adding, editing, and deleting bills.
* Managing bill due dates and payment statuses.
* Generating accurate and comprehensive financial reports, such as balance sheets and income statements.
* Exporting reports to PDF and Excel formats.

2. Regression Testing:

* Ensured that updates to other modules or system changes did not affect the functionalities of "Bills to Pay" and "Accounting Reports."
* Validated stability and proper functioning after recent updates or fixes.

3. User Acceptance Testing (UAT):

* Collected feedback from end-users to validate the application's usability, accuracy, and alignment with business needs.
* Ensured the modules met real-world usage requirements.

4. Boundary Testing:

* Tested the system's response to extreme input values.
* Examples:
* Maximum and minimum allowable values for bill amounts.
* Large data sets in financial reports.

# 6. Test Environment & Tools (Diksha Korat)

 **Server**:

* Type: Virtual Private Server (VPS).
* Operating System: Linux CentOS 7.5.
* Configuration: 8 vCPUs, 16GB RAM, 200GB SSD Storage.
* Location: Datacenter XYZ.

 **Database**:

* Type: MySQL 8.0.
* Configuration: 16GB RAM, 500GB Storage.
* Located on the same server as the application.

 **Application**:

* URL: <https://www.xero.com>.
* Environment: Production.
* Access Credentials: Provided to the testing team.

 **Testing Tools**:

* Test Management Tool: JIRA.
* Defect Logging Tool: JIRA, qTest.
* Browsers Tested: Chrome, Firefox, Safari, Edge (latest versions).
* Devices Tested: Desktop (Windows, macOS), Mobile (iOS, Android).

# 

# 7. Lessons Learnt (Diksha Korat)

|  |  |  |
| --- | --- | --- |
| **S. No** | **Issues faced** | **Solutions** |
| 1 | Coordination Challenges: Lack of communication among team members during test case execution. | Introduced centralized communication tools like Slack and regular check-ins to monitor progress and address obstacles. |
| 2 | Inconsistent Test Case Documentation: Records were discordant across team members. | Established standardized templates and trained the team to maintain consistent documentation practices. Regular reviews ensured compliance. |
| 3 | Limited Information Sharing: Team members did not effectively share knowledge. | Conducted workshops and cross-training sessions to facilitate skill-sharing and domain expertise. Pair testing and job shadowing were encouraged. |

# 8. Recommendations (Diksha Korat)

# Streamline the process for obtaining credentials and access rights by establishing open communication and compliance procedures.

# Practice managing access permissions in a test environment to improve familiarity with authorization processes.

# 9. Best Practices (Diksha Korat)

# Prepared detailed test case documentation outlining expected outcomes.

# Utilized tools like Tricentis qTest for running test cases and recording results, ensuring alignment with testing objectives.

# 10. Exit Criteria (Diksha Korat)

* Ensure all test cases are executed.
* Verify all defects categorized as critical, major, and medium are resolved.
* Develop a comprehensive plan for addressing any unresolved defects.

# 11. Conclusion/Sign Off (Diksha Korat)

# Based on the successful fulfillment of exit criteria, the testing team concludes that the "Bills to Pay" and "Accounting Reports" modules are stable and ready for deployment. All defined testing objectives have been achieved.

# 12. Definitions, Acronyms, and Abbreviations (Diksha Korat)

* **qTest:** A test management tool used for test planning, execution, defect tracking, and reporting.
* **JIRA:** A project management and issue tracking software for managing tasks, bugs, and workflow customization.
* **UAT (User Acceptance Testing):** Validation phase where end-users confirm the application meets requirements before deployment.
* **URL (Uniform Resource Locator):** Web address used for accessing and testing application modules.
* **Defects:** Any deviation from expected behavior identified during testing.
* **Action Plan: Steps for addressing and resolving identified issues during testing.**
* **Test Environment:** Simulated setups for hardware, software, and network configurations used during testing.