CS696F – Computer Science, Spring 2025

Pace University

A blue and yellow logo

AI-generated content may be incorrect.

TEST PLAN DOCUMENT

Xero Accounting Software Testing

Authors: Team 1

Team Manager: Misha Patel

Date: 4/3/2025

TEST PLAN DOCUMENT

Contents

[Test Plan Identifier 3](#_Toc194603325)

[Introduction (Misha) 3](#_Toc194603326)

[Test Items (Misha) 3](#_Toc194603327)

[Software Risk Issues (Misha) 4](#_Toc194603328)

[Features to be Tested (Shahrukh) 7](#_Toc194603329)

[Feature Not To Be Tested (Shahrukh) 12](#_Toc194603330)

[Item Pass/Fail Criteria (Shahrukh) 12](#_Toc194603331)

[Suspension Criteria And Resumption Requirements (Shahrukh) 13](#_Toc194603332)

[Suspension Criteria 13](#_Toc194603333)

[Resumption Requirements 14](#_Toc194603334)

[Test Deliverables (Anvika) 15](#_Toc194603335)

[Remaining Test Tasks (Anvika) 16](#_Toc194603336)

[Environmental Needs (Anvika) 16](#_Toc194603337)

[Staffing and Training Needs (Anvika) 17](#_Toc194603338)

[Responsibilities (Mayank) 17](#_Toc194603339)

[Schedule (Mayank) 18](#_Toc194603340)

[Risks and Contingencies (Mayank) 18](#_Toc194603341)

[Approvals (Mayank) 18](#_Toc194603342)

# Test Plan Identifier

Xero-ManualTesting-Team1\_TestPlan\_v1.0

# Introduction (Misha)

This test plan provides a detailed strategy for validating four critical modules of the Xero accounting software:

1. 04. Financial Data Entry
2. 05. Bank Feeds and Reconciliations
3. 06. Invoicing
4. 07. Accounts Receivable

The goal is to ensure these modules operate accurately, securely, and in alignment with accounting standards, while addressing cross-cutting concerns such as user permissions, data validation, and system integrations.

# Test Items (Misha)

* Financial Data Entry Module Test Items:
* Transaction Entry: Testing of various transaction types such as sales invoices, purchase invoices, receipts, and payments.
* Invoice Management: Testing of invoice creation, editing, deletion, and processing workflows.
* Bill Management: Testing of bill creation, editing, deletion, and payment workflows.
* Expense Management: Testing of expense entry, categorization, and reimbursement processes.
* Journal Entries: Testing of manual journal entries for adjusting transactions or account balances.
* Tax Handling: Testing of tax calculations, deductions, and compliance with tax regulations.
* Multi-currency Support: Testing of currency conversion and exchange rate accuracy for transactions in multiple currencies.
* Data Validation: Testing of data validation rules to ensure the accuracy and integrity of financial data entered into Xero.
* Bank Feeds and Reconciliations Module Test Items:
  + Bank Feeds Integration: Testing of automated import and synchronization of bank transactions from external institutions.
  + Transaction Matching: Testing of automated matching algorithms to reconcile bank transactions with Xero entries.
  + Manual Reconciliation: Testing of processes to resolve unmatched transactions or discrepancies.
  + Statement Import: Testing of importing bank statements (CSV/Excel) for reconciliation.
  + Reconciliation Reports: Testing of reports to verify accuracy and completeness of reconciled transactions.
  + Audit Trail: Testing of reconciliation history and status tracking (e.g., "Reconciled," "Unreconciled").
  + Bank Account Management: Testing of adding, updating, or deleting bank accounts in the chart of accounts.
* Invoicing Module Test Items:
  + Invoice Creation: Testing of invoice customization (logos, payment terms, tax rates).
  + Payment Processing: Testing of integration with payment gateways (e.g., Stripe, PayPal).
  + Multi-Currency Invoicing: Testing of currency conversion accuracy for international invoices.
  + Bulk Invoicing: Testing of generating and sending invoices in bulk.
  + Automated Reminders: Testing of scheduled reminders for unpaid invoices.
  + Draft Approval Workflows: Testing of role-based approval processes for draft invoices.
  + Invoice Status Tracking: Testing of lifecycle states (e.g., "Draft," "Sent," "Paid").
* Accounts Receivable Module Test Items:
* Payment Application: Testing of applying partial/full payments to invoices.
* Credit Limit Management: Testing of enforcing customer credit limits.
* Refund Processing: Testing of issuing and reconciling customer refunds.
* Aging Reports: Testing of accuracy in accounts receivable aging reports.
* Quote-to-Invoice Conversion: Testing of converting quotes into invoices.
* Reconciliation with Bank Feeds: Testing of matching receivable payments to bank transactions.
* Customer Communication: Testing of automated payment reminders and statements.

# Software Risk Issues (Misha)

Below are potential risks identified for the Xero Financial Data Entry, Bank Feeds, Invoicing, and Accounts Receivable modules, along with mitigation strategies. Risks are derived from the test scope, cross-cutting concerns, and feature backlogs.

1. Data Integrity Risks

Risk: Invalid or corrupted financial data due to improper field validation (e.g., negative invoice amounts, incorrect tax rates).

Impact: Inaccurate financial reports, compliance violations.

Mitigation:

Enforce FV-Field Validation rules (e.g., numeric-only fields, mandatory tax codes).

Test edge cases (e.g., $0 invoices, multi-currency decimals).

2. Security and Access Control Risks

Risk: Unauthorized access to sensitive data (e.g., bank feeds, customer credit limits) due to misconfigured user roles.

Impact: Data breaches, financial fraud.

Mitigation:

Validate ET-Entitlements (e.g., Read Only users cannot delete transactions).

Test role-based workflows (e.g., "Invoice Only" users restricted from approving drafts).

3. Integration Risks

Risk: Bank feed imports (e.g., CSV/Excel) fail or sync incorrectly with Xero transactions.

Impact: Reconciliation errors, mismatched balances.

Mitigation:

Test SI-In/SI-Out integrations with sample bank statements.

Validate automated transaction matching (5.1 Perform Bank Reconciliation).

4. Compliance Risks

Risk: Tax calculations (e.g., GST, VAT) in invoices or journals deviate from regulatory standards.

Impact: Legal penalties, audit failures.

Mitigation:

Audit CL-Calculations logic (e.g., tax rounding, multi-currency adjustments).

Cross-check tax reports against manual computations.

5. Performance Risks

Risk: System slowdowns during bulk operations (e.g., 6.6 Generate Bulk Invoices, 5.9 Import Data).

Impact: User frustration, timeouts during critical tasks.

Mitigation:

Load-test bulk operations with 500+ invoices/transactions.

Monitor API response times during CN-Connectivity validations.

6. Concurrency Risks

Risk: Data conflicts when multiple users edit the same transaction (e.g., 4.4 Edit Posted Entries, 5.3 Manage Transactions).

Impact: Overwritten changes, data loss.

Mitigation:

Test CC-Concurrency scenarios (e.g., two users editing a journal entry).

Implement lock mechanisms or version tracking.

7. Dependency Validation Risks

Risk: Deleting critical data (e.g., 4.10 Delete Journal Entries) linked to reconciled transactions.

Impact: Broken audit trails, reconciliation failures.

Mitigation:

Enforce DDV-Data Dependency Validation (e.g., block deletion of journals tied to bank feeds).

8. Training and Knowledge Gaps

Risk: Testers lack accounting expertise, leading to missed defects (e.g., incorrect GL account coding).

Impact: Undetected errors in financial statements.

Mitigation:

Provide training using YouTube tutorials from RCT (e.g., tax handling, reconciliation workflows).

Assign SME (Subject Matter Expert) reviewers for complex features.

9. Environmental Risks

Risk: Demo Company resets during testing, erasing manually added data (e.g., 7.12 Create Quotes).

Impact: Loss of test progress, duplicated efforts.

Mitigation:

Schedule tests around the 28-day reset cycle.

Export/backup test data before critical validations.

10. Third-Party Payment Gateway Risks

Risk: Payment processing failures (e.g., 6.3 Process Payments) due to Stripe/PayPal API downtime.

Impact: Unprocessed invoices, revenue loss.

Mitigation:

Test with sandbox environments for payment gateways.

Implement fallback payment methods.

# Features to be Tested (Shahrukh)

A. Financial Data Entry Module

4.1 Enter Financial Transactions Manually

* Validate manual entry of invoices, bills, receipts, and payments.
* Cross-Cutting Concern: FV-Field Validation, ET-Entitlements.

4.2 Handle Currency Conversions

* Test exchange rate accuracy and multi-currency transaction compliance.
* Cross-Cutting Concern: CL-Calculations, AT-Account Type.

4.3 Save and Review Draft Entries

* Validate draft saving, editing, and approval workflows.
* Cross-Cutting Concern: STS-Status, ET-Entitlements.

4.4 Edit Posted Entries

* Test modifications to finalized transactions.
* Cross-Cutting Concern: CC-Concurrency, FV-Field Validation.

4.5 Delete Posted Entries

* Validate deletion permissions and audit trail updates.
* Cross-Cutting Concern: ET-Entitlements, DDV-Data Dependency Validation.

4.6 Generate Data Entry Reports

* Test accuracy of transaction summaries and exports.
* Cross-Cutting Concern: DF-Out, CL-Calculations.

4.7 Import Financial Data from External Sources

* Validate CSV/Excel imports (e.g., bank feeds, third-party data).
* Cross-Cutting Concern: SI-In, CN-Connectivity.

4.8 Create Journal Entries

* Test manual journal creation for adjustments.
* Cross-Cutting Concern: CL-Calculations, FV-Field Validation.

4.9 Update Journal Entries

* Validate edits to existing journals.
* Cross-Cutting Concern: STS-Status, CC-Concurrency.

4.10 Delete Journal Entries

* Test deletion permissions and dependency checks.
* Cross-Cutting Concern: ET-Entitlements, DDV-Data Dependency Validation.

B. Bank Feeds and Reconciliations Module

5.1 Perform Bank Reconciliation

* Validate automated and manual reconciliation workflows.
* Cross-Cutting Concern: CL-Calculations, STS-Status.

5.2 Manage Bank Account

* Test adding/updating bank accounts in Xero.
* Cross-Cutting Concern: ET-Entitlements, ST-PY Payment Settings.

5.3 Manage Transactions

* Validate manual adjustments for unmatched entries.
* Cross-Cutting Concern: CC-Concurrency, FV-Field Validation.

5.4 Create Rule Template

* Test rule creation for recurring transaction handling.
* Cross-Cutting Concern: DDD-Data-Driven Defaults.

5.5 Manage Rules

* Validate enabling/disabling reconciliation rules.
* Cross-Cutting Concern: CC-Concurrency, STS-Status.

5.7 Check Audit Trail

* Validate visibility of reconciliation history.
* Cross-Cutting Concern: STS-Status, DF-Out.

5.8 View Transaction History

* Test filtering and retrieval of past transactions.
* Cross-Cutting Concern: STS-Status, DF-Out.

5.9 Import Data

* Validate bulk imports of bank statements.
* Cross-Cutting Concern: SI-In, CN-Connectivity.

5.10 Create Chart of Accounts

* Test adding new accounts (e.g., expense, asset).
* Cross-Cutting Concern: AT-Account Type, FV-Field Validation.

5.11 Update Chart of Accounts

* Validate edits to existing accounts.
* Cross-Cutting Concern: DDV-Data Dependency Validation.

5.12 Delete Chart of Accounts

* Test archiving/deleting unused accounts.
* Cross-Cutting Concern: ET-Entitlements, DDV-Data Dependency Validation.

C. Invoicing Module

6.1 Create Invoices

* Validate invoice customization (logos, payment terms).
* Cross-Cutting Concern: ST-Inv Settings, FV-Field Validation.

6.2 Customize Invoices

* Test template designs and branding options.
* Cross-Cutting Concern: ST-Inv Settings, ET-Entitlements.

6.3 Process Payments and Integrate with Payment Systems

* Validate Stripe/PayPal integrations.
* Cross-Cutting Concern: SI-Out, CL-Calculations.

6.4 Manage Customers

* Test customer profile creation and updates.
* Cross-Cutting Concern: FV-Field Validation, ET-Entitlements.

6.5 Handle Multi-Currency Invoicing

* Validate currency conversion accuracy.
* Cross-Cutting Concern: CL-Calculations, AT-Account Type.

6.6 Generate Bulk Invoices

* Test bulk creation and distribution workflows.
* Cross-Cutting Concern: DF-Out, CC-Concurrency.

6.7 Generate Reports and Analytics

* Validate financial summaries (e.g., sales, taxes).
* Cross-Cutting Concern: CL-Calculations, DF-Out.

6.9 Schedule Invoices

* Validate automated invoice scheduling.
* Cross-Cutting Concern: DDD-Data-Driven Defaults.

6.10 View Audit Trail

* Test visibility of invoice lifecycle changes.
* Cross-Cutting Concern: STS-Status.

6.11 Send Automated Invoice Reminders

* Validate email/SMS reminder workflows.
* Cross-Cutting Concern: ST-EM Email Settings.

6.12 Create Items

* Test adding products/services to invoices.
* Cross-Cutting Concern: FV-Field Validation.

6.13 Update Items

* Validate edits to item details (e.g., pricing).
* Cross-Cutting Concern: ET-Entitlements.

6.14 Delete Items

* Test removal of unused items.
* Cross-Cutting Concern: DDV-Data Dependency Validation.

6.15 Approve Draft

* Validate role-based approval workflows.
* Cross-Cutting Concern: ET-Entitlements.

6.16 Delete Draft

* Test draft deletion permissions.
* Cross-Cutting Concern: ET-Entitlements.

6.17 Deposit Awaiting Payment

* Validate tracking of pending payments.
* Cross-Cutting Concern: STS-Status.

6.18 Search (Generic)

* Test search functionality across invoices.
* Cross-Cutting Concern: DF-Out.

D. Accounts Receivable Module

7.1 Send Invoices to Customers

* Validate email/SMS delivery of invoices.
* Cross-Cutting Concern: ST-EM Email Settings.

7.2 Schedule Recurring Invoices

* Test setup of recurring billing cycles.
* Cross-Cutting Concern: DDD-Data-Driven Defaults.

7.3 Track Invoice Status

* Validate real-time status updates (e.g., "Paid," "Overdue").
* Cross-Cutting Concern: STS-Status.

7.4 Send Payment Reminders

* Test automated reminders for overdue invoices.
* Cross-Cutting Concern: ST-EM Email Settings.

7.5 Apply Payments to Invoices

* Validate partial/full payment allocations.
* Cross-Cutting Concern: CL-Calculations, STS-Status.

7.6 Process Customer Refunds

* Test refund workflows and bank reconciliation.
* Cross-Cutting Concern: CL-Calculations, CN-Connectivity.

7.7 Generate Accounts Receivable Reports

* Validate aging reports and credit summaries.
* Cross-Cutting Concern: DF-Out, CL-Calculations.

7.8 Manage Customer Credit Limits

* Test enforcement and adjustment of credit limits.
* Cross-Cutting Concern: DDV-Data Dependency Validation.

7.9 Reconcile Payments with Bank Transactions

* Validate matching payments to bank feeds.
* Cross-Cutting Concern: CL-Calculations, CN-Connectivity.

7.12 Create Quotes

* Test quote-to-invoice conversion workflows.
* Cross-Cutting Concern: ST-Inv Settings.

7.13 Update Quotes

* Validate modifications to draft quotes.
* Cross-Cutting Concern: ET-Entitlements.

Cross-Cutting Concerns Legend

* ET-Entitlements: User role permissions (Admin, Read Only, etc.).
* FV-Field Validation: Data integrity checks (numeric, date formats).
* CL-Calculations: Tax, currency, and financial computations.
* STS-Status: Lifecycle tracking (e.g., "Draft" → "Paid").
* CN-Connectivity: Integration with external systems.
* SI-In/SI-Out: Third-party data imports/exports.
* DDD-Data-Driven Defaults: Context-based default values.
* DDV-Data Dependency Validation: Dependency checks for deletions/updates.

# Feature Not To Be Tested (Shahrukh)

1. 5.6 Manage Taxes

* Reason: Tax compliance and advanced tax handling (e.g., region-specific regulations) are excluded, as they are assumed to be pre-validated by Xero’s internal processes.

2. 6.8 Set Role-based Permissions for Management

* Reason: Role-based access is covered under the cross-cutting concern ET-Entitlements (e.g., Admin vs. Read Only permissions). Detailed testing of permission configurations is out of scope

3. 7.10 Create Contacts and 7.11 Update Contacts

* Reason: Contact management is excluded as it is considered part of 6.4 Manage Customers. Testing focuses on customer workflows, not general contact operations

Rationale for Exclusion

* Scope Alignment: Focus remains on core workflows (e.g., bank reconciliation, invoice creation) and cross-cutting concerns (e.g., user roles, field validation).
* Resource Constraints: Prioritization of features with higher business impact (e.g., 5.3 Manage Transactions, 6.3 Process Payments).
* Assumptions: Tax logic and sub-account dependencies are assumed to be validated by Xero’s standard compliance frameworks.

# Item Pass/Fail Criteria (Shahrukh)

Pass Criteria:

* All system's core functionality must function as expected, as outlined in individual test cases.
* No critical defects (e.g., payment failures, data loss, security breaches) may be present.
* An end user must complete a purchase cycle (create invoice → process payment → confirm receipt) and initiate a refund without errors.
* At least 90% of all test cases must pass.
* No failed test cases may impact the end-user’s ability to perform essential tasks (e.g., creating invoices, reconciling payments).

Fail Criteria:

* Critical defects are identified (e.g., inability to process payments, reconcile transactions).
* <90% of test cases pass.
* Failed test cases block core workflows (e.g., user cannot edit invoices, delete drafts, or view audit trails).
* End users encounter errors during purchase cycles or refund initiation.

# Suspension Criteria And Resumption Requirements (Shahrukh)

## Suspension Criteria

Testing will be paused immediately if any of the following occur:

1. Critical Defects in Core Functionality:
   * Failure in basic CRUD operations (e.g., inability to create invoices [6.1], delete journal entries [4.10], or reconcile transactions [5.1]).
   * Payment processing failures (e.g., 6.3 Process Payments) or data corruption during imports (4.7).
2. Security/Authorization Breaches:
   * Unauthorized access to sensitive features (e.g., Admin-only actions like 5.12 Delete Chart of Accounts).
   * Misconfigured user roles (ET-Entitlements) allowing restricted actions (e.g., Read Only users editing invoices).
3. Environmental Instability:
   * Xero Demo Company resets unexpectedly, erasing test data (e.g., 7.9 Reconcile Payments).
   * Third-party integrations (e.g., Stripe/PayPal) experience downtime, blocking payment testing (6.3).
4. Critical Calculation Errors:
   * Incorrect tax computations (CL-Calculations) in invoices (6.5) or journals (4.8).
   * Currency conversion inaccuracies (4.2) impacting financial reports.
5. Resource Unavailability:
   * Testers lack access to necessary tools (e.g., qTest, JIRA) or training on accounting workflows.

## Resumption Requirements

Testing will resume only after the following steps are completed:

1. Defect Resolution:
   * Critical defects are fixed, retested, and verified (e.g., reconciliation errors [5.1] resolved and validated).
   * Code fixes are deployed to the test environment, and regression testing confirms no new issues.
2. Security/Authorization Compliance:
   * Role permissions (ET-Entitlements) are reconfigured and validated (e.g., Read Only users cannot delete drafts [6.16]).
   * Security gaps (e.g., unauthorized access) are patched and audited.
3. Environmental Stability:
   * Demo Company is restored, and test data (e.g., invoices, bank feeds) is re-imported.
   * Third-party integrations (e.g., Stripe) are confirmed operational via sandbox testing.
4. Calculation Validation:
   * Tax and currency logic (CL-Calculations) is reviewed and verified with accounting SMEs.
   * Reports (4.6, 6.7) are cross-checked against manual calculations.
5. Resource Readiness:
   * Testers complete training on high-risk features (e.g., 7.8 Manage Customer Credit Limits).
   * QA Lead approves the test environment and tool access.
6. Approvals:
   * QA Lead and Project Stakeholders sign off on defect fixes and environment stability.
   * Updated test plans are distributed to the team.

# Test Deliverables (Anvika)

1. Planning & Design Documents

1. Functional Decomposition Diagram (FDD): Visual breakdown of Xero modules (e.g., Financial Data Entry, Bank Reconciliation).

2. Context Diagram: Interactions between Xero and external systems (e.g., banks, payment gateways).

3. Dataflow Diagrams (DFD): Data flow across modules and integrations.

4. Requirements Composition Table (RCT): Maps features (e.g., 4.1, 6.3) to cross-cutting concerns (e.g., ET-Entitlements).

5. Entitlements Specification: Defines user roles (Admin, Read Only) and permissions.

6. User Stories with Acceptance Criteria: Example: *"As a Sales user, I can create invoices to ensure timely payments."*

7. Testing RoadMap: Timeline and scope for test cycles (April 25, May 02, May 09).

8. Test Plan Document: This document, approved by QA Manager and stakeholders.

2. Test Execution Artifacts

1. Test Design Specifications (Cycle 1, 2):

Detailed test cases for each cycle (e.g., 4.1–4.5 in Cycle 1).

2. Test Case Specifications (Cycle 1, 2):

Step-by-step instructions (e.g., "Validate 6.3 Payment Integration").

3. Test Execution Logs:

Test Run Results: Saved in qTest, including pass/fail status, defect links, screenshots, and tester notes.

Defect Log: Tracked in JIRA with severity (Critical/Major/Minor), steps to reproduce, and resolution status.

3. Reporting & Closure

1. Defect Reports: JIRA exports with all defects, resolutions, and retest outcomes.

2. Test Summary Report:

Cycle-wise metrics (pass rate, defect density).

Key risks (e.g., bank feed delays, role misconfigurations).

Recommendations for future testing.

3. Final Presentation (PowerPoint): Summary of testing outcomes, risks, and lessons learned.

4. Final Project Presentation in Class, App Demo (Recorded): Live/recorded demo of tested features (e.g., invoice creation, reconciliation).

# Remaining Test Tasks (Anvika)

1. Test Plan Preparation:
   * Finalize scope, test cycles, and cross-cutting concerns (e.g., ET-Entitlements, CL-Calculations).
2. Test Design & Case Specifications:
   * Write 200+ test cases in qTest (e.g., 4.1 Enter Transactions, 6.3 Payment Integrations).
3. Environment Readiness:
   * Populate Xero Demo Company with:
     + 50+ invoices, 100+ bank transactions.
     + Multi-currency data (USD, EUR, GBP).
     + User roles (Admin, Sales, Read Only).
4. Test Execution:
   * Complete 3 test cycles (April 25, May 02, May 09) with 95%+ pass rate.
5. Test Summary Report:
   * Highlight defect trends, cycle velocity, and unresolved risks.

# Environmental Needs (Anvika)

* Platform: Xero Demo Company ([Link](https://central.xero.com/s/article/Use-the-demo-company)).
* Test Data Requirements:
  + Products: 20+ items priced between 10–10–5,000.
  + Transactions: 50+ invoices, 30+ bills, 100+ bank transactions.
  + Multi-Currency: USD, EUR, GBP entries with live exchange rates.
  + User Roles: Admin (full access), Sales (invoicing), Read Only (view-only).
* Tools:
  + qTest for test management.
  + JIRA for defect tracking.
  + Stripe/PayPal sandbox accounts for payment testing.

# Staffing and Training Needs (Anvika)

* Team Composition:
  + Testers (4): Execute test cases across modules (e.g., 4.1–7.13).
  + Business Analysts (2): Validate tax logic, reconciliation workflows.
  + QA Manager (1): Oversee cycles, approvals, and risk communication.
* Training:
  + Mandatory:
    - Xero basics (invoice creation, bank reconciliation).
    - qTest/JIRA onboarding.
  + Optional:
    - Advanced tax handling (CL-Calculations).

# Responsibilities (Mayank)

| **Role** | **Responsibilities** |
| --- | --- |
| QA Manager | - Approve test plans, reports, and defect resolutions.  - Oversee test cycles and deadlines.  - Communicate risks to stakeholders.  - Validate final sign-off.  - Manage team resources. |
| QA Analyst | - Design and update test cases in qTest.  - Execute tests (e.g., 4.1–7.13) and log defects in JIRA.  - Track defect resolution and retest fixes.  - Analyze test metrics (pass rate, defect density).  - Validate cross-cutting concerns (e.g., FV-Field Validation). |
| Business Analyst | - Define business requirements for test cases.  - Verify alignment with accounting standards.  - Assist in UAT for critical features.  - Review financial reports (e.g., 5.19, 6.7). |
| Software Tester | - Execute assigned test cases (e.g., 6.3 Process Payments).  - Log defects with steps and screenshots.  - Retest resolved bugs within 24 hours.  - Populate test data (invoices, bank feeds).  - Collaborate on complex workflows (e.g., 7.9 Reconcile Payments). |

# Schedule (Mayank)

| **Phase** | **Dates** | **Focus** | **Features** | **Story Points** |
| --- | --- | --- | --- | --- |
| **Cycle 1** | Apr 18–25 | Core CRUD | 4.1–4.5, 6.1–6.4 | 49 SP |
| **Cycle 2** | Apr 26–May 02 | Integrations | 5.1–5.12, 7.5–7.9 | 68 SP |
| **Cycle 3** | May 03–09 | High-Effort Tasks | 6.3, 6.5–6.17 | 60 SP |

# Risks and Contingencies (Mayank)

| **Risk** | **Impact** | **Mitigation** |
| --- | --- | --- |
| Cycle 1 delays due to CRUD failures | Blocks Cycle 2 | Extend Cycle 1 by 2 days; reallocate testers. |
| Testers lack accounting knowledge | Inaccurate results | Assign BAs as mentors; use Xero tutorials. |
| Bank feed downtime | Halts reconciliation testing | Use pre-imported CSV data for manual matching. |
| Payment gateway failures | Blocks invoicing tests | Switch to sandbox environments for Stripe/PayPal. |

# Approvals (Mayank)

Approval to Proceed:

* QA Manager: Validates test completion, defect resolution, and report accuracy.
* Professor/Stakeholder: Confirms alignment with academic/business objectives