PUPT Faculty

Academic Requirements

Management System

SOFTWARE TEST PLAN

Version 1.0

Prepared by:

FARM Development Team

Fidel, Diana Rose V.

Nabayra, James V.

Mingo, Ed Judah E.

Villamarzo, Kazel S.

August 1, 2024

Table of Contents

REVISION HISTORY

Version Number	Revision Date	Summary of Changes
V.0.1	August 2024	Initial Specifications on Modules
V.0.2	September 18, 2024	Submission of Accomplishment Report via pdf
V.0.3	October 13, 2024	Editing of Daily Time in with ticket
V.0.4	November 8, 2024	Revisions on Time in, Bi-weekly submissions

1 Introduction

This Software Test Plan outlines the process for testing the Faculty Academic Requirements Management System (FARM), designed to streamline the management of academic documents at PUP - Taguig. FARM centralizes faculty-related documentation, automating submission, tracking, and updates to enhance efficiency and reduce errors. By improving the workflow, FARM ensures better resource utilization and secure handling of academic records, ultimately supporting the university's academic management processes.

2 Test Items

This activity focuses on the following:

1. Log-in Module
2. Dashboard
3. Audit Trail Module
4. Request Upload Access
5. Archive Module
6. Accomplishment Module
7. Maintenance Module

Table 1: List of items to test.

ITEM TO TEST	TEST DESCRIPTION	RESPONSIBILITY
Login Module		
Entering Valid Credentials	Test the login functionality using HRIS account for both roles.	Included the temporary functions in the email for user account creation

Dashboard		
Faculty and Director Dashboard Functional Testing	The dashboard provides an overview of academic activities, including submission statuses, storage usage, and pending reviews. The test will validate that the data displayed in the dashboard is accurate, up-to-date, and represented clearly across all elements. Total of Approved: Displays the number of approved submissions. The test ensures accuracy and real-time updates as submissions are approved. Total Pending Review: Shows the number of items pending review. The test checks for correctness and real-time updates when reviews are completed or initiated. Storage Used: Indicates	Ensure that the "Total of Approved" count updates correctly with approvals, "Total Pending Review" adjusts with new or completed reviews, "Storage Used" reflects real-time changes, "Submitted Status per Folder" chart displays accurate statuses with correct colors, and "Storage Usage" pie chart shows precise proportions of used vs. available storage.

	current storage usage. The test verifies accurate calculation and updates when files are added or removed. Submitted Status per Folder: Graph showing submission statuses (Approved, Declined, To Review). The test ensures colors and statuses accurately reflect the folder's current state. Storage Usage (Pie Chart): Visual of used vs available storage. The test confirms the chart proportions and labels are accurate.	
Admin Dashboard Functional Testing	The test items for the Admin Dashboard, including specific features and their associated responsibilities, ensuring that all dashboard functionalities are thoroughly tested. Generate Reports Verify the functionality that allows admins to generate reports on faculty submissions and their statuses. Total Faculty Users Confirm that the dashboard displays the correct total number of faculty users registered in the system. Total of Files Submitted Test the feature that counts and displays the total number of files submitted by faculty members. Total of Pending Review Check that the system correctly shows the number of submissions currently pending review by admins.	Ensure reports are accurately generated and contain all necessary data. Monitor and verify the number of registered faculty users. Track the total number of submitted files for all academic requirements. Review and manage the pending submissions effectively.

	Completed Reviews Ensure that the dashboard shows the accurate total of reviews that have been completed by admins.	Track and confirm the total number of completed reviews.
	Status Rates Verify that the status rates (e.g., completed, pending, reviewed) are calculated and displayed correctly.	Ensure accurate calculation and display of submission statuses.
	Submitted Files per Requirement Test if the dashboard displays the correct number of files submitted for each academic requirement.	Track the submission count for each requirement and ensure data accuracy.
	Submitted Status per Requirement Ensure that the dashboard accurately reflects the submission status (e.g., submitted, in-progress) for each requirement.	Monitor the submission status per requirement to keep track of faculty progress.

Audit Trail Module		
Audit trail	The audit trail is responsible for tracking user activities in the system to maintain a secure and transparent record of actions. It serves as a critical component for system monitoring, helping administrators to detect any unauthorized or suspicious activities. This feature contributes to data integrity, compliance, and accountability by providing a clear log of user interactions within the system.	Includes temporary logging functions to capture user actions (such as login, logout, and other relevant activities) associated with email addresses during user account creation and general system access. This functionality ensures that each action is documented in real-time, contributing to user activity tracking and security monitoring.

Request Upload Access		
Request Upload Access	The "Request Upload Access" feature allows faculty members to submit requests for permission to upload documents, specifying their reason for the request. This helps in managing and regulating document submissions by providing controlled access.	Includes a temporary function for faculty members to request upload permissions, capturing details like date, time, faculty name, and reason. This ensures a controlled document submission process by validating and tracking upload access requests.

Archive Module		
Archive File	Ensure that academic records and submissions are properly archived and can be retrieved as needed.	Manage archival and retrieval of past records.
Selecting Archive	Verify that the system allows the admin to archive a folder by selecting it from the list. Ensure that the folder is removed from the active list and moved to an archive section without being deleted, preserving the data for future reference.	Confirm that the admin can restore an archived folder back to the active list when needed. Ensure that once restored, the folder reappears in its original category.

Maintenance Module		
Maintenance (Manage Main Requirements)	Verify that the system allows the admin to create a new folder under a specific category (like "Classroom Management" or "Test Administration"). Ensure that when the "Add Folder" button is clicked, the user is prompted to input the folder name and choose the main folder category, and then successfully save the folder.	The admin is responsible for organizing and managing the folders related to different academic requirements. They must ensure that the folders are named clearly and placed under the correct main category (e.g., Classroom Management or Test Administration) to maintain an organized system.
Edit Folder (Manage Main Requirements	Ensure that the admin can edit the name or category of an existing folder. When the "Edit" button is clicked, the system should allow updates to the folder's details and save them.	The admin must take responsibility for ensuring that any edits made to a folder (e.g., changing the name or category) are accurate and necessary. This includes ensuring that the changes reflect the correct data and structure within the system to prevent confusion or mismanagement of academic records.
Delete Folder (Manage Main Requirements)	Confirm that the admin can delete a folder by clicking the "Delete" button next to it. The system should ask for confirmation before removing the folder.	The admin has the responsibility to ensure that folders are only deleted when absolutely necessary. Deleting a folder can result in the permanent loss of data or organization, so the admin must confirm that the folder and its contents are no longer needed or that the information has been safely archived elsewhere.
Maintenance (Manage Announcements)	Verify that the system successfully sends email reminders to the selected	The admin has the responsibility to ensure that the email reminders

	faculty members regarding their pending academic requirements. Ensure that the "To," "Subject," and "Body" fields are properly filled, and the message is delivered without errors.	are accurate, clear, and sent to the correct faculty members. The admin should verify that the message contains all necessary information, such as deadlines or submission instructions, and that the recipients list is accurate.
Viewing Announcements	Verify that faculty members can view the announcements posted by the admin on their dashboard. Ensure that the announcements are displayed clearly, with details such as the subject, date, and recipient list visible.	Confirm that the faculty member receives an email notification when an announcement is sent by the admin. The email should include the subject, sender, and message content, reflecting the announcement posted on the platform.
Maintenance (Upload Schedule)	Test the functionality to determine if it's possible to set a deadline schedule for requirements.	The deadline scheduling functionality for requirements.

Accomplishment Module		
Upload the Academic Requirements	Test the ability for faculty to submit academic requirements they need to submit	Submit requirements and track progress.
Approve	Verify that when the requirements meet the specified criteria, the admin can successfully approve the submission. The status of the request should be updated to "Approved," and the submitter should be notified of the approval.	The admin has the responsibility to carefully evaluate the submitted requirements before approving them.

The Faculty Academic Requirements Management System (FARM) integrates security and compatibility, but several factors may affect its functionality, which can hinder efficiency and impact system processes:

A. Device Specifications and Software Compatibility:
The system’s performance may vary based on the user's device and computer specifications. Older devices and outdated software can affect system compatibility and speed.

B. Browser Dependency:
The system is optimized for use with Google Chrome. Using other browsers may lead to unexpected errors and slow response times.

C. Internet Connectivity:
The system’s responsiveness is heavily reliant on a stable internet connection. Users with poor connection may experience delays and interruptions, affecting their interaction with the system.

D. Data Backup and Security:
The current system lacks an external backup for its database. Integrating cloud storage solutions would help prevent data loss, safeguarding essential academic records.

E. User Authentication and Data Protection:
The system ensures security through encrypted login credentials, accessible only to authorized personnel based on their assigned roles.

4 Features to be tested

The following areas will be tested to ensure the system's reliability:

A. Account access for different user roles
B. Faculty document submission and tracking
C. Report generation for academic requirements
D. Validation and approval of submissions by admin

5 Features not to be tested

The following is a list of the areas that will not be specifically addressed. All testing

in these areas will be indirect as a result of other testing efforts.

A. Security testing against malwares would not be included

B. Distributing reports to non-essential roles not required to review and receive alerts.

C. Alteration of document(s)

D. Crash testing and network feasibility testing due to inclement weather and network connectivity issues.

6 Test Approach

The developmental approach will use the Agile methodology, an iterative method where testing phases will be divided dynamically. The phases include Planning, Design, Development, Testing, Deployment, and Review. These phases will be completed throughout the duration of the study, with the development team and stakeholders continuously involved in the system's improvement. Business requirements and rules are provided based on user experience and the production backlog. The requirements are prioritized in the sprint backlog for record-keeping and maintenance.

The product backlog and user testing are documented during each sprint phase, where the design and development are integrated into the Agile methodology. The dynamic nature of Agile allows the development team to assess completed and incomplete modules for fault testing and quality assurance. The planning phase updates introduced modules and incorporates client requirements. The design and development phases ensure updates enhance user experience, addressing discrepancies and errors that arise during testing.

The development team continuously updates and fixes bugs for quality testing while adhering to business requirements. Agile provides a steady pace, allowing the team to deploy the system and ensure safeguards such as privacy and security are upheld. During sprint planning and design phases, the team records testimonials from current faculty members regarding the old submission process for academic requirements.

In the first phase, the development team will complete the following:

- Record testimonials from faculty members to establish a baseline for system development, which will inform the sprint backlog for system integration. Case reviews for functionality that improve the old process will be included for effort estimation. Testimonials will prioritize the sprint backlog.
- Decide the number of hours for each module, including testing and presentation to the client. This will ensure that the system is reviewed with recommendations that align with client needs.

During the sprint, the development team will:

- Test each module upon completion.
- Perform security, back-end, and usability testing for non-functional areas.
- Conduct user testing after sprint modules are completed.
- Conduct cross-testing among developers after two or more modules are completed in a sprint.
- Refer to test case documents and follow standard formats set by the academe for module completion.
- Use User Acceptance Testing (UAT) after completing all modules, with permission from the client and compliance with the university's ethics committee. It should be explained to users the importance of UAT for the integrity and completion of the system, ensuring that acceptance criteria are coordinated between the client and developers.
- Mark failed test cases in red with reasons provided for the failure. Test cases for revision will be marked in yellow, while passed cases will be marked in green.
- Log bugs and errors during the testing period for analysis and functionality testing, and constantly communicate with the user to verify if issues persist.

In the sprint review, the team will:

- Evaluate testing activities, and the stakeholders may provide feedback.
- Discuss incomplete tasks, functionality, and security concerns, and include necessary fixes in the next sprint.

In the sprint retrospective, the team will:

- Discuss improvements and concerns regarding process, functionality, tools, time constraints, and system integration for the next sprint. All adapted changes will be documented for the upcoming sprint.

7. Item Pass / Fail Criteria

All the functions of the system must work as intended. At least 95% of test cases should pass. No failed test cases should critically affect the system's processes or the user's ability to use the application. If any critical function fails, it must be resolved before moving to the next phase.

8. Suspension Criteria and Resumption Requirements

Testing may be suspended under the following conditions:

- A critical problem is identified that requires significant re-testing of the affected function.
- It is discovered that business or technical specifications require major modifications due to escalated test issues. These changes would necessitate additional test analysis or modification to the Detailed Test Plan.
- The testing environment is unavailable for any reason.
- Performance issues arise, where the environment suffers performance drops below 50% of its normal capacity, requiring substantial re-testing.

Resumption will only occur once these issues have been addressed and resolved, and the testing environment is restored to optimal conditions.

9. Test Deliverables

The following items will be delivered after testing is complete:

- Screenshots for Unit Testing
- Software Test Plan
- Software Test Report
- 100% Passed Test Cases
-

10. Remaining Test Tasks

The following tasks remain to be completed:

- Preparation of the test plan
- Creation of test cases
- Execution of the tests
- Reporting of bugs

11. Environmental Needs

The entire project team will be engaged in evaluating the system. Any proposed modifications by users due to errors during testing and development will be assessed, and environmental needs such as access to the testing platform, database, or servers will be ensured.

12. Staffing and Training Needs

The testing process will primarily be handled by the developers and stakeholders. If stakeholders are unavailable, at least one assigned tester must handle system acceptance testing phases. All involved personnel should receive proper training to understand the test cases, procedures, and tools.

13. Responsibilities

- The **Lead Tester** will oversee the testing process and manage the availability of resources and personnel.
- The **Development Team** will participate in reviewing the system and addressing change requests resulting from defects identified during testing.
- **Stakeholders** will verify test results and address any concerns or questions that arise during testing.
-

14. Schedule

The project plan includes allocated time for various testing activities, with specific dates and times outlined in the project timeline. The schedule covers:

- **A. Review of Requirements Document:** Ensuring all team members are aligned with the system requirements.
- **B. Development of Test Plan:** Allocating time for the creation of the test plan.
- **C. Unit Test Time:** Performing tests during the development process to ensure the system functions as expected.

The coordination of personnel for each task will be managed by the **Project Manager** in collaboration with the development and test team leaders.

15. Planning Risks and Contingencies

The testing process may be interrupted due to the following risks:

- **A. Undetected Issues:** Certain issues may arise that weren't identified during earlier phases of testing.
- **B. Unavailability of Key Personnel:** If necessary staff, including testers and developers, are unavailable during critical testing phases, delays may occur.
- **C. Testing Environment Failure:** If the test environment (such as servers or databases) becomes unavailable, the schedule may need adjustment.

Contingency Plan: If any of these risks materialize, the project manager will adjust the timeline, and alternative personnel may be assigned to mitigate the delay. Additionally, backup plans for testing environments (e.g., secondary servers) will be put in place to avoid interruptions.

16 Approvals

Project Manager / Developer	**Nabayra, James V.**
Lead Developer	**Mingo, Ed Judah E.**
Quality Assurance / Developer	**Fidel, Diana Rose V.**
Document Analyst/Developer	**Villamarzo, Kazel S.**

17 Glossary

Term	**Definition**
PUPT FARMS	Polytechnic University of the Philippines - Faculty Academic Requirements Management System
Stakeholders	Those involve in the project such as developers, users and client.

Academic Requirements	The necessary documents or tasks that faculty members need to complete, such as syllabi, classroom management plans, and test administration results.
Announcement Module	A feature that allows the system administrator to send out important updates, reminders, and announcements to faculty members.
Report Generation Module	A feature that creates summaries of faculty submissions and system activities for review by administrators and directors.