# Overall Test Strategy for PMO Projects

[View the Mind Map](https://www.xmind.net/m/2VtTK7)

The test strategy is a set of ideas that will guide the test planning throughout the project. It is the reasoning behind the testing and will be determined by the context of the project. It will also serve as a high-level overview of the test plan. We may update the test plan as we learn more about the project but will continue to share it with the team members to enhance the level of transparency. The Test Strategy, however, will not normally change.

## Determine the Testing Overview

A picture is very helpful here. Show the approximate number of sessions. Not all projects need (or want) the PMO to manage their testing efforts. We can help by sharing our processes, working together, or managing the whole testing process. That will be determined here.

Testing Sessions

* + How many?
	+ Internal, UAT, or both?

Who will manage the testing?

* + PMO
	+ UConn Department
	+ Vendor

## Determine Testing Types

The types of testing needed will be based on the context of the project. Here we will also determine who will be writing the test cases or if we can reuse older ones. Sometimes, venders or consultants will supply sample test cases or scripts that can be easily modified.

Scenarios/Stories

Integration

Functional

Verification

User Acceptance

Exploratory

Performance/Load

Documentation Testing

Regression

## Determine Testers

In most cases the testers will be the users or the SMEs of the system. If programmers are involved, they will most likely be doing their own unit testing before releasing the changes.

Who will be testing?

* + Staff
	+ Stakeholders
	+ Outside group

Notify Testers of dates and requirements.

Is test training needed?

## Determine the Test Plan Specifics (Deliverable)

The test plan will be created as soon as enough information is available to make the plan. The act of planning can reveal issues so this will be a dynamic document and will be stored in the **Files** tab of the **QA and Testing** channel in the project’s Team site.

Based on Strategy

* + Will there be multiple phases?

When will we test?

* + Create a Schedule

Where will we test?

* + Teams
	+ In a physical room?

Test Environment

* + Always test in a test environment
	+ How will we transfer to production?

Logging Defects

* + JIRA
	+ Spreadsheet
	+ Teams

## Setup Teams for Testing

All PMO projects will be set up in Microsoft teams. We have a common structure that helps to organize the information (files, meeting notes, recordings, etc.) Testing will also be organized in teams.

Create a Separate Channel for testing and tracking issues.

Use for grouping all testers in one place.

Create a Tab for testing tasks.

* + For issues
	+ For status
	+ For files

Testing kick-off meeting to introduce the testing process.

* + Spin off a meeting chat.
	+ Keep open for duration of testing.

## Defect Tracking

Issues found while testing will be tracked by a ticket in the testing channel in teams. These issues will be worked on immediately during the testing event. For Kuali projects, Jira may be used if issues are not resolved quickly. Jira will also be used when the project moves to production.

In Teams while testing

* + Issues submitted by tester.
	+ Assigned to team members by QA team.
	+ Retested by tester when resolved.

Jira for Kuali and production.

## Test Tools

Currently, we have no special testing tools other than what is listed below. However, I am always looking and evaluating new tools.

Teams

Xmind

Excel/Word

Jira

Automation Tools?

## QA Documentation

If there are documents that needs to be tested, we can do that. We will also supply our own documentation as part of the testing process. These are the documents that may result from this process.

Does project documentation need to be tested?

Test Plan (deliverable)

Test Cases/Scenarios/Scripts (as needed)

Issue Log (in Teams or Spreadsheet)

Status reports (as needed)