

Table of Contents

Introduction	3
What is Agile?	3
Purpose of this Reference Manual	3
Outline of the PMI Agile Practice Guide	3
Selecting the Appropriate Life Cycle for a Project	4
Examples of Agile Practices within ND State Government Projects	5
1. DOT BI Project	5
2. ND Foods - ITD SW Development	5
Iteration Management	6
Iteration Plans	7
Large Project Oversight for Iterative Development	7
3. WSI CAPS Program	8
CAPS Approach	8
How Agile Practices are applied to the CAPS Program	8
Best Practices and Lessons Learned	10
4. WSI Reports/BI Project	10
Addendum 1 – Methodology Selection Tool	12
Instructions:.....	12
Tool Reference:	12
Summary	12

Introduction

This document has been developed for the Information Technology Department (ITD) Enterprise Project Management Office (EPMO). The target audience is the project managers within the EPMO, but the content is intended to be helpful for others in State Government.

NOTE: there are links to documents within this manual that will only be available to employees in ITD. Please contact one of the Agile team members for those documents.

What is Agile?

Agile approaches and methods are umbrella terms that cover a variety of frameworks and methods. If one performs a Google search on the phrase “what is agile,” the return is over 111,000,000 results. Therefore, since this document serves as a guide for project managers, the definition of agile and its key principles is taken from the Project Management Institute (PMI) Project Management Body of Knowledge (PMBOK) Sixth Edition:

Agile: “A term used to describe a mindset of values and principles as set forth in the Agile Manifesto.”

Agile Mindset: “A way of thinking and behaving underpinned by the four values and twelve principles of the Agile Manifesto.”

Agile Manifesto: “The original and official definition of agile values and principles.”

Agile Life Cycle: “An approach that is both iterative and incremental to refine work items and deliver frequently.”

Purpose of this Reference Manual

This document is not intended to teach the project manager about agile concepts. The recommended resource for that purpose is the PMI Agile Practice Guide which was bundled into the latest PMBOK.

The purpose of this Agile Reference Manual (Manual) is to provide you with examples and options for managing an agile project in State Government, utilizing concepts presented in the Agile Practice Guide.

Outline of the PMI Agile Practice Guide

You are highly encouraged to review the PMI Agile Practice Guide before continuing further with this Manual. The electronic version is bundled with PMBOK v6, available for free download with your PMI membership and located under the Foundational Standards area on the PMI website. <https://www.pmi.org/pmbok-guide-standards/foundational/pmbok/sixth-edition>. There is also a hard-copy version available in the EPMO.

The outline of the PMI Agile Practice Guide is as follows:

1. Introduction
2. An Introduction to Agile
3. Life Cycle Selection
4. Implementing Agile: Creating an Agile Environment
5. Implementing Agile: Delivering in an Agile Environment
6. Organizational Considerations for Project Agility
7. A Call to Action
8. Annex, Appendices, References, Bibliography, Glossary, Index

Selecting the Appropriate Life Cycle for a Project

All projects are different and there are a variety of life cycles / methodologies that can be applied. The PMI Agile Practice Guide provides a thorough review of the four general types of project management life cycles, including Agile methodologies.

The following guidance is recommended for selecting the appropriate life cycle for a project, prior to Project Plan approval. The project should be assessed, both in requirements and the means of delivery.

Task	Primary Owner
1. Review the PMI Agile Practice Guide, in particular, <i>Section 3 – Life Cycle Selection</i> and <i>Annex 3 - Common Agile and Lean Frameworks</i> .	Project Manager
2. Review the next section below “Examples of Agile Practices with ND State Government Projects.”	Project Manager
3. Complete the Methodology Selection Tool in Addendum 1 of this Manual.	Project Manager (with input from Sponsor, Technical Lead, and Business Lead)
4. Identify the methodology for the project. Document the selected methodology within the Project Plan.	Project Manager (with input from Sponsor, Technical Lead, and Business Lead)
5. Implement and adjust as needed.	Project Team

Examples of Agile Practices within ND State Government Projects

How is agile being applied on our projects today? The following section summarizes the agile approaches to past and current State Government projects. The project manager(s) who worked on these projects are also named for reference if you have further questions.

NOTE: some of these links may change, please contact the project manager (or a PM Team Lead if that individual has moved on) in case a link is no longer working.

1. DOT BI Project

Project Manager – Heather Raschke (previously Darin Anderson)

The DOT BI Project delivers a one-stop shop for all DOT executives for road construction projects state-wide. The technology is a Cognos Dashboard.

The objective for Phase II is to add data sources to the project dashboard. Since this was to involve small chunks of work, it was decided to use some agile methods to manage this project.

The first agile method is using a sprint/iteration for each data source to be added. This worked well due to the efforts for all data sources repeating the work each time a data source was added.

The second agile method incorporated into the DOT BI Phase II project was stand up meetings. Instead of doing these daily, the team has two stand-up meetings per week (Tuesday/Thursday). These meetings are kept very short and any in-depth conversations are taken offline. This is strictly to give status, discuss road blocks and resolutions for issues.

These methods have worked well for the project and will be incorporated into similar projects in the future.

2. ND Foods - ITD SW Development

Project Manager – Heather Raschke

Originally developed and implemented in 2012 by ITD, NDFoods is a computer system for program application, claims and United States Department of Agriculture (USDA) Foods management. While the application has been a good product for (DPI), it is in need of repair and upgrading to meet the current needs of the agency and its customers.

This project supports DPI's Vision, Mission, and Priorities in the following manner:

- The project's solution will fix inaccuracies in the system and re-work system components to allow for a higher quality of data and decision making.
- The project's solution will incorporate additional budget types and financial management into the system. This is currently managed as a separate process outside of the system, leading to inefficient processes, communications, and decision making.

The Systems Development Life Cycle (SDLC) employed on this project is iterative. Information regarding the iterative development methodology is described below and can also be found in the Project Plan in Section 17.

An iterative development methodology is an approach used to develop software through smaller iterative cycles containing analysis, design, development, and testing processes. With each subsequent iteration, additional features may be added until a full system is formed.

An iterative development methodology allows small but usable chunks of code to be developed, tested, and potentially pushed to production at the end of each iteration. Doing so allows the stakeholders to get a return on their investment throughout the duration of the project. It also allows the project manager and project team to adjust processes as necessary, thereby becoming more efficient with each passing iteration. Lastly, an iterative development methodology reduces the risk of software development by ensuring upfront that the user community and the developers have a similar understanding of the requirements.

Iteration Management

The iterative development methodology to be used on this project will be tailored to the specific circumstances of this project. Factors taken into account include:

1. An existing application will be modified, enhanced, and re-worked to meet the needs of the stakeholders
2. The scope is identified, but may be vague in some instances.
3. The development team lead manages her team through the use of individual WMS assignments for each scope item, which allows the work to be tracked at a more granular level

For this project, there will be three phases. See the project charter for a list of the business objectives and measurements associated with each phase.

<https://share.nd.gov/pmo/dpi/NDFOODS2/Deliverables/NDFOODS%20project%20charter.docx>

Each phase will consist of multiple of iterations. For phase 1, the iterations will be six calendar weeks in length (which may or may not equate to six weeks of business days depending on holidays). Because the development team deploys code into production on Tuesdays, each iteration will start on a Thursday (i.e. two days after the prior iteration is deployed to production). Future scope items and data will be collected during this time for future iterations. Only one iteration will be planned at a time.

Each iteration will have distinctive planning, executing, and closing project management (PM) phases. Each PM phase will coincide with a particular software development life cycle (SDLC) phase

PM Phase	SDLC Phase	Notes
Planning	Analysis	Iterations will be planned using the result of the analysis conducted by the developers. Planning and analysis will take place for new iterations while the current iteration is in testing.
Executing	Design, Development, Testing	This is basically a traditional development life cycle, but in a shortened form with a small scope.

<i>(Monitoring and Controlling)</i>		
Closing	Implementation	<p>If applicable, the tested code is pushed to production.</p> <p>Lessons learned are documented which assists the team in improving processes for subsequent iterations.</p>

Iteration Plans

Iteration plans will be stored in SharePoint at the following location:

<https://share.nd.gov/pmo/dpi/NDFOODS2/Deliverables/Forms/Default.aspx>

An iteration plan template can be found at the following location:

<https://share.nd.gov/pmo/dpi/NDFOODS2/Deliverables/Iteration%20Plan%20Template.xlsx>

Each Iteration Plan will be sent to the project sponsor for approval in WMS – once at the beginning of the iteration and once at the end. The project sponsor will be asked to sign off that the iteration plan is complete and contains the negotiated scope. At the end of the iteration, the project sponsor will again sign-off on the iteration plan, indicating the acceptance of the iterations’ outcome.

Iterations are:

1. Time-boxed. A specific amount of time is chosen for each iteration. The Time-box does not change even if the scope is not completed per the scheduled iteration end date.
2. Flexible. Scope may be transferred from one iteration to another. However, if the scope is not completed at the end of the time-box, the iteration is not considered to be 100% complete.
3. Realistic. Not every scope item will be ready for production at the end of its first iteration. Instead, the status of each scope item should be planned for (e.g., requirement 142 will be 20% coded at the end of the iteration).

Iteration Plans contain both summary and detailed information. The summary provides an overview of the status of the iteration. The detail section contains information about the work to be performed (such as who is assigned, what the anticipated cost is, etc.).

Note: The iteration plans used for this project contain calculated fields in blue text.

Large Project Oversight for Iterative Development

As stated above, time-boxes may be completed, but the associated scope may not be. The variance reports will show the percent complete the scope is for the iteration even through the time-box may have passed. In this manner, a variance can still be calculated.

3. WSI CAPS Program

Project Manager – Jennifer Kunz

CAPS Approach

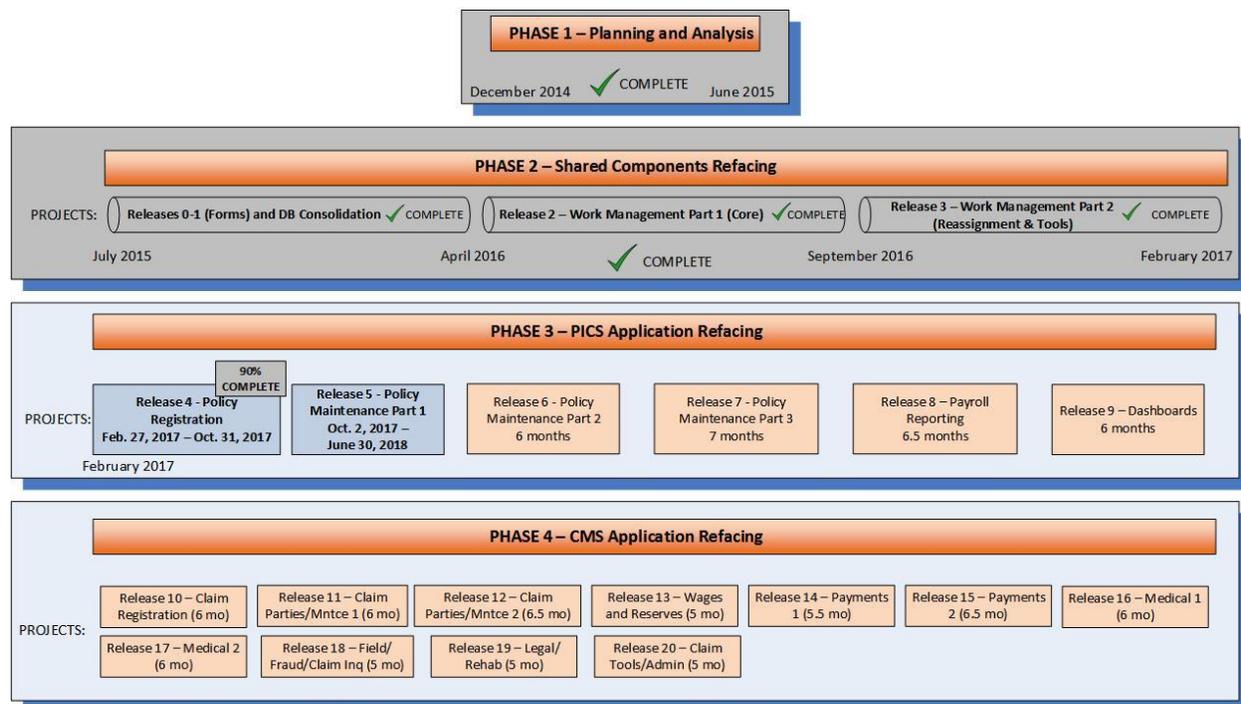
The CAPS program replaces core Workforce Safety and Insurance (WSI) business applications (Work Manager, Claims, and Policy systems) through an evolutionary approach. The user interface is incrementally re-fac'd into one system, CAPS.

There are 20 'Releases' (Projects) over several years, estimated at \$30M. Each Release is planned, baselined, managed, and closed as a 'major IT project' (each is approx. 6-10 months and \$1-2M). Each Release delivers functionality into Production.

The primary project manager is with ITD. ServiceLogix provides the leadership and team for the application re-fac'ing alongside WSI staff. The infrastructure is hosted and supported by ITD.

WSI Claims and Policy System (CAPS) Program

Revised 9/15/17



*End date for Phase 3 and Start/End dates for Phase 4 are unknown due to remaining releases not yet base-lined

How Agile Practices are applied to the CAPS Program

Foundation: The Release Plans (Product Roadmap)

- The Release Plans are the roadmaps for each phase:
 - Shared Components Re-fac'ing (three Releases)
 - Policy System Re-fac'ing (six Releases)
 - Claims System Re-fac'ing (11 Releases)
- Use Cases for each Release are listed and ordered within the Release Plan, as well as the high-level Sprint Plans.

Overview of a Typical Release/Project

Planning

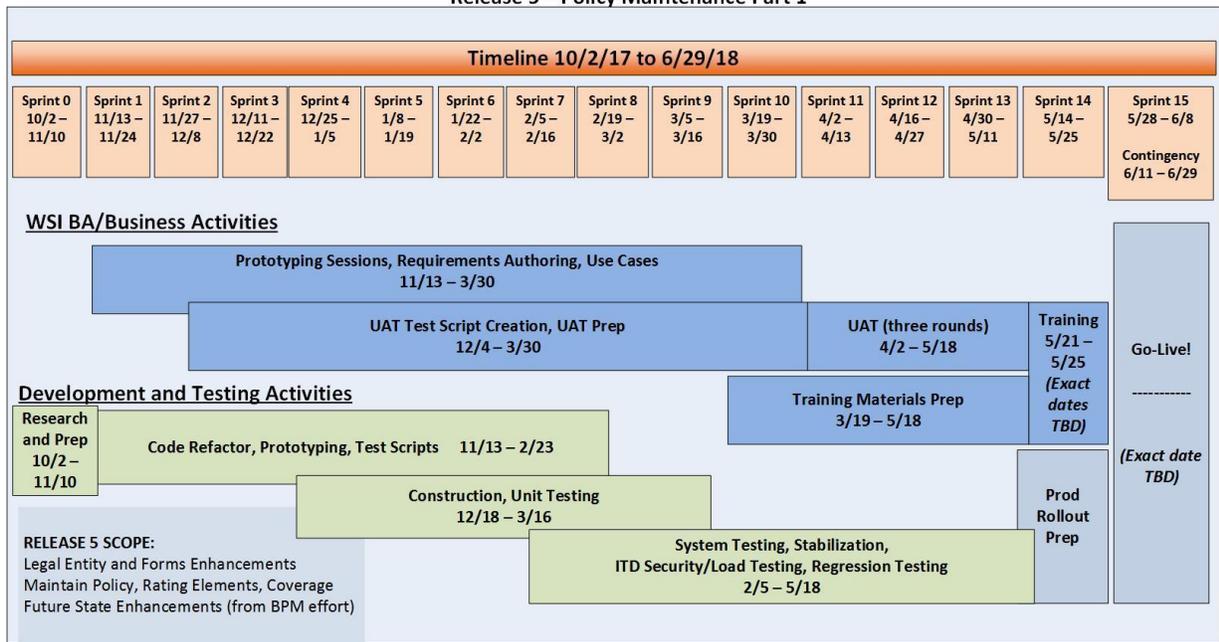
- Release planning begins while the current release is underway, led by the primary PM
- The Release Plan is the basis for scope
- The team develops a sprint summary plan, baseline schedule, budget, and updates to the overall Program Plan
- A new amendment and SOW is developed with ServiceLogix for the release, as well as other procurement revisions
- The ESC approves all planning documents for the release; the Iterative Start-up Report is submitted to LITC
- The development team begins prototyping while the current release is nearing completion
- The Release is kicked off within a week of go-live of the previous release

Execution

- Each Release is comprised of two-week sprints which break major project activities in well-defined, manageable portions of work
- Each sprint has a kickoff with team sprint planning; effort hours are estimated for each task
- ServiceLogix PM leads daily 15-30-minute Scrum meetings
 - Each team member reports what they accomplished yesterday and what's on their plate today; issues are discussed after Scrum
 - A Burn-down chart is calculated daily, showing effort remaining against time
- The Planning Team (Primary PM, Agency/Vendor PM's, Sponsor, other key WSI Managers)
 - Meet weekly to review status of the current release and plan the upcoming release
- Code refactoring, prototyping, and requirements/use case development are iterative activities up to user acceptance testing (UAT)

WSI Claims and Policy System (CAPS) Release 5 – Policy Maintenance Part 1

Revised 11/15/17



Go-Live and Closing

- At each release Go-Live, the new functionality is migrated to Production and previous functionality is no longer available in the legacy system
- CAPS connects to the legacy system, so the user continues to use older functionality for the majority of their work and uses CAPS for areas that have been re-faced
- Closing
 - Release Retrospective – lessons learned are gathered from the main team (ServiceLogix PM), but Sprint Retrospectives are also performed throughout
 - Project Closeout - Primary PM develops the Post-Implementation Survey, Report, and Closeout Report

Best Practices and Lessons Learned

- State PM standards/practices can work well with this hybrid Waterfall-Scrum approach
- Vendor has solid processes throughout, including quality code development and configuration management approach
- Comprehensive QA and UAT Testing
- Solid requirements built through prototyping and comprehensive use cases
- Comprehensive tool for task, workflow, artifact, issue/CR/defect management – ServiceLogix enCorps
- Weekly Tech Touch-base meetings between WSI, ITD, and ServiceLogix team members have been valuable
 - Plan for and manage infrastructure, security, and hardware upgrades
 - Strong architect presence on vendor team is important
- WSI Communications Team and SharePoint Team Site – key communications to all agency staff

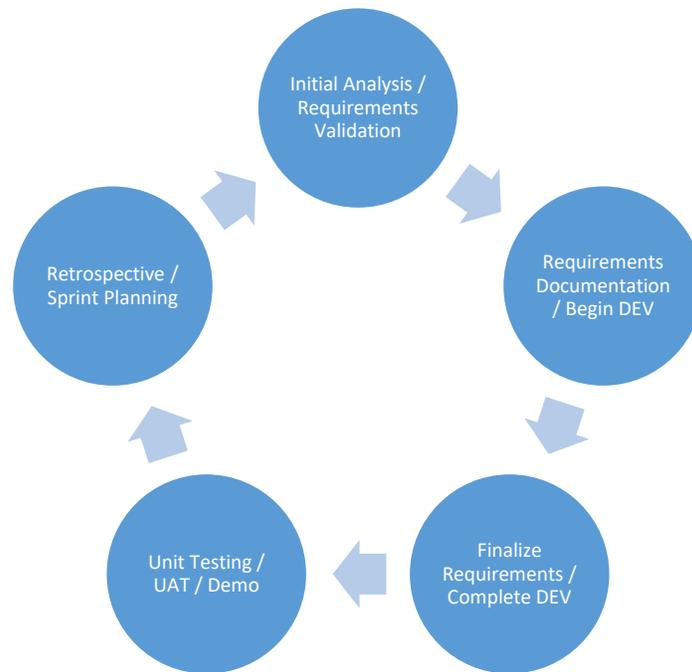
4. WSI Reports/BI Project

Project Manager – Jennifer Kunz

The WSI Reporting Enhancements Project converted WSI transactional Oracle reports to Microsoft SQL Server Reporting Services (SSRS) and added various analytical reports for use with Microsoft Power BI. The benefits achieved include the migration from an unsupported version of Oracle Reports software and provision of business analytics to end users.

The method of project management used in this project is based on the PMBOK, ND project management standards, and some agile methods for the day-to-day project management of reports development and delivery:

- Reports requirements, development, testing, and delivery are estimated and implemented in two-week sprints. This cycle is represented in the chart below:



During the sprint planning sessions, the report developer and business analyst review each report and assign a point value based on the complexity of various factors. The combined point total equals the story points for the report conversion. Development velocity is tracked and measured based on the number of reports developed in the previous sprint.

Addendum 1 – Methodology Selection Tool

The following Methodology Selection Tool (courtesy of the University of Notre Dame’s PMO) is intended to help assess and discuss whether projects should be undertaken using predictive, iterative, agile, or blended approaches.

Instructions:

1. Make a copy of the Excel File (located on PMO SharePoint site at: [EPMO Methodology Selection Tool](#))
2. Answer each question for the project, more than 1 answer is perfectly acceptable
3. Identify top scoring methodologies for Planning and Execution
4. Combine Planning and Execution methodologies to identify best pure or blend options. See sample1 and sample2
5. If a blend, use the Blend Matrix ([Agile Methodology Blends Matrix](#)) to guide decision on which to choose
6. Identify the methodology for the project. Document the selected methodology within the Project Plan.
7. Implement and adjust as needed.

Tool Reference:

Original tools courtesy of University of Notre Dame PMO
ntrda.me/decision
ntrda.me/blends

Summary

Tools like the one above and in Appendix X3 of the PMI Agile Practice Guide, are useful for identifying potential fits and gaps for agile approaches. They should not be used as definitive inclusion or exclusion gates, but instead as topics for objective discussion with all interested parties.