

Chapter 1

Application Lifecycle Management Introduction

This chapter discusses the various documents and resources that are available in the PDSA Agile ALM from which you can create your own Software Development Guidelines. This chapter also serves as an introduction.

This document contains the *road map* to the other chapters as well as the references to the resource documents.

Table of Contents

Chapter 1	1-1
Application Lifecycle Management Introduction.....	1-1
Table of Contents	1-2
Your Road Map to the ALM Chapters	1-3
Definition of ALM	1-5
The PDSA Agile ALM Process.....	1-6
Requirements Analysis	1-6
Client Requirements	1-7
Discovery Phase (P1)	1-7
Architecture and Design (P2)	1-7
Software Configuration Management.....	1-7
Build or Construction Phase	1-8
Software Release and Issue Management	1-8
User Manuals	1-9
Quality Assurance & Test Plans	1-9
Performing QA	1-10
Customer Buy Off	1-10
Change Orders	1-11
Lessons Learned	1-11
Application Documentation	1-12

Your Road Map to the ALM Chapters

The collection of documents in the DOCS\ALM folder is as follows:

Location: Docs\ALM\	Description
Title: Application Lifecycle Management Introduction Ch01-ALM-Intro.docx	This document provides an overview of all the material contained in the DOCS\ALM collection.
The following chapters are the detailed processes and procedures that your software development project manager, senior developer, and developer will use to ensure application software is built following a structured process.	
Title: PDSA Discovery Phase Ch02-ALM-P1-Discovery.docx	The following are the action items associated with this phase: <ul style="list-style-type: none"> • Value proposition • ROI • Proof of concept • Initial prototype • Initial specifications • Use Case Summaries • Initial data model
Title: PDSA Architecture and Design Phase Ch03-ALM-P2-ArchDesign.docx	The following are the action items associated with this phase: <ul style="list-style-type: none"> • Prototype • Specifications • Security • Business Rules • Data Model • Use cases • Estimates/Milestones
Title: PDSA Build Phase Ch04-ALM-P3-Build.docx	This document describes the portion of the ALM process from when a Project Manager (PM) receives the deliverables of PDSA Architecture and Design Phase and constructs the software and performs unit testing. <ul style="list-style-type: none"> • Software

Location: Docs\ALM\	Description
	<ul style="list-style-type: none"> • Unit Tests • Iterative development • Source Code Control
<p>Title: PDSA Test Phase Ch05-ALM-P4-Test.docx</p>	<p>Testing an application is a very challenging task. If solid specifications, business rules and data model are developed, then testing should be easier. Other issues that can impact testing are the project budget and client relations.</p> <ul style="list-style-type: none"> • Integrated testing • Independent testing • Customer testing and buyoff
<p>Title: PDSA Implementation Phase Ch06-ALM-P5-Implementation.docx</p>	<p>The PDSA Implementation Phase ensures that the client’s infrastructure is in place to support and run their new system, as well as perform the actual installation.</p> <p>Since the PDSA development process requires incremental delivery, this step must be in place before the first incremental release and not just at the end.</p>
<p>Title: PDSA Support Phase Ch07-ALM-P6-Support.docx</p>	<p>PDSA provides a variety of options for support. This phase typically begins when the project is completed and installed and running in production.</p> <p>Support Options:</p> <ul style="list-style-type: none"> • PDSA provides knowledge transfer and hands over continued support of the application to the client. • PDSA provides continued support for the application on a time and material basis. • PDSA and Client put into place a Service Level Agreement (SLA). This is a contractual support document.

Resource Documents

Documents that you should read through for this section:

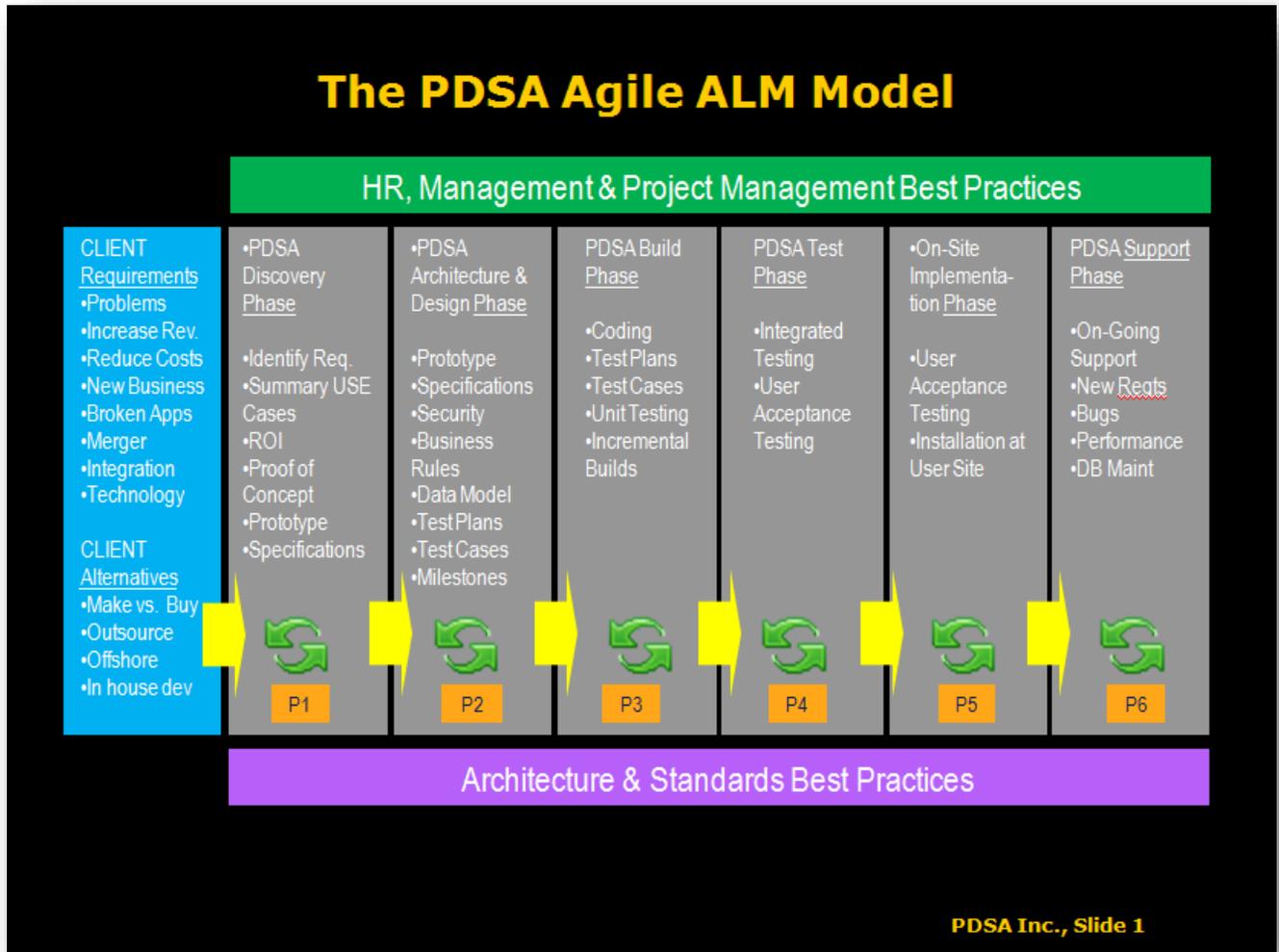
Resource	Description
\Resources\ ApplicationLifecycleManagement*	All the resource documents for the this chapter.

Definition of ALM

ALM is the administration and control of an application from inception to its demise.

1. It embraces requirements management, system design, software development and configuration management and implies an integrated set of tools for developing and controlling the project.
2. Also incorporates: portfolio prioritization, architecture, service, and security policies to concept, delivery, ongoing management, and change in operations.

The PDSA Agile ALM Process



Requirements Analysis

Performing an analysis of user needs and capturing those requirements is an art form unto itself. However, there are certain aspects that you want to make sure you capture for every project. There are a few different examples of Requirements Analysis documents that you can read to get an overview of how to gather requirements and what should be a part of a specification document that you can turn over to a programmer.

Specifications also include a full data model. We have included a sample document for that as well. If you have tools that can diagram your database, that is very helpful as well.

Client Requirements

How do projects begin in your organization? Some formal and some informal?

Discovery Phase (P1)

Architecture and Design (P2)

Prototyping

Data Model / Database Design

Detailed Business Specifications

Estimate

Resources

Scheduling

Software Configuration Management

Software Configuration Management (SCM) is critical to your organization. If it is not, it sure should be. These resources describe how to apply SCM practices to your organization. Once you start a project you will need to keep your source code safe in some sort of source code control system. Visual SourceSafe (VSS) comes as a part of Visual Studio and has been used successfully for a lot of companies, including PDSA, Inc. One document listed below presents the usage of VSS 6.0 in the software configuration management practice. PDSA, Inc. strongly recommends that all software be maintained by a software configuration management tool, such as VSS. Another document, in detail, describes how to use VSS with ASP.NET applications – which can be tricky, so there is much to learn here.

The following document is an overview of the usage of, and guidelines for using VSS.

Resource	Description
\\ALM\SCM-VSSforDOTNET.docx	This word document is a tutorial and step by step help guide describing the best practice for using VSS with ASP.NET applications.
\\ALM\SCM-UsingVSS.docx	This document covers in detail the Training Guide and Usage Practices for Microsoft's Visual SourceSafe 6.0.
\\ALM\UsingTFS.docx	Coming soon!

Build or Construction Phase (P3)

Yes, we know. This is the fun part. Before any project is kicked off, the whole team should be reminded of your best practices.

Also, unit testing is now critical in the development of your software. This document describes the use of nUnit for VS2005.

Resource	Description
\\ALM\BuildBestPractices.docx	This document outlines what PDSA does at the beginning of the development phase to ensure that we follow a consistent set of standards and architecture practices.
\\ALM\BuildUnitTesting.docx	This document describes how to use Unit Testing in .Net.

Software Release and Issue Management

Typically, you will release your software in increments. Maybe a large application will have 6 releases. After each release the customer or user will begin testing the functionality. As issues arise via conversation, phone calls, and emails, you must log them, track them, and provide on-going status to your customer. We have provided a simple spreadsheet to help you. Of course, the hard part here is dedication to project management.

Also, using a web-based tool to manage this process is more efficient. PDSA uses "bugtracker," a service tool on the internet. This or something like this would be very helpful.

Resource	Description
\\ALM\IssueTaskListTemplate1.xls	The XLS spreadsheet is a very simple way to keep track of all the open/closed issues that occur during system release.
\\ALM\IssueTaskListTemplate2.xls	The XLS spreadsheet is a very simple way to keep track of all the open/closed issues that occur during system release.
\\ALM\TestingUsingPDSABugTracker.docx	Sample instructions on its use.

Testing/User Acceptance (P4)

User Manuals

It is important to have user manuals. Yes, even in this day and age. A very easy and low cost way is to write a user manual using MS Word and take screen shots to add to the document. We recommend SnagIt for easy screen capture. Then simply make a PDF file and make a link on your application to bring it up. Or use this technique for multiple chapters too.

Resource	Description
\\ALM\UserManual-Sample1.docx	This document is an example of a user manual.
\\ALM\UserManual-Sample2.docx	This document is an example of a user manual.

Quality Assurance & Test Plans

Most programmers are not very good at testing their own applications. This is mainly due to they just do not think about the testing process they should go through. If you make them sit down and create a test plan for their application however, you will find that they can actually think of quite a few scenarios of how to break their code. It is even better to have one of their peers write up procedures for how to break another programmer's code. The following documents describe the types of tests that you should employ for creating test plans for testing applications.

Resource	Description
\\ALM\TestPlan-Sample1.docx	This document is the test plan vs. the test cases themselves. This document outlines WHAT needs to be tested.
\\ALM\TestCases-Sample1.docx	This document is a collection of test cases: HOW to test a functional feature.
\\ALM\TestCases-Sample2.docx	This document is a collection of test cases: HOW to test a functional feature.

Performing QA

It is recommended that you always have someone other than the original programmer perform the Quality Assurance on an application. They should follow any test plans developed by the original programmer, but also add any additional test plans as well. By adding additional tests you can make any regression tests by other programmers easier to perform.

Customer Buy Off

The last step in delivering an application to a user is getting them to buy off that all of the items in the specification are in the working program. After verifying this with the user you should get the buy off from the user that they are satisfied. We do this so we can start the warranty period for the customer. During the warranty period we fix any bugs free of charge. After the warranty period then any bugs are fixed at our normal hourly rate. The customer buy off form clearly states this fact. This leads to a good paper trail to keep all parties involved honest and working toward the best interest of the relationship.

The following is the form we use for customer buy off that a project is complete.

Resource	Description
\\ALM\CustBuyOff.docx	This document is basically a sign off form stating that the application is complete and details the warranty period and definitions of what a warranty is.

Implementation Phase (P5)

Support Phase (P6)

Change Orders

During or after a project, changes to the system are requested. A documented process for change orders is recommended. As part of that process, this sample is a detailed example of just what a change order should look like.

Resource	Description
\\ALM\PDSA-ChangeOrder-Sample.docx	This document is a detailed sample of a change order.

Lessons Learned

PDSA-ChangeOrder-Sample.docx After a project is complete, the project manager and the team needs to sit down and document how to improve the processes that were used and/or create new processes and tools for improvements for the next time.

The following is the form we use for customer buy off that a project is complete.

Resource	Description
\\ALM\LessonsLearnedProcess.docx	This document is basically a sign off form stating that the application is complete and details the warranty period and definitions of what a warranty is.

Application Documentation

After a project is complete, the project manager needs to initial the application or technical documentation. Time must be taken to sufficiently describe and document the system so that other programmers in the future (or even the programmers that wrote the system) can easily understand the architecture and layout of the system.

Resource	Description
\ALM\ApplicationDocumentation-Sample1.docx	This is a sample of an application description document.
\ALM\ApplicationDocumentation-Sample2.docx	This is a sample of an application description document.

Summary

In this section of the PDSA Agile ALM you learned many of the various guidelines that you should employ to create your Software Development Lifecycle. By following each of the procedures in this section you will find that your project management will go much smoother.