[Project]

Quality Plan

CxTemp_QualityPlan.doc

Draft X

May 26, 2004

[Organization Name]

[Paste Your Organization's Logo Here]

Powered By

www.construx.com

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and Number	Full Name	Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. *CxPattern_RevisionHistory* provides details on CxOne's recommended way to handle document revisions.	00/00/00

The paragraphs written in the "Comment" style are for the benefit of the person writing the document and should be removed before the document is finalized.

This template is used to create a quality plan. A quality plan defines a project's quality goals and outlines the quality assurance and quality control practices the project will use to ensure those quality goals are met.

*This template is intended medium to large projects that have separate quality and test plan documents. Very small projects may roll quality and test planning up into the project plan document (see **CxTemp_ProjectPlan**). Small to medium projects without significant testing activities may use **CxTemp_QualityPlanLite** to document both quality and test activities.*

***CxStand_Quality** is the reference that defines software quality for CxOne. The document created from this template should leverage information defined in that standard and other CxOne materials as much as possible, only documenting what is specific and unique to this project. The document created from this template should not simply repeat information that may be found in other project documents, unless it is adding additional information specific to quality.*

See CxGuide_CxOneArtifact for details on how to utilize the advanced features of CxOne artifact templates.

Contents

1 Introduction

*CxOne defines the quality plan as a supporting plan to the project plan. Static quality control activities (i.e., detecting defects through reviews) are defined in this plan. This template assumes that dynamic quality control (i.e., detecting defects through testing) is delegated to a separate test plan (see **CxTemp_TestPlan**).*

Quality assurance planning (i.e., the prevention of defects) is also defined in this plan. Quite often quality assurance decisions manifest themselves in broader project planning such as lifecycle selection and risks management processes. The quality plan should not be redundant with information in the project plan, only providing detail on processes and decisions when they are unique, specific, and critical to the project's quality goals.

1.1 Overview

Provide a summary of the purpose, scope, and objectives of the quality plan. Briefly summarize activities and describe how they relate to other project activities. If appropriate, state why the plan exists, define the audience, and provide other background or contextual material as appropriate.

1.2 Assumptions and Constraints

State the bounds, assumptions, and limitations of the plan including both items within and beyond the scope of this plan. This should build off assumptions in the project charter and project plan with issues specific to quality.

1.3 Risks and Assets

Describe any risks and assets that impact quality planning. Risks should be accompanied by mitigation plan. This section should not repeat risk and asset planning already covered in other documents, unless there is a specific quality aspect that has not been dealt with. This section should cover long term risks that affect a significant portion of the project's lifecycle. Short-term risks requiring frequent update should be handled through the project's risk management mechanisms.

1.4 Reference Materials

(Optional) – Bibliographic listing of non-project and non-CxOne materials referenced in the plan. CxOne and project materials are normally referenced in-line.

1.5 Definitions and Acronyms

*(Optional) – List any configuration management definitions and acronyms introduced to the project by this plan. Do not repeat items covered in the **CxOne_TermsAndAcronyms** document or in a global project definitions and acronyms document.*

When appropriate, this section should simply contain pointers to other project documentation.

2 Quality Goals

Describe the quality goals or requirements for the project (e.g., functionality, accuracy, reliability, usability, learnability, schedule, etc.) If this information is contained in the requirements or other project documentation, there should be a link to it.

For each goal summarize the quality approach the project will use to meet it. This should include the workflow of components through the project lifecycle. Whenever possible, this should be outlined in a diagram as well as text. Section 3 will expand and support the information provided here.

3 Quality Approaches

Describe the tools, processes, techniques, and standards used to ensure that the quality goals for the project will be meet. Listed below is a standard set of areas that each project should address. Additional sections should be added as appropriate to meet specific project goals.

3.1 Reviews

*Describe the review levels (per **CxStand_ReviewProcess**) to be used for this project's artifacts. Normally this is done by identifying groups and classes of artifacts (e.g., production source code), but critical artifacts may be named individually in this section (e.g., project plan).*

Some examples of this are:

Project Artifact	Review Level
Project Plan	Inspection
Software Quality Plan	Inspection
Requirements Documents	Inspection
UI Prototypes and designs	All will be inspected with walkthroughs
Architecture	Inspection
Low level designs	All low-level designs will be reviewed. • 20% reading or full inspection • 30% walkthrough • 50% desk check
New production code Automated test cases	40% of new production code will be reviewed • 5% reading inspection or full inspection • 15% walkthrough • 20% desk check In addition, at least 80% of the code should see collaborative construction coverage of at least 2 engineers
Existing production code	Existing production code that is not modified will only be reviewed on a case by case basis if the design or construction lead is concerned about interactions with new code.
Supporting code (infrastructure, test automation framework, utilities)	15% of non production code will be reviewed • 5% walkthrough • 10% desk check In addition, at least 50% of the supporting code should see collaborative construction coverage of at least 2 engineers

3.2 Testing

*This section should link to the test plan for test planning information details, or incorporate the details of **CxTemp_TestPlan** in this section. If linking, a brief summary may be provided here. In some cases it will make sense to start the test plan here early in a project, and then migrate it from this section into its own document.*

3.3 Standards

Describe or list the standards that the project will conform to. Examples include construction standards, documentation standards, test standards, design standards, etc.

3.4 Audits

Describe the internal or external audits (project reviewer, etc.) that will occur during the project. Examples include functional or physical audits, in process audits, managerial reviews, etc.

3.5 Tools, Techniques, and Methodologies

Describe any other special tools, techniques, and methodologies that will be used to perform quality assurance and quality control for the project. This section shouldn't repeat information in the project plan, unless there are specific quality aspects not covered there.

4 Project Metrics

If appropriate, this section should describe processes and specific metrics used to control the quality of both the project and resulting product. There is normally significant overlap in this area between the quality plan and other plans, especially the project plan. Even if other plans do not explicitly call out processes and metrics, they may be obviously implied by certain tools, processes and techniques.

This section provides a good location to document in a centralized location an overview of all metrics processes and what is being measured and use on a project. However, depending on your organization, size and nature of the project, project goals, other planning and processes, etc. this section may not make sense.

If your organization has mature and well defined metric processes for projects, this section can just point to that material.

4.1 Metric Goals

This is section is an informal GQM (Goal, Question, Metric) analysis of what you are trying to accomplish through project metrics.

4.2 Metric Processes

Summarize the processes used to collect measures, convert measures into metrics, and mechanisms to use that information to control project activities and improve project processes. If appropriate discuss any relationships to organizational metrics.

4.2.1 Project Measures

Describe the data that will be collected for the project. Consider including the following items:

Project Management

- *Time*
- *Effort*
- *Cost*
- *Milestones achieved*
- *Staffing level and type*

Construction

- *Lines of Code*
- *Numbers of artifacts like files, classes, screens, etc.*
- *Function Points*

Requirements and Change Control

- *Requirements delivered*
- *Change request counts (best when weighted)*

Quality Assurance

- *Defect counts (best when weighted)*
- *Defect trends*
- *Reviews completed*
- *Test cases passed*

4.2.2 Project Metrics

Describe the type of metrics that derived from measures and used on the project. Consider including the following items:

- *Percentage of plan completed*
- *Variances from planned estimates*
- *Growth of size over time*
- *Change request and defect trends*

5 Quality Gates

Describe the quality objectives that will enable the project to measure the quality of the project at multiple points in the project lifecycle. No text is necessary between the heading above and the heading below unless otherwise desired

5.1 Major Milestone Quality Criteria

Describe the major quality checkpoints during the project lifecycle. These are used to ensure the required features are present, stable, and functional. By verifying these items at multiple points during the project, the team can ensure that a high quality product is being produced at all times.

5.1.1 Milestone One Criteria

Summarize the milestone and describe its quality goals and measurements.

Topic	Description
Purpose	*Describe the significant elements of the milestone that should be verified by the Quality Plan.*
Quality Goals	*Describe the quality goals that should be met for this milestone. E.g. all review goals have been meet, verification of the requirements has been completed, etc.*
Participants	*Describe the critical decision makers involved in determining if the appropriate criteria have been met. Examples include the QA Lead, Project Business Manager, Customer representative, etc.*
Entry Criteria	*Describe any prerequisite criteria that must exist before the milestone verification can begin. Examples include a stable install exists, all known issues have been dispositioned by the Change Control Board, etc.*
Exit Criteria	*Describe the criteria that must be met before the milestone can be declared complete. Examples include stated quality goals are met, source code and documents are versioned and archived, etc.*
Outcomes	*Describe the outcomes from the milestone. Examples include software and documentation is released to the customer, the next phase of the project begins, etc.*

5.1.2 Milestone N Criteria

Summarize the milestone and describe its quality goals and measurements.

Topic	Description
Purpose	
Quality Goals	
Participants	
Prerequisites	
Exit Criteria	
Outcomes	

5.1.3 Final Release Criteria

Describe the quality goals and measurements for the final product release.

Topic	Description
Purpose	
Quality Goals	
Participants	
Prerequisites	
Exit Criteria	
Outcomes	