

**Tailored CMM for a Small e-Commerce Company
Level 2: Repeatable**

Annie I. Antón
Ryan A. Carter
Hema Srikanth
Ashish Sureka
Laurie A. Williams
Kai Yang
Lingyun Yang

{anton, williams}@csc.ncsu.edu

North Carolina State University
Department of Computer Science
Technical Report
TR-2001-09

August 23, 2001

Abstract

E-commerce software developers are under pressure to develop applications at a record pace. The widespread lack of process discipline and procedural guidance available for such market driven environments highlights the need for a framework to support practitioners' development efforts. The EPRAM Model (NCSU Computer Science TR-2001-08) explicitly addresses the challenges inherent in small-team rapid development projects requiring no more than 12 people. The model was intentionally designed to comply with the Level 2 KPAs (Key Process Areas) of the Software Engineering Institute's Capability Maturity Model (CMM); it combines evolutionary prototyping with an aggressive risk mitigation strategy to ensure that proposed requirements adhere to all established security and privacy policies. This document presents the Tailored Capability Maturity Model for small E-Commerce development teams.

Overview

The SEI's CMM is an important standard by which the "maturity" of a software process can be judged. The CMM includes a set of rigorous practices that impose large amounts of overhead that may be especially detrimental for small resource-challenged organizations. It is widely applicable given a willingness to tailor it and the SEI advocates such tailoring [PCC93]. While we were initially hesitant about using the CMM due to its overhead and rigor, we found it possible to incorporate the practices that support our process model while tailoring the remaining practices so that they made sense in the context of the environments in which we would employ the EPRAM model.

The CMM contains five levels of increasing maturity. It reinforces a series of best practices through its use of Key Process Areas (KPAs) for each level of maturity. Each KPA is subdivided into goals that must be met before that KPA is satisfied. Additionally, all KPAs must be satisfied before a process obtains the recognition of having achieved a given maturity level. For purposes of this paper, we focus on the Repeatable Level (Level 2) which provides a suitable basis for process quality without introducing overhead unnecessary for small teams. We examined five of the six KPAs in the Repeatable Level of the CMM: Requirements Management, Software Project Planning, Software Project Tracking and Oversight, Software Quality Assurance, and Software Configuration Management [PCC93], but omitted Subcontract Management, as it is not often relevant to the entrepreneurial rapid development environments targeted by our efforts.

The CMM does not directly support rapid, iterative, poorly-resourced software development efforts. For this reason, we were initially concerned that our EPRAM model may not meet the CMM's strict practices for process maturity, and yet, we realized the need to incorporate the ideals of the CMM. We opted for "some" improvement, recognizing that the introduction of any process improvements would, at a minimum, yield far better quality in the organization as well as the resulting process. The EPRAM model, introduced in Section 3, accommodates small teams, working on small e-commerce applications. The CMM Level 2 KPAs are incorporated as much as possible so our model upholds the spirit of the CMM and benefits from its process improvement guidance [JB00].

Our CMM tailoring began by determining the roles and responsibilities for a team's members. We established nine roles for each team; one person may have multiple roles or multiple people may fulfill a single role. Each role is assigned to one of three groups (the Software Engineering Group, the Software Development Group, or the Software Quality Assurance Group) that collaborate in specific areas or for a particular purpose during a project. Once the roles and responsibilities were defined, we extracted and discussed each CMM Level 2 goal, commitment, ability, activity, measurement, and verification. In many cases, the tailoring simply required a bit of "wordsmithing" and/or clarification to more adequately reflect the needs of the smaller organizations for which the model is intended; however, others required more significant changes. There are 115 elements (Goals, Commitments, Abilities, Activities, Measurement, and Verification) in the five KPAs that we considered for Level 2. Of these, 43 were modified, 9 were rejected as not applicable, and 1 was marked as optional. We maintained the spirit of the CMM throughout our tailoring efforts as it reflected new team roles, established a hierarchy of team leadership, accounted for limited resources, satisfied the demands for rapid development, and removed overhead that was not applicable to the new process model for our target audience. The final tailored CMM served as the basis for the EPRAM model .

References

- [ACE01] A.I. Antón, R.A. Carter, J.B. Earp and L.A.. Williams. *EPRAM: Evolutionary Prototyping Risk Analysis & Mitigation (e-Commerce Software Development Process Document)*, North Carolina State University, Department of Computer Science, TR-2001-08, 20 August 2001.
- [CAD01] R.A. Carter, A.I. Antón, A. Dagnino and L.A.. Williams. Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model, to appear *IEEE 5th International Symposium on Requirements Engineering (RE'01)*, Toronto, Canada, August 2001.
- [CAD01] R.A. Carter, A.I. Antón, A.Dagnino and L. Williams. Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model. To appear *5th IEEE International Symposium on Requirements Engineering*, Toronto, Canada, August 2001.
- [JB97] D.L. Johnson & J.G. Brodman. Tailoring the CMM for Small Businesses, Small Organizations, and Small Projects, *Software Process Newsletter*, No.8, IEEE Computer Society, Winter 1997.
- [PCC93] M.C. Paulk, B. Curtis & M.B. Chrisis. *Capability Maturity Model for Software. Version 1.1*, Software Engineering Institute Technical Report, CMU/SEI-93-TR, February 24, 1993.

Tailored CMM for a Small eCommerce Company

Level 2: Repeatable

Software Engineering Seminar Group
October 24, 2000

The following set of tables includes a description of a tailored CMM approach developed for a small eCommerce company. This tailoring was performed based on these assumptions:

- The eCommerce company is a small company, employing approximately 4 to 6 people.
- The organization needs to achieve a level 2 (Repeatable) maturity rating. Higher levels are not appropriate for the organization.
- The organizational structure of eCommerce teams is relatively flat.
- Team roles are the Project Manager, Technical Writer, Requirements Engineer, Business Manager, Software Quality Assurance Manager, Chief Architect, Lead Programmer, User Interface Design Specialist, Subject Specialist(s).
- One person may have multiple team roles.
- The Project Manager, Technical Writer, Requirements Engineer, and Business Manager compose the Software Engineering Group; the SQA Manager and Chief Architect compose the Software Quality Assurance Group; the Lead Programmer, User Interface Design Specialist, and Subject Specialist(s) compose the Software Development Group.
- Project time is limited. Approximately 800 development hours are allocated for the project.

Code Description

Code	Description
A	Accept – Necessary and desirable practice that is acceptable as written.
E	Expand – Necessary and desirable practice that requires the addition of local definitions for one or more terms to be used in this environment.
T	Tailor – Necessary and desirable practice that requires some adjustment to be used in this environment.
O	Optional – Practice may be useful for some, but not all, projects in this environment.
NR	Not Recommended – Practice is not recommended for this environment.

CMM Elements – Requirements Management

CMM Element	Code	Rewrite / Rework / Clarify	Rationale
Goal 1: System requirements allocated to software are controlled to establish a baseline for software engineering and management use.	A		General enough to accept as written.
Goal 2: Software plans, products, and activities are kept consistent with the system requirements allocated to software.	A		General enough to accept as written.
Commitment 1: The project follows a written organizational policy for managing the system requirements allocated to software.	T	The project follows a written project policy for managing the system requirements allocated to software. This policy typically specifies that allocated requirements are documented; the allocated requirements are reviewed by the requirements engineer under the project manager and individuals responsible for the tasks listed; and the software plans, work products, and activities are changed to be consistent with changes to the allocated requirements.	<ul style="list-style-type: none"> • Changed to reflect roles established for the project • The project policy is the organizational policy. • Changed to denote sections within the policy.
Ability 1: For each project, responsibility is established for analyzing the system requirements and allocating them to hardware, software, and other system components.	A		Valid.
Ability 2: The allocated requirements are documented.	A		Valid.
Ability 3: Adequate resources and funding are provided for managing the allocated requirements.	E	Adequate resources and funding are provided for managing the allocated requirements. Subject specialists are assigned to manage appropriately allocated domain or technical requirements. Tool support for the activities for managing requirements may or may not be available – funding dependent.	<ul style="list-style-type: none"> • Changed to reflect roles established for the project. • Since resources are limited in small companies, tools may not be available. Company dependent.

Ability 4: Members of the software engineering group and other software-related groups are trained to perform their requirements management activities.	T	Members of the project team and other software-related groups are trained to perform their requirements management activities. Training can take the form of on-demand lecture, self-study, peer study, out-sourced training, or other applicable methods.	The methods of training must be flexible since smaller companies can not afford extensive training (time or cost).
Activity 1: The software engineering group reviews the allocated requirements before they are incorporated into the software project.	E	<p>The software engineering group reviews the allocated requirements before they are incorporated into the software project.</p> <ul style="list-style-type: none"> • Incomplete and missing allocated requirements are identified. • The allocated requirements also need to be reviewed to determine whether they include security and privacy considerations. • Any allocated requirements identified as having potential problems are reviewed with the individuals responsible for analyzing and allocating system requirements, and necessary changes are made. Any changes can be written as complementary parts to the original documents. • Commitments resulting from the allocated requirements are negotiated with the individuals responsible for or affected by the changes. 	<ul style="list-style-type: none"> • In an eCommerce company, security and privacy should be considered during the requirements stage. • There is no need to write an entirely new document for changes. Helps to ease the documentation load for time – limited project. • Changes are made to reflect the roles established for the project.
Activity 2: The software engineering group uses the allocated requirements as the basis for software plans, work products, and activities.	A		Valid.

Activity 3: Changes to the allocated requirements are reviewed and incorporated into the software project.	E	<p>Changes to the allocated requirements are reviewed and incorporated into the software project.</p> <ul style="list-style-type: none"> The impact to existing commitments is assessed and changes are negotiated as appropriate. Changes to commitments made to individuals and groups external to the organizations are reviewed with the project manager. Changes to commitments within the organization are negotiated with the individuals responsible for or affected by the changes. The changes that need to be made to the software plans, work products, and activities can be written as complementary parts to the original documents. 	<ul style="list-style-type: none"> Changes are made to reflect the roles established for the project. There is no need to write an entirely new document for changes. Helps to ease the documentation load for time – limited project.
Measurement 1: Measurements are made and used to determine the status of the activities for managing the allocated requirements.	A		Valid.
Verification 1: The activities for managing the allocated requirements are reviewed with senior management on a periodic basis.	NR		<p>Not Recommended. Performed in Verification 2.</p> <ul style="list-style-type: none"> These ‘reviews’ are done within the reviews established elsewhere. The flat management structure makes this irrelevant.
Verification 2: The activities for managing the allocated requirements are reviewed with the project manager on both a periodic and event driven basis.	A		Valid.
Verification 3: The software quality assurance group reviews and/or audits the activities and work products for managing the allocated requirements and reports the results.	T	The software quality assurance group or individual reviews and/or audits the activities and work products for managing the allocated requirements and reports the results.	Changed to reflect that the software quality assurance group may be one individual.

CMM Elements – Software Project Planning

CMM Element	Code	Rewrite/Reword	Rationale
Goal 1: Software estimates are documented for use in planning and tracking the software project	A		General enough to accept as written.
Goal 2: Software project activities and commitments are planned and documented	A		General enough to accept as written.
Goal 3: Affected groups and individuals agree to their commitments related to the software project	A		General enough to accept as written.
Commitment 1: A project manager is designated to be responsible for negotiating commitments and developing the project's software development plan.	A		Valid. The project manager works with everyone on the team to perform this due to the company's flat organizational structure.
Commitment 2: The project follows a written organizational policy for planning a software project.	A		
Ability 1: A documented and approved statement of work exists for the software project.	A		Valid.
Ability 2: Responsibilities for developing the software development plan are assigned.	A		Valid.
Ability 3: Adequate resources and funding are provided for planning the software project.	T	Adequate resources are provided for planning, yet are limited and used conservatively.	The company is a start up and has limited resources with which to begin.
Ability 4: The software managers, software engineers, and other individuals involved in the software project planning are trained in the software estimation and planning procedures applicable to their areas of responsibility.	T	The project manager has work experience and training in managing the technical and personnel aspects of the software process. Different modules are developed for training purposes by the management. Training can take the form of on-demand lecture, self-study, peer study, out-sourced training, or other applicable methods.	<ul style="list-style-type: none"> • Training is provided by developing modules for the employees and higher management is expected to have work experience. • The methods of training must be flexible since smaller companies can not afford extensive training (time or cost).
Activity 1: The software engineering research group participates on the project proposal team.	E	The software engineering group participates on the project proposal team. The software engineering research group consists of a technical writer and project manager.	As the project has limited resources, there might not be a separate software engineering group.

Activity 2: Software project planning is initiated in the early stages of, and in parallel with, the overall project planning	A		Valid.
Activity 3: The software engineering research group participates with other affected groups in the overall project planning throughout the project's life.	T	The project manager involves the whole team in the software process and planning.	As the project has limited resources, there might not be a separate software engineering group in place.
Activity 4: Software project commitments made to individuals and groups external to the organization are reviewed with senior management according to a documented procedure.	T	Software project commitments made to individuals and groups external to the organization are reviewed with the project manager according to a documented procedure. Documented procedure is included within the provided Review Meeting template.	Changed to reflect the roles and templates established in the project.
Activity 5: A software life cycle with predefined stages of manageable size is identified or defined.	A		Valid. Project manager takes responsibility for this.
Activity 6: The project's software development plan is developed according to a documented procedure.	A	Documented procedure is included within the provided Project Plan template.	Valid.
Activity 7: The plan for the software project is documented.	A		Valid.
Activity 8: Software work products that are needed to establish and maintain control of the software project are identified.	A		Valid.
Activity 9: Estimates for the size of the software work products (or changes to the size of software work products) are derived according to a documented procedure.	A	Documented procedure is included within the provided Software Estimation template.	Valid.
Activity 10: Estimates for the software project's effort and costs are derived according to a documented procedure.	A	Documented procedure is included within the provided Software Estimation template.	Valid. Estimates from the past cases may not be available to the team. Therefore, good judgment would be used.
Activity 11: Estimates for the project's critical computer resources are derived according to a documented procedure.	A	Documented procedure is included within the provided Software Estimation template.	Valid. Estimates from the past cases may not be available to the team. Therefore, good judgment would be used.

Activity 12: The project's software schedule is derived according to a documented procedure	A	Documented procedure is included within the provided Software Estimation template.	Valid.
Activity 13: The software risks associated with the cost, resource, schedule, and technical aspects of the project are identified, assessed, and documented.	A		Valid.
Activity 14: Plans for the project's software engineering facilities and support tools are prepared	A		Valid.
Activity 15: Software planning data are recorded.	A		Valid.
Measurement 1: Measurements are made and used to determine the status of the software planning activities.	A		Valid.
Verification 1: The activities for software project planning are reviewed with senior management on a periodic basis.	NR		Not Recommended. Performed in Verification 2. <ul style="list-style-type: none"> • These 'reviews' are done within the reviews established elsewhere. • The flat management structure makes this irrelevant.
Verification 2: The activities for software project planning are reviewed with the project manager on both a periodic and event-driven basis.	A		Valid. Risk analysis and project reviews are done periodically.
Verification 3: The software quality assurance group reviews and/or audits the activities and work products for software project planning and reports the results.	T	The software quality assurance group or individual reviews and/or audits the activities and work products for software project planning and reports the results.	Changed to reflect that the software quality assurance group may be one individual.

CMM Elements – Software Project Tracking and Oversight

CMM Element	Code	Rewrite / Reword	Rationale
Goal 1: Actual results and performances are tracked against the software plans.	A		General enough to accept as written.
Goal 2: Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans	A		General enough to accept as written.
Goal 3: Changes to software commitments are agreed to by the affected groups and individuals.	A		General enough to accept as written.
Commitment 1: A project software manager is designated to be responsible for the project's software activities and results.	T	A project manager is designated to be responsible for the project's software activities and results.	Changed to reflect the roles established for the project.
Commitment 2: The project follows a written organizational policy for managing the software project.	T	The project plan is created and modified as the project progresses for tracking purposes.	<ul style="list-style-type: none"> The project policy is the organizational policy. The restatement is boiled down from the description of what 'policy' typically contains. All other functions accounted for by virtue of this being a smaller project. See L2-32.
Ability 1: A software development plan for the software project is documented and approved.	A		Valid. Project plan created according to Project Planning practices noted elsewhere.
Ability 2: The project software manager explicitly assigns responsibility for software work products and activities.	T	The project manager explicitly assigns responsibility (creation, effort, cost, schedule, and budget) for software work products and activities.	<ul style="list-style-type: none"> Changed to reflect roles established for the project. Responsibilities explicitly named for clarity
Ability 3: Adequate resources and funding are provided for tracking the software project.	A		Valid. This will have to be done as project's limited resources allow.
Ability 4: The software managers are trained in managing the technical and personnel aspects of the software project.	E	The project manager has experience in managing the technical and personnel aspects of the software process. Experience may be acquired from previous work / course experience, or may be provided from in-house training, self study, peer study, out-sourced training, or other applicable methods of gaining the previously mentioned experience.	<ul style="list-style-type: none"> Changed to reflect roles established for the project. Training must reflect experience, and the methods needed to gain the training must be flexible since smaller organizations can not afford extensive training (time or costs).

Ability 5: First-line software managers receive orientation in the technical aspects of the software project.	NR		This practice is not relevant. The 'managers' should have a day-to-day working knowledge of the technical aspects of the project, and thus need no special orientation except in the informal situation where colleagues share technical expertise with the entire group or the lead software developer.
Activity 1: A documented software development plan is used for tracking the software activities and communicating status.	A		Valid. Project plan created according to Project Planning practices noted elsewhere.
Activity 2: The project's software development plan is revised according to a documented procedure.	E	The project's software development plan is revised according to a project plan template-documented procedure. As a part of this procedure, reviews / status reviews should be held periodically with the development group. Documented procedure is included within the provided Project Plan template.	<ul style="list-style-type: none"> To save the small organizations time and effort, the procedures should be documented within the context of the project plan template. Reviews should be held every cycle to keep all project members informed.
Activity 3: Software project commitments and changes to commitments made to individuals and groups external to the organizations are reviewed with senior management according to a documented procedure.	T	Software project commitments and changes to commitments made to individuals and groups external to the organization are reviewed with the project manager according to a documented procedure. Documented procedure is included within the provided Review Meeting template.	<ul style="list-style-type: none"> Changes to reflect roles for the project. Template provided within the context of the model for the review meeting.
Activity 4: Approved changes to commitments that affect the software project are communicated to the members of the software engineering group and other software-related groups.	T	Approved changes to commitments that affect the software project are communicated to and negotiated with the individuals responsible for or affected by the changes. Includes changes in staffing, costs, computer resources, and schedules.	<ul style="list-style-type: none"> Types of changes noted for clarity. Reflects idea of less formality and more communication among group members of the smaller group.
Activity 5: The sizes of the software work products (or size of the changes to the software work products) are tracked, and corrective actions are taken as necessary.	O		Some groups may not need size measures kept. Project dependent.
Activity 6: The project's software effort and costs are tracked, and corrective actions are taken as necessary.	E	The project's software staffing, effort and costs are tracked, and corrective actions are taken as necessary.	Term 'staffing' added to clarify and include this CMM intended aspect.

Activity 7: The project's critical computer resources are tracked, and corrective actions are taken as necessary.	A		Valid. Compared against project plan projections to identify trouble areas.
Activity 8: The project's software schedule is tracked, and corrective actions are taken as necessary.	E	The project's software schedule is tracked (including activities, milestones, and other commitments), and corrective actions are taken as necessary. Schedule changes (early <i>or</i> late completion) are evaluated for effects on future schedules.	<ul style="list-style-type: none"> Compared against project plan projections to identify trouble areas. Expounded upon how schedule changes are handled. Included schedule markers for clarity.
Activity 9: Software engineering technical activities are tracked, and corrective actions are taken as necessary.	E	Software engineering technical activities are tracked, and corrective actions are taken as necessary. Releases are evaluated for functionality promised in the project plan. Defects are identified, documented, and tracked to closure.	Reworded to include areas of concern with regard to 'software engineering technical activities'.
Activity 10: The software risks associated with cost, resource, schedule, and technical aspects of the project are tracked.	E	The software risks associated with cost, resource, schedule, and technical aspects of the project are tracked. Risk priorities and contingencies are adjusted as additional information becomes available. Periodic meetings are held to review risks.	<ul style="list-style-type: none"> Additional information added to clarify and include these CMM intended aspects of risk.
Activity 11: Actual measurement data and replanning data for the software project are recorded.	A		Valid. See Project Planning activities.
Activity 12: The software engineering group conducts periodic internal reviews to track technical progress, plans, performance, and issues against the software development plan.	A		Valid. Reviews held per cycle.
Activity 13: Formal reviews to address the accomplishments and results of the software project are conducted at selected project milestones according to a documented procedure.	NR		Not Recommended. During each cycle, reviews will be held as noted in Project Tracking Activity 12. Thus, additional reviews are not needed for the smaller organizations.
Measurement 1: Measurements are made and used to determine the status of the software tracking and oversight activities.	A		Valid. Examples of these measurements are effort, resources expended, and change activity.

Verification 1: The activities for software project tracking and oversight are reviewed with senior management on a periodic basis.	NR		Not Recommended. <ul style="list-style-type: none"> • These 'reviews' are done within the reviews established elsewhere. • The flat management structure makes this irrelevant.
Verification 2: The activities for software project tracking and oversight are reviewed with the project manager on both a periodic and event-driven basis.	NR		Not Recommended. <ul style="list-style-type: none"> • These 'reviews' are done within the reviews established elsewhere. • Management's involvement in day-to-day activities makes this irrelevant.
Verification 3: The software quality assurance group reviews and/or audits the activities and work products for software project tracking and oversight and reports the results.	E	The software quality assurance group or individual reviews and/or audits the activities and work products for software project tracking and oversight and reports the results.	Changed to reflect that the software quality assurance group may be one individual.

CMM Elements – Software Quality Assurance

CMM Element	Code	Rewrite / Reword	Rationale
Goal 1: Software quality assurance activities are planned.	E	The software quality assurance (SQA) activities are planned. The SQA plan should cover the following areas: <ul style="list-style-type: none"> • Functional testing • Load testing • Performance baseline • Back-end database • Scalability • Response time • Different Configuration Combination 	The enumerated areas are the top testing areas for a web site. According to TechWeb, these are the vital areas to test before a site goes live.
Goal 2: Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.	E	Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively. Adherence to standards and requirements can include: <ul style="list-style-type: none"> • To perform functional testing. • To pass the eCommerce load and stress test. • To perform optimization. 	<ul style="list-style-type: none"> • The definition now reflects the desire to verify that an application does what it is supposed to do and performs according to the original design specification. • The goal of the organization is to assess the true scalability of its web application. • The product must meet the testing of all the different configuration combinations and ensure that all the supporting infrastructure works under varying pressures and conditions.
Goal 3: Affected groups and individuals are informed of software quality assurance activities.	T	The project SQA group will have two roles: A SQA manager and the Chief Architect. It will be the responsibility of the SQA group to inform the developers immediately about the software quality assurance activities. The notification can be via an email or a small note so that there is a record of it and yet, little overhead is created in writing large documents.	Due to the small size of the group, two people will compose the SQA group. Since the time duration of the project is comparatively small, the SQA should be done in parallel to the development and the developers should be informed of any SQA activities as soon as possible. The entire team, however, will have an active role in determining quality for the project.
Goal 4: Noncompliance issues that cannot be resolved within the software project are addressed by senior management.	T	The group as a whole resolves any noncompliance issues during regular meeting times.	Since a small company has a relatively flat organizational structure and has no identified senior management, any noncompliance issues should be resolved by discussing these issues within the group.
Commitment 1: The project follows a written organizational policy for implemented software quality assurance (SQA)	E	The project follows an appropriately detailed written project policy for implemented software quality assurance (SQA).	<ul style="list-style-type: none"> • The project policy is the organizational policy. • Members of small organizations should not be subjected to overwhelming amounts of documentation.

Ability 1: A group that is responsible for coordinating and implementing SQA for the project (i.e., the SQA group) exists.	E	A group that is responsible for coordinating and implementing SQA for the project exists. This group is a small group consisting of two people with multiple part-time project roles: one SQA manager and a chief architect.	A group can vary from a single individual assigned to roles part time to several part-time individuals assigned from different departments, to several individuals dedicated full time. Therefore, the definition is expanded to list the part-time roles.
Ability 2: Adequate resources and funding are provided for performing the SQA activities.	A		Valid.
Ability 3: Members of the SQA group are trained to perform their SQA activities.	E	Members of the SQA group are trained to perform their SQA activities. Examples of this training includes: <ul style="list-style-type: none"> • Software engineering skills. • Familiarity with the technology, the business, and the customer requirements. • (For testing on the Web tier) An understanding of different web browsers. The tester must also understand ASP, HTML, DHTML, Java, and VBScript. • (For testing on the middle tier) Knowledge of business logic. This includes Web server applications, tax and shipping calculations, discount, up-sell and down-sell mechanisms, and shopping cart functionalities. • (For testing on the data tier) Proficiency with the database software, Microsoft SQL Server commands, and file and storage systems. 	This definition was extended to include area in which training should be completed. These areas are provided by Biraj Rath, Author, NIIC (USA), Inc. According to the author, an eCommerce system consists of three tiers – Web, middle, and data.
Ability 4: The members of the software project receive orientation on the role, responsibilities, authority, and value of the SQA group.	T	The members of the team receive orientation on the role, responsibilities, authority, and value of the SQA group by informal discussions among themselves and during regular meetings.	Since the group is a small group, there is little need for the original level of formalization.
Activity 1: A SQA plan is prepared for the software project according to a documented procedure.	E	A SQA plan is prepared for the software project according to a documented procedure. Documented procedure is included within the provided Quality Assurance template.	Template is provided by virtue of the model.

Activity 2: The SQA group's activities are performed in accordance with the SQA plans.	A		Valid.
Activity 3: The SQA group participates in the preparation and review of the project's software development plan, standards, and procedures.	A		Valid.
Activity 4: The SQA group reviews the software engineering activities to verify compliance.	A		Valid.
Activity 5: The SQA group audits designated software work products to verify compliance.	A		Valid.
Activity 6: The SQA group periodically reports the results of its activities to the software engineering group.	A		Valid.
Activity 7: Deviations identified in the software activities and software work products are documented and handled according to a documented procedure.	A	Documented procedure is included within the provided Quality Assurance template.	Valid.
Activity 8: The SQA group conducts periodic reviews of its activities and findings with the customer's SQA personnel, as appropriate.	T	The SQA group documents its activities and reviews its activities with the customer at the customer's request.	Time is limited and reviews should be done on an as-needed or as-requested basis.
Measurement 1: Measurements are made and used to determine the cost and schedule status of the SQA activities.	A		Valid. Examples of these measurements are effort, resources expended, and change activity.
Verification 1: The SQA activities are reviewed with senior management on a periodic basis.	NR		Not Recommended. Performed in Verification 2. <ul style="list-style-type: none"> • These 'reviews' are done within the reviews established elsewhere. • The flat management structure makes this irrelevant.
Verification 2: The SQA activities are reviewed with the project manager on both a periodic and event-driven basis.	A		Valid.
Verification 3: Experts independent of the SQA group periodically review the activities and software work products of the project's SQA group.	T	The team reviews the activities and the software work products of the project's SQA group in their regular meetings.	Since the organization is small, its very difficult to have a separate group independent of SQA group and reviewing the SQA activities. So the whole team can review the SQA activities in their periodic meetings.

CMM Elements – Software Configuration Management

CMM Element	Code	Rewrite / Reword	Rationale
Goal 1: Software configuration management activities are planned.	E	Team members need to plan for their software configuration management activities.	Expanded to make the goal more specific.
Goal 2: Selected software work products are identified, controlled, and available.	A		General enough to accept as written.

Goal 3: Changes to identified software work products are controlled.	A		General enough to accept as written.
Goal 4: Affected groups and individuals are informed of the status and content of software baselines.	A		General enough to accept as written.
Ability 1: A board having the authority for managing the project's software baselines (i.e., a software configuration control board – SCCB) exists or is established.	T	The project manager can manage the project's baseline, which includes key files that should not be modified. This process prevents the team members from having to go through a more formal change control procedure.	Too general for the project. Reduces the amount of formalization required for the small team.
Ability 2: A group that is responsible for coordinating and implementing software configuration management for the project (i.e., the SCM group) exists.	T	A group or individual (such as the project manager) that is responsible for coordinating and implementing software configuration management for the project is identified.	Too general for the project. Changed to reflect the small team structure and to assign this responsibility.
Ability 3: Adequate resources and funding are provided for performing the SCM activities.	NR		Redundant. Group established as directed in Ability 2. Tools may not be available to all teams.
Ability 4: Members of the SCM group are trained in the objectives, procedures, and methods for performing their SCM activities.	T	Members of the SCM group or the SCM individual are/is trained in the objectives, procedure, and methods for performing their/his/her SCM activities. Training may be acquired from previous work / course experience, or may be provided from in-house training, self study, peer study, out-sourced training, or other applicable methods of gaining the previously mentioned experience.	<ul style="list-style-type: none"> • Changed to reflect that the software configuration management group may be one individual. • For a small company or group, there may be no formal training. Training must reflect experience, and the methods needed to gain the training must be flexible since smaller organizations can't afford extensive training (time or costs).

Ability 5: Members of the software engineering group and other software-related groups are trained to perform their SCM activities.	T	Members of the software engineering group and other software-related groups are trained to perform their SCM activities. Training may be acquired from previous work / course experience, or may be provided from in-house training, self study, peer study, out-sourced training, or other applicable methods of gaining the previously mentioned experience.	For a small company or group, there may be no formal training. Training must reflect experience, and the methods needed to gain the training must be flexible since smaller organizations can't afford extensive training (time or costs).
Activity 1: A SCM plan is prepared for each software project according to a documented procedure.	T	There is a SCM plan that is used by the project group. Documented procedure is included within the provided Configuration Management template.	There is one SCM plan that is documented for the project and used by the organization.
Activity 2: A documented and approved SCM plan is used as the basis for performing the SCM activities.	A		Valid.
Activity 3: A configuration management library system is established as a repository for the software baselines.	A		Valid and necessary.
Activity 4: The software work products to be placed under configuration management are identified.	A		Valid. The project members must be clear about what to place under SCM.
Activity 5: The Change requests and problem reports for all configuration items/units are initiated, recorded, reviewed, approved, and tracked according to a documented procedure.	A	Documented procedure is included within the provided Configuration Management template.	Valid.
Activity 6: Changes to baselines are controlled according to a documented procedure.	A	Documented procedure is included within the provided Configuration Management template.	Valid and important.
Activity 7: Products from the software baseline library are created and their release is controlled according to a documented procedure.	A	Documented procedure is included within the provided Configuration Management template.	Valid and important.
Activity 8: The status of configuration items/units is recorded according to a documented procedure.	A	Documented procedure is included within the provided Configuration Management template.	Valid

Activity 9: Standard reports documenting the SCM activities and the contents of the software baseline are developed and made available to affected groups and individuals.	E	Standard reports documenting the SCM activities and the contents of the software baseline are developed and made available to affected groups and individuals. The reports are broadcast – possibly web-based.	Make explicit how the team members will receive the SCM reports.
Activity 10: Software baseline audits are conducted according to a documented procedure.	A	Documented procedure is included within the provided Configuration Management template.	Valid. This is one of the most important steps in the project.
Measurement 1: Measurements are made and used to determine the status of the SCM activities.	T	The project team members will have regular meetings to check the status of the SCM activities.	Unexpanded definition was too general.
Verification 1: The SCM activities are reviewed with senior management on a periodic basis.	NR		Not Recommended. Performed in Verification 2. <ul style="list-style-type: none"> • These ‘reviews’ are done within the reviews established elsewhere. • The flat management structure makes this irrelevant.
Verification 2: The SCM activities are reviewed with the project manager on both a periodic and event-driven basis.	A		Valid.
Verification 3: The SCM group periodically audits software baselines to verify that they conform to the documentation that defines them.	A		Valid and important.
Verification 4: The software quality assurance group reviews and/or audits the activities and work products for SCM and reports the results.	A		Valid.