

Customer Best Practices - Post Mortem Phase

November 2005 - Pragmatic Software Newsletters

According to the Standish Group, the primary contributor for project success (projects delivered on-time and on-budget) is great customer involvement. Knowing that, it makes sense to define best practices for dealing with customers. In the coming months, we will look at best practices for customer involvement that deal with all phases of the software life cycle:

1. **Planning** - Getting the customers involved in the planning process is critical. Gathering, analyzing and evaluating requirements is key to ensuring that your project meets the needs of your customer. [More Information...](#)
2. **Design** - Good designs are key to ensuring that changes the client requests require minimal changes to your framework. [More Information...](#)
3. **Iterative Coding and Testing** - Customers play a major role in Quality Assurance and User Acceptance testing prior to production releases. [More Information...](#)
4. **Production** - Having good release management and support ticket management procedures in place will ensure quality releases and great customer support as issues arise. [More Information...](#)
5. **Post Mortem** - Upon completion of projects, it is of paramount importance to involve your customers in the project review process. Post Mortem reviews can aid in discovering and document common issues that arose during the project and allows for others to better plan for those risks in upcoming projects.

Post Mortem Phase

Once your software is in production, customer interaction is just as important as it was during the development phase. Below are some best practices to aid project management in the Post Mortem phase:

1. **Plan Your Post Mortem Review** - Upon completion of a project, the Project Manager should conduct a "**Post Mortem**" review. This is where the Project Manager invites all the major players of the team (Analysts, Lead Programmers, Quality Assurance Leaders, Production Support Leaders, Customer, etc) to a meeting to review the successes and failures of the project.
2. **Require Team Participation** - Ask the attendees to bring a list of 2 items that were done well during the project and 2 things that could be improved upon.
3. **Hold the Post Mortem Review Meeting** - Go around the table and have each person to discuss the 4 items they brought to the meeting. Keep track of how many duplicate items you get from each team member. At the end of the round table discussion of items, you should have a count of the most popular items that were done well and the most agreed upon items that need improvement. Discuss the top 10 success items and the top 10 items that need improvement.
4. **List Items Done Well and Things Needing Improvement** - Upon listing of the 10 success and improvement items, discuss specific things that can be done to avoid the items that need improvement upon the next release. If some items need more investigation, assign specific

individuals to finding solutions.

5. **Create a Post Mortem Report** - The best way to keep this information organized is to create a "Post Mortem" report, where you document your findings. Send the Post Mortem report to all team members. Before team members embark on their next project, make sure they review the Post Mortem report from the prior project to gain insight from the prior project. We have created a template that you can use for the document, download it by clicking [here](#).

Helpful Templates

Below are some helpful templates to aid you in developing software solutions on-time and on-budget:

- **Project Management Guidelines** - <http://www.PragmaticSW.com/Pragmatic/Templates/ProjectMgtGuidelines.rtf>
- **Functional Specifications** - <http://www.PragmaticSW.com/Pragmatic/Templates/FunctionalSpec.rtf>
- **Architectural Overview** - <http://www.PragmaticSW.com/Pragmatic/Templates/ArchitectureOverview.rtf>
- **Detailed Design** - <http://www.PragmaticSW.com/Pragmatic/Templates/DetailedDesign.rtf>
- **Strategic Planning Document** - <http://www.PragmaticSW.com/Pragmatic/Templates/StrategicPlanning.rtf>
- **Test Design** - <http://www.PragmaticSW.com/Pragmatic/Templates/TestDesign.rtf>
- **Risk Assessment** - <http://www.PragmaticSW.com/Pragmatic/Templates/Risk%20Assessment.rtf>
- **Weekly Status** - <http://www.PragmaticSW.com/Pragmatic/Templates/WeeklyStatusRpt.rtf>
- **User Acceptance Test Release Report** - <http://www.PragmaticSW.com/Pragmatic/Templates/UATRelease.rtf>
- **Post Mortem Report** - <http://www.PragmaticSW.com/Pragmatic/Templates/PostMortem.rtf>
- **All Templates** - <http://www.PragmaticSW.com/Templates.htm>
- **Prior Newsletters** - <http://www.PragmaticSW.com/Newsletters.htm>
- **Software Planner** - <http://www.SoftwarePlanner.com>
- **Defect Tracker** - <http://www.DefectTracker.com>
- **Remoteus (Remote Desktop Sharing)** - <http://www.PragmaticSW.com/Remoteus.asp>

About the Author

Steve Miller is the President of **Pragmatic Software** (<http://www.PragmaticSW.com>). With over 20 years of experience, Steve has extensive knowledge in project management, software architecture and test design. Steve publishes a monthly newsletter for companies that design and develop software. You can read other newsletters at <http://www.PragmaticSW.com/Newsletters.htm>. Steve's email is steve.miller@PragmaticSW.com.

Pragmatic Software Co., Inc.
1745 Shea Center Drive
Suite 400
Highlands Ranch, CO 80129

Phone: 720.344.4846
Fax: 720.344.4847
Web site: <http://www.PragmaticSW.com>
E-mail: info@PragmaticSW.com