

Customer Best Practices - Planning Phase

June 2005 - Pragmatic Software Newsletters

According to the Standish Group (1994), 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.
2. **Design** - Good designs are key to ensuring that changes the client requests require minimal changes to your framework.
3. **Iterative Coding and Testing** - Customers play a major role in Quality Assurance and User Acceptance testing prior to production releases.
4. **Production** - Having good release management and support ticket management procedures in place will ensure quality releases and great customer support as issues arise.
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.

Planning Phase

During the Planning phase, the project rules and procedures are set and requirements are gathered. Below are some best practices to aid project management during the planning phase:

- **Identify Customer Sponsor** - Every project must have a customer sponsor. This is the person that is responsible for making key decisions. The sponsor must be dedicated to the project and must be evaluated on the project success, so that they have "skin in the game".
- **Manage Customer Expectations** - According to the Standish Group (1994), 10% of projects are cancelled due to unrealistic expectations. Customers don't always understand the length of time it takes to deliver features, nor should they be expected to. However, customers do have features they need and features they want. When reviewing requirements with customers, it is important to communicate to them the relative effort (easy, moderate, difficult) and different approaches to solve the same problem. Many times, the client may have a predetermined way to solve a problem (which may be effort intensive), but with more understanding of the problem, alternative ways to solve the problem may be found.
- **Requirement Gathering Best Practices** - Too often, requirements are verbally communicated and agreed upon. For projects to be successful, requirements must be written, analyzed, and agreed upon in WRITING. This helps in many ways. First, it forces the business analyst and the customer to fully define the requirement and agree on it's implementation. Second, it forces the development team to present several design possibilities, each with different levels of effort. Third, it forces the customer to prioritize features that are most important to them, allowing the team to deliver the solution in iterations, speeding up the development cycle. Last, having approval from the client shows that the client has agreed to the features and the design. For optimal requirements management, it is best to use on-line tools to track and provide the approval workflow for requirements. [Example of using Software Planner to track customer requirements](#)
- **Life Cycle Best Practices** - Upon requirements gathering, it is important for the customer to "stack rank" their requirements. This simply means that the customer evaluates each requirement and places an importance on each requirement (This Release, Next Release, Future Release). By doing

this, projects can be shortened by implementing them in iterations. For example, release 1.0 will contain just the requirements needed in the initial release. By implementing a smaller number of initial requirements, the development life cycle is shortened and issues are uncovered more quickly than waiting on an extended life cycle. It is important to move to the next release (1.1) upon implementation, so the client continues to see iterations of the software moved to production. [More information about Requirements scrubbing](#)

- **Risk Management Best Practices** - There are many customer related risks in a project. These include the fact that some customers don't understand what they want. Some won't commit to written requirements. Some are not easy to contact. Some are slow to make decisions. The list goes on. The best way to mitigate these risks are to plan for them. [More information on Risk Management](#)

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