

Project Retrospective Questions – used to provoke thought and discussion...

- **General Questions:**

- Are you proud of the project deliverables? If yes, what's good? If not, what's not so good?
- What was the single most frustrating part of the project?
- How would you do things differently next time to avoid this frustration?
- What was the most gratifying or professionally satisfying part of the project?
- Which methods or processes worked particularly well?
- Which methods or processes were difficult or frustrating to use?
- If you could change anything about the project, what would you change?
- What was the original Development schedule - Planned and Actual
- What was the original QA schedule - Planned and Actual

- **Project Plan**

- How accurate were original estimates of size and effort? What did we over or under estimate?
- How could we improve estimates of size and effort to make them more accurate?
- Did we have the right people assigned to all project roles? (Consider subject matter expertise, technical contributions, management, review and approval, and other key roles) If not, how can we make sure that we get the right people the next time?
- Were there early warning signs of problems that occurred later in the project? How should we have reacted to these signs? How can we be sure to notice these warning signs next time?
- Were constraints, limitations, and requirements made clear to all developers and QA staff from the beginning? If not, how could we have improved this?
- Were all team/stakeholder roles and responsibilities clearly delineated and communicated? If not, how could we have improved these?
- Were the deliverables specifications, milestones, and specific schedule elements/dates clearly communicated? If not, how could we improve this?

- **Software Requirements Spec (SRS)**

- Were you proud of the SRS? If not, how could we have improved it?
- Did all the important project players have input into the SRS? If not, who were we missing and how can we assure their involvement next time?

- Did those who reviewed the SRS provide timely and meaningful input? If not, how could we have improved their involvement and the quality of their contributions?
- How could we have improved our work process for creating SRS?
- **Design and Code:**
 - Were you proud of the design for this project? If not, how could we have improved?
 - Did all the important project players have input into the design? If not, who were we missing and how can we assure their involvement next time?
 - Did those who reviewed the design provide timely and meaningful input? If not, how could we have improved their involvement and the quality of their contributions?
 - How could we have improved our work process for creating design documents?
 - Were you proud of the code developed for this project? If not, how could we have improved?
 - Were code reviews planned for this project? Were planned code reviews held? If yes, were they effective? If not, why not? What could be done to improve effectiveness of code reviews?
 - Did those who reviewed the code provide timely and meaningful input? If not, how could we have improved their involvement and the quality of their contributions?
 - Were coding standards used for this project? If not, why not? If yes, were they followed and were they effective?
- **Test and Release**
 - Did the QA Team have sufficient domain knowledge? If not, how could we increase domain knowledge of QA staff in the future?
 - Did the QA Team prepare a Test Plan? Was the Test Plan an accurate representation of the testing work that was required? Did the Test Plan include test estimates and schedules? If yes, were the estimates and schedules accurate? If the estimates and schedules were not accurate, what could be done to improve?
 - Did the test environment support the overall testing effort? Were there any environmental issues that impacted testing? If yes, what were they and how could they be avoided next time?
 - Were bug reports submitted by QA sufficiently detailed so that developers were able to effectively diagnose problems without additional information?
 - Were any testing measures used during the test phase? What were they? Were they effective in accurately representing test status/progress? How to improve?
 - How much regression testing was planned and how much was actually performed?
 - How many bugs were found as a result of regression testing?
 - How many known bugs were left in the released product?

- How many defects were reported by QA?
- How many tests were developed by QA?
- How many complete QA testing cycles were executed?