Software Engineering

Team Roles and Group Work

Team Roles

Each member of each group plays an important team role.

Important team roles include:

- Customer Interface Manager
- Design Manager
- Implementation Manager
- Test Manager
- Planning and Quality Manager

Customer Interface Manger

Customer interface manger is responsible for (1) team's relationships with its customer(s), (2) clearing up ambiguities in requirements specification. Some questions that the customer interface manager addresses are:

- Are we being responsive to customer requests?
- Is quality of requirements documentation sufficient to guide development?
- Do all team members understand requirements?

Design Manager

Responsible for quality of team's design documentation. Tracks that design documentation standards are being met and recorded. Some questions that design manager addresses are:

- Do all team members understand how to use design methods (presented in class)?
- Is the team's design work of high quality?
- Does software design consider future product evolution?
- Is design properly documented?
- Is each team member contributing to design?

Implementation Manager

Responsible for quality of team's implementation. Some questions implementation manager addresses are:

- Are all team members competent in Java? If not, what remedial actions are recommended?
- Is design documentation clear enough to begin implementation?
- Are all team members following Bob's Concise Coding Conventions?
- Are team members taking advantage of shared and or reusable code where possible?

Test Manager

Responsibilities for quality of team's testing and test-related work. Some questions the test manager addresses are:

- Are test plans produced when the process needs them?
- Do all classes have unit tests defined? Are all methods tested by unit tests?
- Are sub-system tests defined?
- Are test plans complete and thorough?
- Does each team member understand how to produce tests? If not, what remedial action is recommended?
- What is team's integration test strategy?

Planning and Quality Manager

Responsible for team's plans, reporting, plan status. Some questions support manager addresses are:

- Is team meeting often enough?
- Is team following minutes protocol?
- Does team want to use version control?
- Are code inspections being carried out?
- Is implementation properly documented using doxygen?
- Do quality of all classes and sub-systems meet team's quality standards? If not, what is recommended to fix this.
- What is our plan in case member A does not contribute?

Teamwork and Management

- In general, each member of team assumes a management role
- Each team member contributes to design, implementation, testing, and documentation
- Each team member contributes to all phases of software life-cycle
- E.g., Implementation manger ensures that each team member contributes properly to implementation
- Problems and remedial actions are documented in group report

Team Work and Risk Management

- ~90-100 enter year 1
- ~70-80 enter year 2
- ~50-60 enter year 3
- ~35-40 graduate

What are the implications?

Team Work and Risk Management (2)

- Some students may not contribute to team work
- Each team is required to complete entire assignment
- **Expect, anticipate, and prepare for non-contributors**
- **Misconception:** non-contributors “win”. Non-contributors do poorly in all modules.

Advantages of Smaller Teams

Despite disadvantages of smaller team, there are significant advantages, including:

- Planning, communication, and coordination is simplified. The larger the team, the more complex.
- Meetings are easier to plan
- Responsibilities are easier to track, less diffusion of responsibility
- A small dedicated team is more effective than a large not-so-dedicated team
- Remaining team members learn more
- Project seems large, but could be done by 1, hard-worker

Identifying Potential Non-Contributors

- Class absence
- Tutorial absence
- Group meeting absence
- Did they hand in their assignment? E.g., A1 and/or A2 from another module?

- Group meeting absence is documented in minutes of meeting protocol
- Non-contributors are documented in group report.

Handling Non-Contributors: Advice

- Expect, anticipate, and plan for non-contributors
- Set and use internal group deadlines
- Re-assign management role to someone else
- Re-allocate responsibilities, e.g., divide and re-assign classes amongst remaining group members
- Avoid use of email for conflict resolution
- Email amplifies conflict
- See, “The Secret Cause of Flame Wars”

http://www.wired.com/science/discoveries/news/2006/02/70179

- 50% of email is misinterpreted.

Acknowledgements

Thanks to:

- All those that have worked with Bob in a team.

Recommended Reading:

- **Introduction to the Team Software Process** (SEI Series in Software Engineering) **Chapter 8 Team-Member Roles**, by Watts S. Humphrey, Addison-Wesley