Project Teams

CSE 403, Winter 2003
Software Engineering

http://www.cs.washington.edu/education/courses/403/03wi/
Readings and References

• References
  » *Rapid Development*, Steve McConnell
    • Chapter 4, Software Development Fundamentals
    • Chapter 12, Teamwork
    • Chapter 13, Team Structure
  » *The Mythical Man-Month*, Brooks
    • Chapter 3, The Surgical Team
Issues

• Most projects need teams of people for success
  » many skills required
  » time is limited

• Communication requirements increase with increasing numbers of people
  » everybody to everybody $\rightarrow \frac{n(n-1)}{2}$
  » even just somebody to everybody $\rightarrow n-1$

• Every effort at communication is a chance for miscommunication
Take risks, but manage them

• The need for many people exposes us to risk
• What are the tools that we use to manage it?
  » Good, well-known product definition
  » Planning and organization
  » Monitoring and direction as needed
    • we have a plan
    • we’ll work to the plan and monitor our performance
    • we’ll change the plan if we need to
  » Transparency - no secrets
Management Fundamentals: Planning

- “We have a plan”
- Estimation and scheduling
- How many people with what skills, when?
- Organization of the team
- Lifecycle events
- Managing the risks
- Strategic decisions
  » for example, build or buy decisions
Management Fundamentals: Tracking

• “We’ll work to the plan and monitor our performance”

• Some tools
  » Task lists, status meetings, status reports, milestone reviews, budget reviews

• Management by walking around

• “We’ll change the plan if we need to”
  » Can only be effective if all the facts are known
Management Fundamentals: Measurement

- Help validate comparisons between this project and previous/future work
- Basic measurements of the code
  - Non Commenting Source Statements (NCSS)
  - Number of modules, packages
- Project build: success and frequency
- Change and defect data
- Be careful: we optimize to the metric in use
Teamwork and Organization

- Teams of people can achieve big goals
  - Panama Canal, man in space, Mt. Everest
  - but it ain’t easy
- The members of a good team
  - know what the goals of the team are
  - know what their own task responsibilities are
  - have the tools they need to accomplish their tasks
  - have reason to believe that the team will succeed
Results-driven Structure

• Roles are clear within the team
  » Each person is accountable for their work

• Effective communication system
  » Change management, schedule, tracking, decisions

• Monitor individual performance
  » Who is doing what, are we getting the work done?

• Fact based decisions
  » Focus on the facts, not the personalities
Team Models

- **Business Team**
  - peer group headed by technical lead

- **Chief Programmer Team**
  - Brooks’ surgical team - surgeon plus support

- **Skunkworks team**
  - Black box, creative but maybe ad-hoc

- **Feature team, Search-and-Rescue team, SWAT team, Professional Athletic team, Theater team, etc, etc**
Brooks: Surgical Team

- Surgeon
  - Administrator
    - Secretary
  - Editor
    - Secretary
  - Copilot
    - Programming Clerk
    - Toolsmith
    - Tester
    - Language Lawyer
zombie.com Dev Team
Managers and Technical Leads

- No matter what you call the structure, teams usually have:
  - several “regular” developers
  - a technical lead developer
  - a project management function, assigned to:
    - the technical lead
    - a separate project manager
    - the group supervisor
    - ...

Adapted from McConnell, from Boddie
Responsibility

• Take individual responsibility for your tasks
• In order to succeed, the team must
  » Decide what the tasks are
    • task content, interfaces, order, ...
  » Clearly define who is going to do each task
  » “Sign up” to do them
  » Let ‘er rip
• Communicate as you go