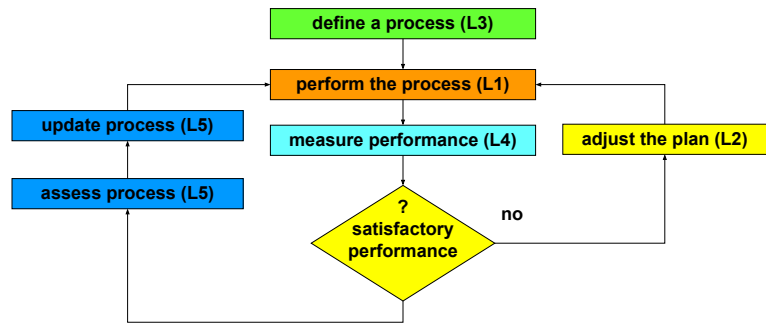


Basic Project Skills

- Project Post-Mortems
 - motivations and their use in this course
 - examine the Diablo II post mortem
- Project Planning
 - work break-down, dependency analysis
 - risk assessment and mitigation
 - estimation and scheduling
- Version Control
 - motivation and its use in this course
 - git and GitHub

Process Improvement 1A (the Capability Maturity Model)



Post Mortems/Retrospectives

- complex skills must be developed/refined
- every project is a learning opportunity
 - improve our skills with existing methodology
 - try new techniques, confront new problems
 - learn from our mistakes
- post-mortems are pro-active learning
 - reflect and discuss as a group
 - what worked and what didn't work?
 - what should we do differently next time?

Post Mortems/Retrospectives

- there are many techniques and formats
- all have the same basic requirements:
 - honesty: willingness to recognize mistakes
 - introspection: willingness to analyze them
 - safety: no penalties for admitting mistakes
- in this course
 - you will develop post mortem processes
 - they will help you learn from the projects
 - you will learn to use them as a learning tool

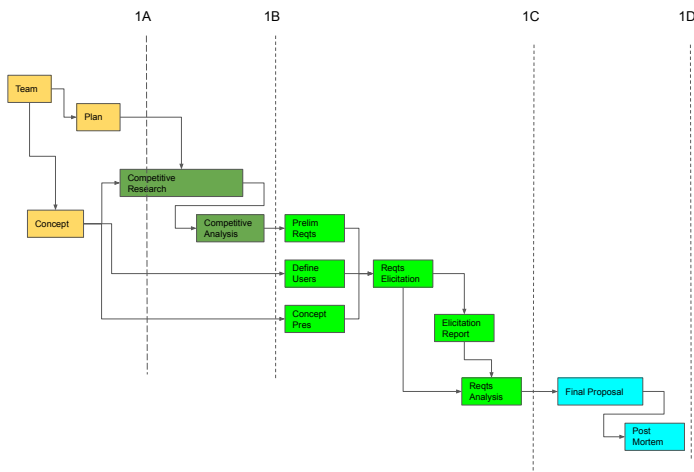
Diablo II Post Mortem

- Gamasutra article by the D2 design lead
 - brief history leading up to this project
 - born from D1 wish-lists and complaints
 - organization, methodology, tools
 - discussion of things that worked
 - D2 is Diablo, hiring/development process, new skill trees, massive Q/A effort, world-wide release
 - discussion of things that didn't work
 - the old battle.net, # of users, non-state-of-the-art graphics, weak tools, a new game-save feature

Common Themes (prior PMs)

- problems
 - communication and pace management
 - insufficient tool/API/service mine-sweeping
 - insufficient planning for impl/integration
- successes
 - shared board/screens for brainstorming
 - timezone/schedule aware mtgs and pairing
 - clarity gained from competitive research
 - social networks for requirements gathering
 - team reviews of all deliverables
 - reviewing other projects improved their own
 - benefits of reviews, pair-programming, TDD

Project 1: Concept and Proposal



Teams and Planning

- Share your thoughts on
 - general product concept
 - relevant experiences and strengths
 - implementation tools & technologies
- Consider
 - how well it will meet P2-4 requirements
 - team skill composition (and timezones)
 - is this the right team for you
- wrap-up at 14:25

Basic Project Skills

8

Plan Check List

- task descriptions:
 - everyone believes the list to be complete
 - all task owners understand:
 - what their tasks mean, and how to do them
 - what they will deliver, when, in what form
- risk exposures and mitigation:
 - everyone agrees w/assessments & plans
- schedule:
 - achievable w/adequate room for problems

Basic Project Skills

9

Regular Team Status Meetings

- a good meeting
 - has clear goals ... and accomplishes them
 - is brief ... avoids tangents and rat-holes
- Daily SCRUM or XP *Stand-Up*
 - what have you completed since last meeting?
 - what do you plan to do by the next meeting?
 - what is getting in your way?
 - if there is a problem, assign people to work it

Basic Project Skills

10

version control: GitHub

- version control is critical to any s/w project
 - track changes, previous versions, backup
 - work products come from version control 🍌
- centralized control is the old paradigm
 - new projects are distributed collaborations
 - distributed version control is more powerful
 - “git” is today’s dominant open-source tool
- all code work products should be on GitHub
 - suggest you also use issues and project board

Basic Project Skills

11

For next lecture

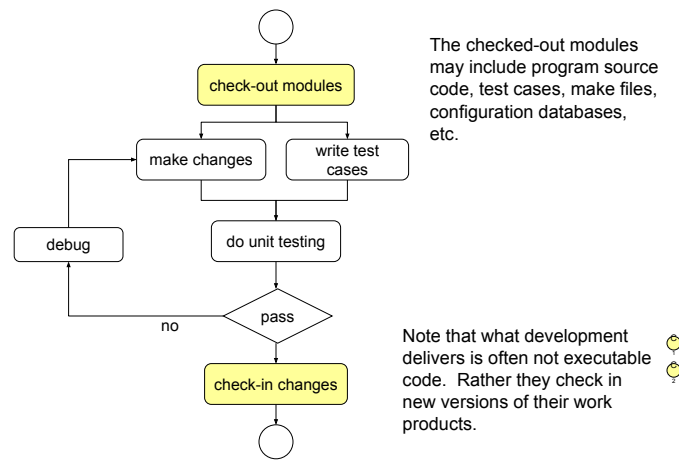
- McConnell 3-3.2
 - the importance of having a plan
- McConnell 34.2
 - process models
- Kampe: S/W Process Models
 - introduction to project phases and models
- Boehm: Spiral Development
 - iterative development: what, why, and how
- Ambler: “Big Requirements Up Front”
 - the agile critique of the classic Waterfall

Basic Project Skills

12

Back up slides

software development process



Basic Project Skills

14

the laws of version control

- All of our work products are versioned
 - we can tell what version we are dealing with
- All official changes are tracked
 - we know exactly what changes were made
 - we know who made each change, when, why
- We can reconstruct any version at any time
 - not just the current version, any prior version
- Files exist in multiple parallel branches
 - each of which has its most current version

Basic Project Skills

15

version control procedures

- never deliver a work product directly
 - rather, deliver a version-controlled file
 - ensures proper recording of all work
- build from the version controlled files
 - extract specific (or current default) versions
- associate versions with deliverables
 - release has a list of all versions used to build it
 - test/bug reports associated w/specific releases
 - bug fixes are associated w/new file versions
 - work product approvals specify a version

Basic Project Skills

16

change control

- who can change what, where, when?
- sometimes, some change is good
 - it represent progress as work is completed
 - such changes should be facilitated
- sometimes, some change is bad
 - changes can be disruptive to the product
 - we need processes to detect & prevent these
- hopefully these processes are adaptive
 - adjusting the burden in response to the risk

Basic Project Skills

17




change control mechanisms

- may be performed by version control tools
 - may control who can modify which files
 - may notify interested parties of changes
 - these features are usually configurable
- may be managed by human processes
 - publication and objection
 - designated component reviewers
 - change control boards
- should have mechanism/policy separation

Basic Project Skills

18

Conflicting Updates

- People may work at cross purposes
 - independent changes to same module
 - different understandings of how things work
- File Change Notifications
 - subscribe to notifications for selected files 
- File Locking
 - at check out time, or independently 
 - locks can be advisory or enforced 
- Change merge assistance
 - automatic difference analysis, proposed merge