4156 Thursday 9/19/13 page 1

team ~~definition~~ composition assignment
due today 5pm

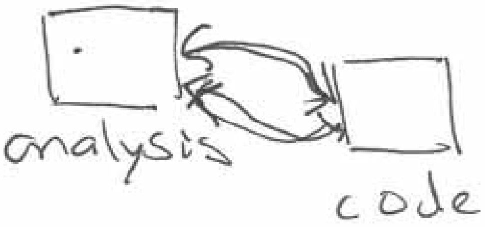
does anyone not have a four person team?

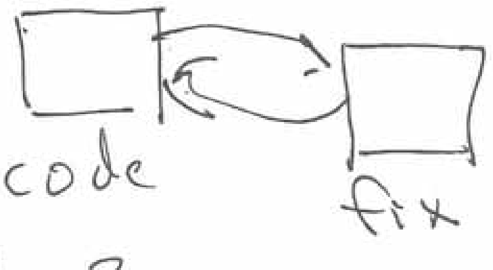
check current enrollment in 5502
~~50~~ 49

introduce next assignment
project proposal
due Thursday 9/26/13 5pm

software process

every software process includes at least two steps





which is better?

internal use personal use

system requirement → software requirements → analysis →

→ show courseworks figure

program design → coding → testing → operations

more steps needed for delivery to customer

what really happens in practice?

ideal case - backlinks <u>only</u> to previous step not to more remote step(s)

→ show courseworks figure

change process scoped to ~~reasonable~~ manageable limits with each successive step

but in practice testing comes too late & may find insurmountable problems that requires return to origin → show courseworks figure

4156 Thursday 9/19/13 page 3

note the insurmountable problems described
in paper probably would not be big
issues today, but there can be other
big issues

→ timing, storage, I/O transfers
probably only big problem today
for embedded devices

return to origin may mean 100%
overrun in schedule &/or costs

#1 solution in paper is to design for those
steps early & then treat as constraint
(part of requirements) ~~fundamental~~

#2 and also produce huge amount of
documentation at every step in process

#3 final part of solution is
"do it twice" - not just a prototype
but a small scale model of system

#4 special emphasis on testing
specialist testers independent
from developers, visual code inspection
coverage of every logic path

#5 involve the customer throughout
commits at various earlier points
prior to delivery

results in very complicated
~~exten~~ (& somewhat convoluted)
version of waterfall
→ show courseworks figure

but basically had gist of many
~~&~~ modern ideas on software development
customer usually wants sw not docs → (except documentation part) exceptions military
Unfortunately, many readers took
away only the basic version of
waterfall - and understood it as
recommended rather than the strawman
that didn't work

so eventually other ~~ps~~ software
processes ~~were develop~~ became popular

mostly variants on
- incremental delivery

OR - iterative ~~development~~ planning
often combined
- iterations produce increments

4156 Thursday 9/19/13 Page 5

pure incremental

entire requirements & design

cycle here only

then code ~~&~~ test, deliver increment

more realistic

agile = very short iterations

cycle

partial requirements
partial design
then corresponding partial code, tests deliver

pure iterative would build an entire new version of system each time reusing very little if any previous work

"extreme programming" is an <u>extreme</u> example of an iterative incremental approach

- lots of specific rules about how to do each step

→ show courseworks links

let's talk about the 4156 process

~~variant of waterfall, but employing some agile and techniques e.g. pair programming~~

variant of waterfall, since we have time for only one iteration, but employing some agile techniques

there will be a team milestone nearly every week

a few weeks there will be individual assignments ~~or exams~~ instead

next assignment is proposal
after that is plan
- firm up requirements
- use JIRA (optional tutorial)

~~then design,~~
~~then testing~~
~~then design,~~

then design,
coding, testing,
more testing,
lot of testing

tutorial Friday?