Thursday 10/9/14

will cover project planning quickly —
already covered some on Tuesday —
and then do mini-review in preparation
for next Tuesday's review session

start with either user stories or use cases,
we'll do user stories here for simplicity

for each user story, estimate how long
it will take one pair to do
& add them all up to see how long
project will take

time can be real units (½ day, 1 day)
or relative time ("points")

points can be converted to days, if
done similar project in past, use
actual time spent per point then

everyone on team (or every pair) does
own estimates, then compare
→ consensus on new estimates
convergence & confidence

if there are outliers & wide spreads, maybe
problems w/ assumptions — go to customer

if less than half day combine

if more than ~3 days need to break up
   for 1-week iterations
   ~15 days for 1-month iterations

"AND" rule - do this AND do that
   separate into this & that

only 20 working days per month
need to talk to customer about how
long whole project will take - will
almost certainly be far too long

so customer needs to prioritize, focus
on baseline functionality to fit

can we add more people?
communication & training time

          etc       diminishing
                                        returns

let's say we have right number of
people to do milestone 1.0 in
N months, assuming 20 days/mo
are we good to go?

Thursday ~~10th~~ 10/9/14

will those 20 days/month all be
* productive * days?
  - people get sick or take personal days
  - vendor for mission-critical software
    issues a critical security path that
    must be installed on everyone's PC
  - there's an accident on the Cross
    Bronx backing up NJ traffic
    for 3 hours

and we don't know how accurate our
estimates were anyway

"project velocity" - how much productive
  time, on average, we really get
  0.7 (70%) is good starting point
  - so if milestone 1.0 scheduled for
    N months, more likely N/0.7

so now we're back to too long again,
need to take productivity into
consideration when talk to customers
  about prioritizing

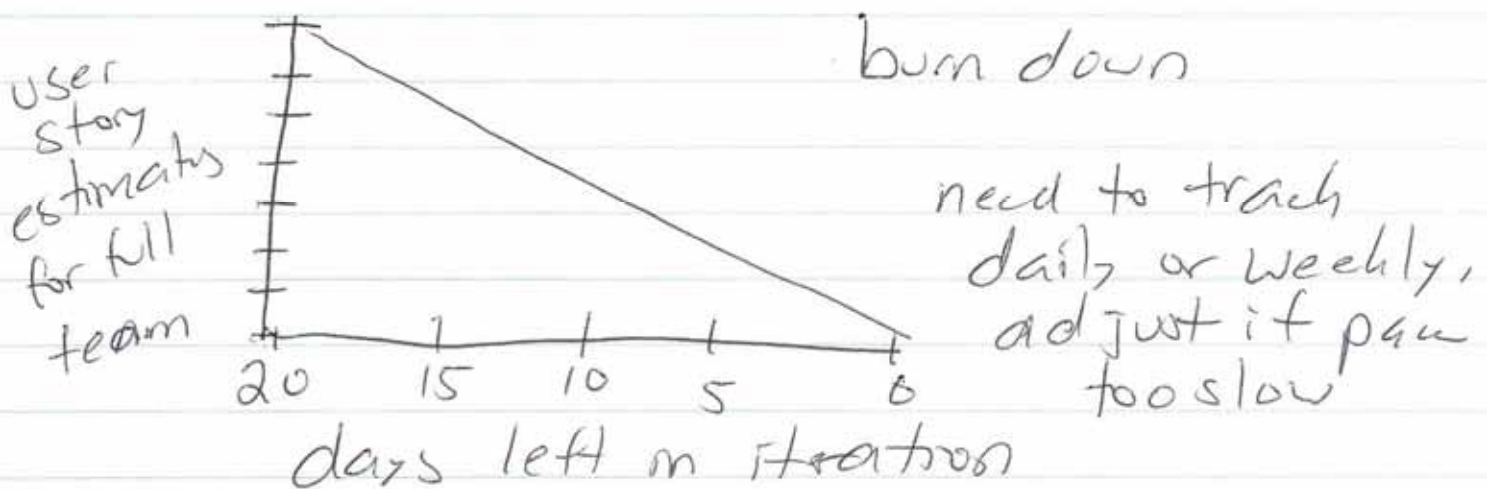recalibrate project velocity every iteration
  might be 0.6 or 0.8

Thursday 10/9/14

define iterations - 1 week or 1 month
3 iterations until 1st release -
90 day releases is common because
of quarterly financial planning

~~divide~~ split user stories among
iterations according to relative priority
& dependencies

user
story
estimates
for full
team

| | | | | | |
20  15  10  5  0

days left in iteration

burn down

need to track
daily or weekly,
adjust if pace
too slow

_____

now review - go through Head First books
          + Ron Patton chapters
student should bring specific
questions on Tuesday