Helping students and graduates ”see around corners”

Musings on what we can do to reduce errors in judgment

DON'T PANIC LABS

Don’t Panic Labs

- Small software development company (40 people)
- Specialize in building first versions or product reinventions
- 10-20 projects per year
- 100+ interns over 10+ years

DON'T PAN!C LABS.

Key observation on new graduates

- They are (by and large) well-educated on the body of knowledge
- They are almost all prone to significant errors in judgment

- Their internship experiences (and the value of them) varies widely

- Companies like ours cannot afford long “ramp up time” for engineers to be productive

DON'T PANIC LABS

Examples of errors in judgment

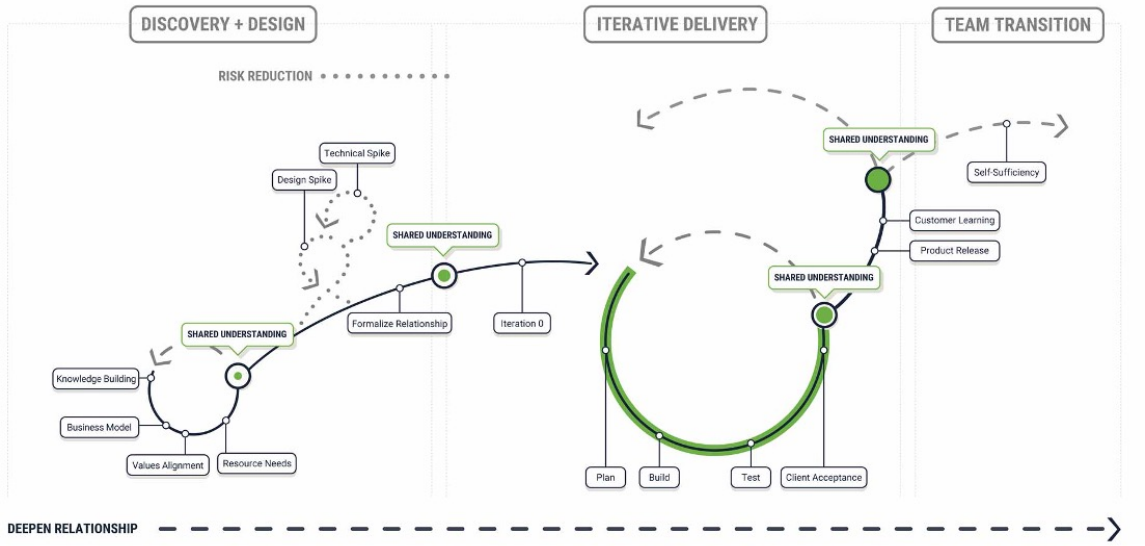
- Failure to…
  - recognize incomplete or ambiguous requirements
  - recognize implicit assumptions they are making
  - seek help or feedback from peers and leaders (impostor syndrome?)
- Bias towards…
  - leaping in vs planning and critical thought
  - focus on the “happy path”
  - testing to verify vs testing to break
  - “YOLO. What could go wrong?”

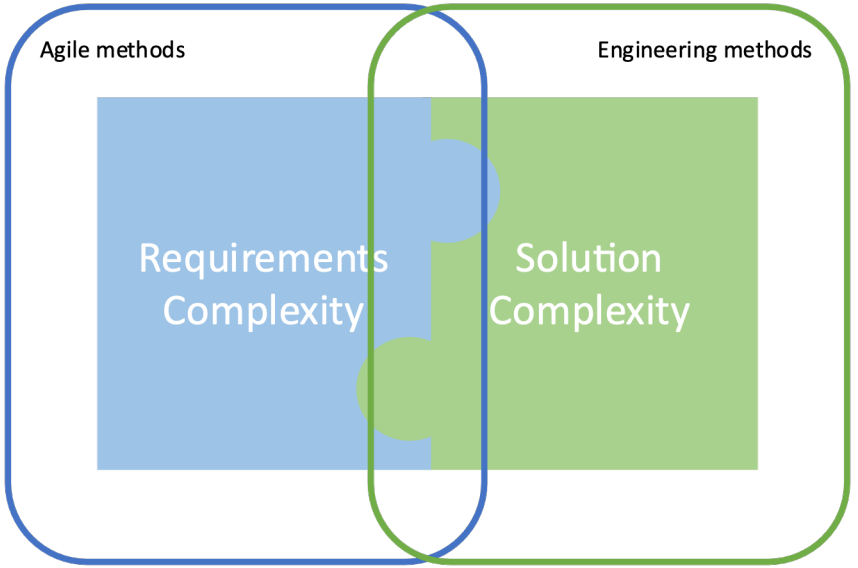
Experienced/valuable engineers…

- Develop stakeholder empathy
- See around corners
- Understand the importance of a shared understanding
- Understand need for modern tools, techniques, and processes
- Appreciate the complexity of building modern systems

We teach them to how the tools work but not how to use them to build a complex system

Key to our success… having a system that leaves little to chance





Key benefits

- Having the structure and discipline of a development system provides “guard rails” that allows the engineers to “fall into the pit of success”
  - Outcomes are more predictable
- The system creates the context for how and when to use their tools and knowledge and where to focus their continuing education and growth

- The alternative in software development is a “cowboy culture” where processes and practices vary widely

DON'T PANIC LABS

Some thoughts on what can help…

Increased educational focus on…

- … processes, tools, and methods for analyzing, designing, building, testing, and maintaining complex systems
  - Arm them with a system/approach to converge on a solution
  - Critical thinking tools/techniques that can help avoid errors in judgment
- … what they should expect to see when they enter industry
  - Roles, processes, etc
- … context for their course work (e.g., the value of engineering economics)

More (multiple) experiences with…

- … unconstrained/messy design problems
  - Working as a team
  - Non-deterministic outcomes
  - Both new systems and revisions to old systems
  - Required tradeoffs (i.e., put them in conflict)
  - Ambiguous/incomplete requirements
  - Iterative attempts in order to succeed
    - Make failure a key part of the learning
- … troubleshooting system problems
- … designing and executing tests that are intended to find problems as opposed to verify success

Thank you

"If builders built buildings the way programmers wrote programs,
then the first woodpecker that came along
would destroy civilization."

Gerald Weinberg

- ddurham@dontpaniclabs.com
- dougdurham.com
- @dnsdurham

DON'T PANIC LABS