

# E AGORA, JOSÉ?

ACH2006 – ENGENHARIA DE SISTEMAS DE INFORMAÇÃO

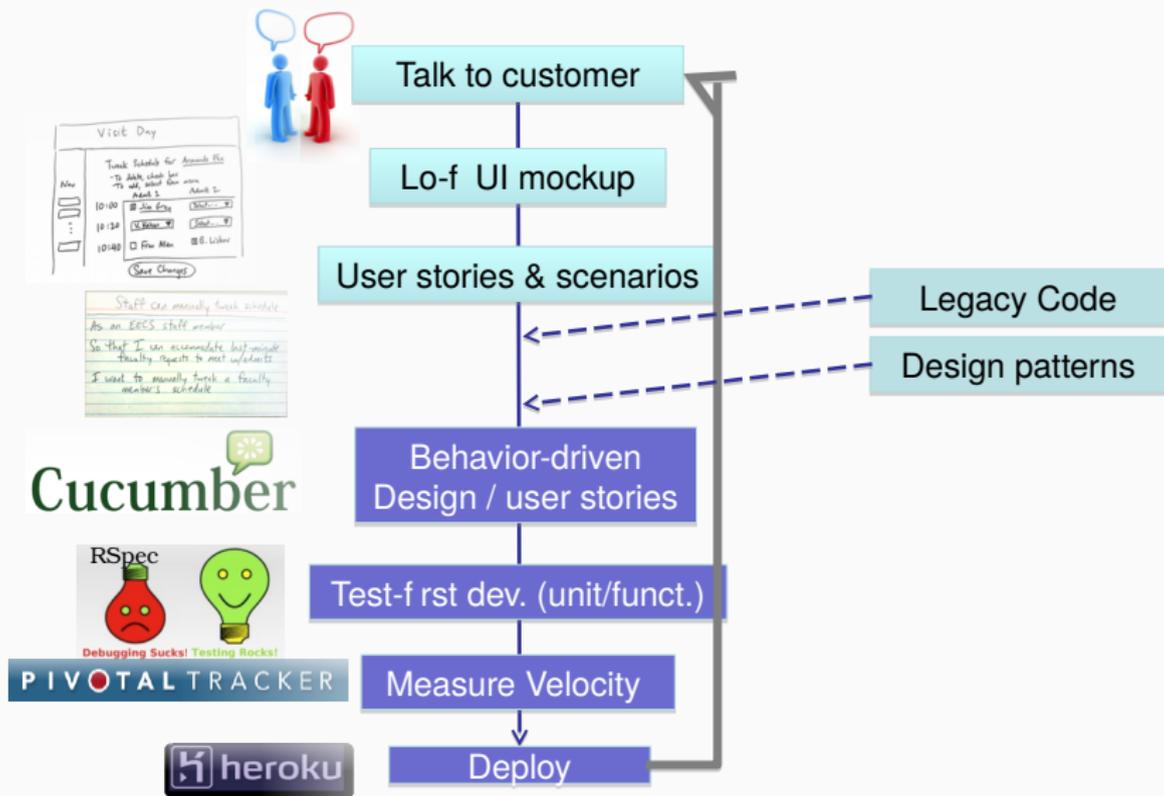
SIN5005 – TÓPICOS EM ENGENHARIA DE SOFTWARE

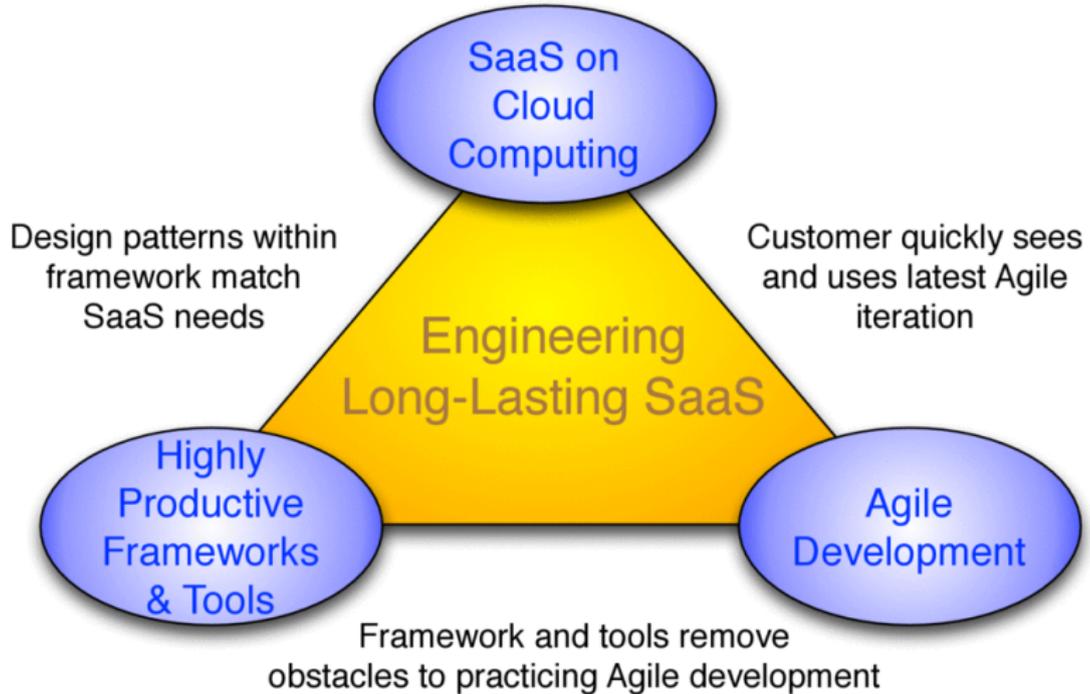
---

Daniel Cordeiro

Escola de Artes, Ciências e Humanidades | EACH | USP

# ENGENHARIA DE SOFTWARE EM 2 SLIDES





## SE VOCÊ FOR ESQUECER TUDO, LEMBRE-SE APENAS DISTO:

Ideia	Tópico da aula / leituras
Projeto conciso	Padrões de Projeto & Refatoração Aspectos (validações, filtros) Metaprogramação ( <code>attr_accessor</code> )
Testabilidade	Desenvolvimento guiado por testes Inserção “cirúrgica” de stubs e mocks SOFA & Testes
Desenvolvimento como um processo	Abrace as mudanças BDD usando cenários/iterações Ágil == sempre ter código funcionando

- Hacking ≠ desenvolver
  - há pouco espaço para desenvolvimento em áreas totalmente desconhecidas, exceto em algumas startups
  - hacker solitário ≠ herói do software
- Código funcionando ≠ código pronto
- Desenvoltura na fala, notas ≠ empregabilidade
  - seu código é o seu currículo
  - sua habilidade em pensar sobre um projeto por si só é sua munição para uma entrevista
  - aprenda uma nova linguagem de programação a cada 1–2 anos
  - Peter Norvig: “Teach Yourself Programming in Ten Years”

## POR QUE FALAR DESSES ASSUNTOS?

- 1/2 dúzia de companhias consultadas para saber o que elas queriam com EngSoft
    - Amazon Web Services, eBay, Facebook, GitHub, Google, Heroku, Microsoft, Pivotal Labs
1. Melhorar (o mal documentado) código legado
  2. Fazer de teste um cidadão de primeira classe
  3. Trabalhar com clientes não técnicos
  4. Fazer revisões de projeto
  5. Trabalhar em equipes

## POR QUE FALAR DESSES ASSUNTOS?

- Comitê de currículo de Engenharia de Software da ACM/IEEE: familiaridade com os processos Planeje-e-Documente<sup>1</sup>
  - processos de SW, gerenciamento de projetos de SW, ferramentas e ambientes, engenharia de requisitos, projeto de SW, construção de SW, validação e verificação de SW, evolução de SW, métodos formais, confiabilidade de SW
- 1/3 dos ex-alunos do curso de Berkley usam P-e-D no trabalho

---

<sup>1</sup>Fox, A., & Patterson, D. "Is the New Software Engineering Curriculum Agile?" IEEE Software, 30:5, Sept/Oct 2013.

## QUÃO POPULARES SÃO OS MÉTODOS ÁGEIS?

- Empresas produtoras de software que usam Ágil: Amazon, eBay, Facebook, Microsoft, Salesforce, ...
- Enquete sobre o estado de métodos ágeis em 2015 com 3880 pessoas:
  - 95% dos entrevistados trabalham em empresas que aplicam métodos ágeis
    - 43% em empresas onde métodos ágeis é a metodologia principal
    - 4% disseram que trabalham em empresas totalmente tradicionais/não-ágeis
    - apenas 1% dos 3880 disseram que a implantação de Ágil falhou
  - 82% dos entrevistados responderam que tinham pelo menos uma equipe distribuída aplicando Ágil (contra 35% três anos antes)
  - <https://versionone.com/pdf/VersionOne-10th-Annual-State-of-Agile-Report.pdf>

## COMO SE MANTER ATUALIZADO?

- Revistas Gerais da Área
  - IEEE Spectrum: <https://spectrum.ieee.org/>
  - Communications of the ACM: <https://cacm.acm.org/>
  - Computação Brasil: <https://www.sbc.org.br/publicacoes-2/298-computacao-brasil>
  - SBC Horizontes: <https://www.sbc.org.br/horizontes>
- Aprenda novas linguagens de programação e novos arcabouços
  - aprender novas estratégias de resolução de problemas vs. aprender novas ferramentas
- Livro *The Pragmatic Programmer: Your Journey To Mastery* de David Thomas e Andrew Hunt:  
<https://pragprog.com/titles/tpp20/>

- O software que vocês construíram é vosso!
- Mas se possível for, mantenha-os como código aberto :)
- Me avise se você quiser continuar com o projeto no Pivotal Tracker por mais um tempo
- Pretendo manter uma galeria com os projetos passados; se você não quiser que o seu integre a galeria, me avise!