I Already Know Agile, Thank You Very Much
Alan Atlas

Innovation Specialist, 18F
CTO, Scanbuy Inc.
30 years in high tech – HW Eng, SW Eng, Development Mgr/Dir/VP
Three industry-award-winning development projects (incl. Amazon S3)
Instrumental in scrum adoption at amazon.com
Agenda

Why Agile
Agile Manifesto and Principles
Iterative Development
Why did we need Agile?

- In the 80s, software development left much to be desired
  - slow and unpredictable
    - poor ability to deliver
  - bad quality
  - wrong features
  - bad UX
Why was it so bad?

- Waterfall methodology
- Immature tools and technology
- Lack of understanding of user-centered design
“Waterfall” Methodology
Waterfall Challenges

- Cost of change
- Cost of release
- Large decisions with least knowledge
- Predicting the future
- Illusion of progress
- Inability to respond to change
- Ambiguous accountability
- Poor quality
Agile

Design
Build
Test

Inspect
Potential Benefits of Agility

- Higher productivity
- Higher quality
- Increased flexibility and customer focus
- Reduced time to market
- Lower risk (both Market and Technical)
- More fun, more satisfying ==> better morale
We have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

(2001, Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas)
Agile Principles - Delivering

- Satisfy the customer through early and continuous delivery
- Deliver frequently, preferring a shorter timescale
- Working software is the primary measure of progress
Agile Principles - Working

- Technical excellence and good design enhances agility
- Simplicity, maximizing the amount of work not done
- Welcome changing requirements
- Sustainable pace
Agile Principles - Team

- Business & technical people work together daily
- Build teams with motivated individuals & trust them
- Best results emerge from self-organizing teams
- The team regularly reflects to become more effective
- Face-to-Face communication
So what is Agile, then?

- Agile is a culture, a way of thinking, and a way of creating things.
- Agile people try to do things in an agile way.
- Agile is an overall approach to delivering value and pleasing customers.
“Agile is something you are, not something you do.”

-Alan
Agile is not...

- a process
- a software development method(ology)
- a way to cut corners or build crap
- a way to produce low quality products quickly
- any old thing that you think is “a good thing to do”
- the Engineering Liberation Front
- a silver bullet that will fix everything that is wrong at your agency/organization
- always a good solution for your agency/organization
Agile ≠ agile!
Delivering Frequently
(Iterative Development)
Why deliver frequently?

- User feedback
  - Discover value (solutions)
  - Validate design
- Minimize Technical Risk
  - Avoid a large pile of unplanned, unknown work at the end by completing things in small bites
  - Make problems small and solve as you go
- Agility
  - Easy to quickly change direction
Agile development model

- Define
- Code
- Design
- Acceptance Tests
- Integration Tests
- Document
- Unit Tests
- Feature 1
- Feature 2
- Feature 3
- Feature 4
- Feature 5
- Feature 6

Time
Example:
Fed Ramp Dashboard
{{FEDRAMP Dashboard goes here}}
FedRAMP Dashboard

Total Authorized: 2
Leveraged ATOs: 0
Cost Savings: 0
FedRAMP Dashboard

Total Authorized: 0 (Sum of authorized CSPs)
Leveraged ATOs: 0 (Sum of leveraged ATOs)
Cost Savings: 0 (Total reuse * $250,000)
FedRAMP

Total Authorized: 0 (Sum of authorized CSPs)
Leveraged ATOs: {{homeController.leveragedAtos}} (Sum of leveraged ATOs)
Cost Savings: ${{homeController.totalCostSavings}} (Total reuse * $250,000)
Achieving a Vision

Plan-Driven

Desired

Actual

Start

Iterative

Desired

Actual

Start
Achieving a Vision, cont’d.

- Plan-Driven
- Iterative

Actual → Desired → Start

Actual → Desired → Start → Desired
What about things other than software development?

- Fit the agile model as closely as you can
- Deliver something usable frequently
- Build iteratively and do iterations
- Small, self-organized teams
- Continuous learning
The Family Tree

Lean
- Respect for People
- Continuous Learning

Agile
- Responding to Change
- Interactions Between People
- Working Software
- Customer Collaboration

eXtreme Programming (Coding)
Scrum (Proj. Mgmt.)
Kanban (Process Mgmt.)

Waterfall
Thanks!

Contact 18F@gsa.gov
Follow @18F
Work Together github.com/18F
Appendix
The Iterative Thought Process

- Focus on the underlying need
- Fulfill some of that need in the simplest way possible
- Produce a Usable Product ASAP
- LEARN
- Lather, rinse, repeat
What is an Agile organization?

- An organization that nurtures and supports Agile delivery teams
- An organization that incorporates Agile culture and thinking in its daily work
Not like this....

Like this!