Intro to Agile: DevPoint Labs

Hala Saleh

www.27sprints.com | @HalaSaleh1
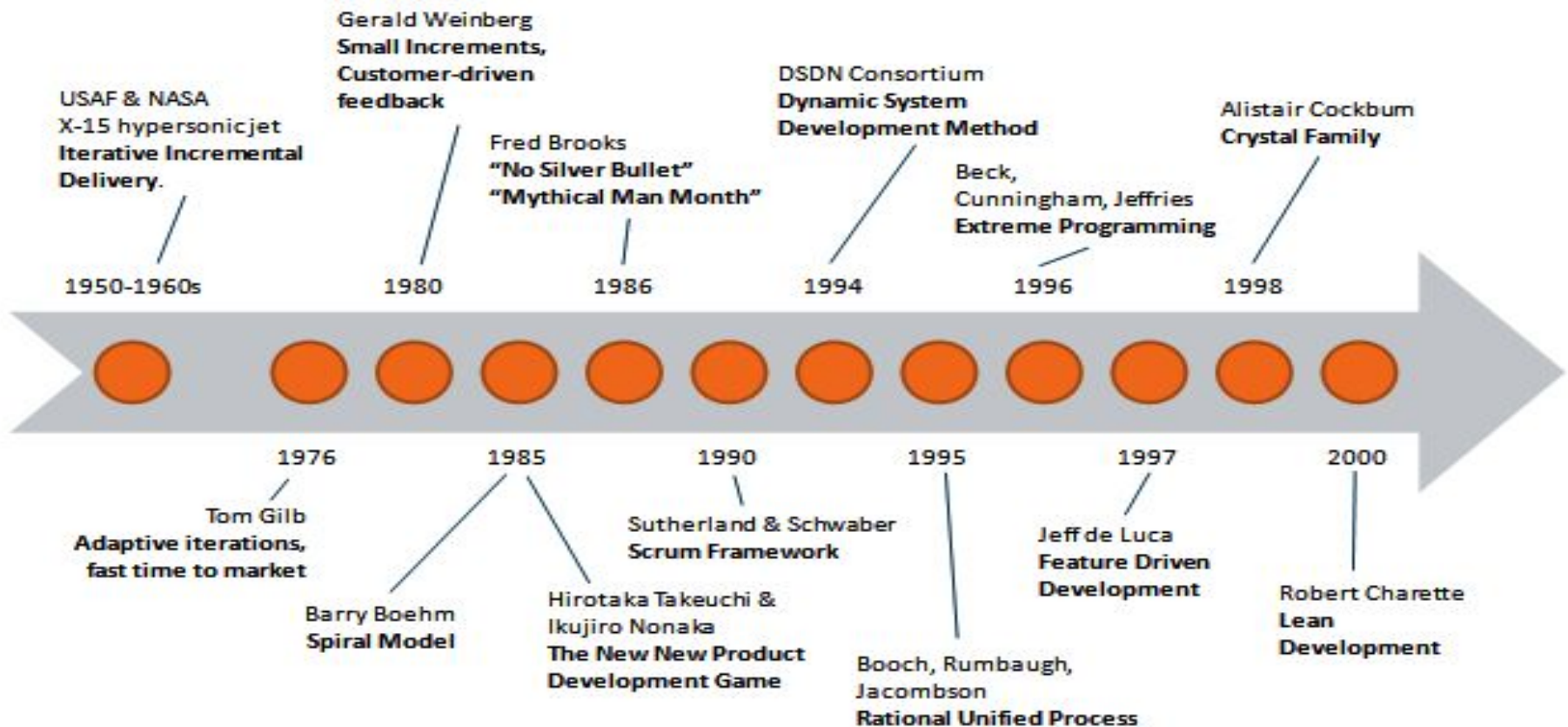
27Sprints

Objectives +
The Parking Lot

Image source: http://www.decoist.com/2013-03-12/stunning-parking-garage-designs/

AGILE: HISTORY & BACKGROUND

Gerald Weinberg
Small Increments,
Customer-driven
feedback

USAF & NASA
X-15 hypersonic jet
Iterative Incremental
Delivery.

DSDN Consortium
Dynamic System
Development Method

Alistair Cockburn
Crystal Family

Fred Brooks
"No Silver Bullet"
"Mythical Man Month"

Beck,
Cunningham, Jeffries
Extreme Programming

1950-1960s   1980   1986   1994   1996   1998

1976   1985   1990   1995   1997   2000

Tom Gilb
Adaptive iterations,
fast time to market

Sutherland & Schwaber
Scrum Framework

Jeff de Luca
Feature Driven
Development

Barry Boehm
Spiral Model

Hirotaka Takeuchi &
Ikujiro Nonaka
The New New Product
Development Game

Robert Charette
Lean
Development

Booch, Rumbaugh,
Jacombson
Rational Unified Process

Process Control Models

Linear/ Defined

Inspect   Adapt
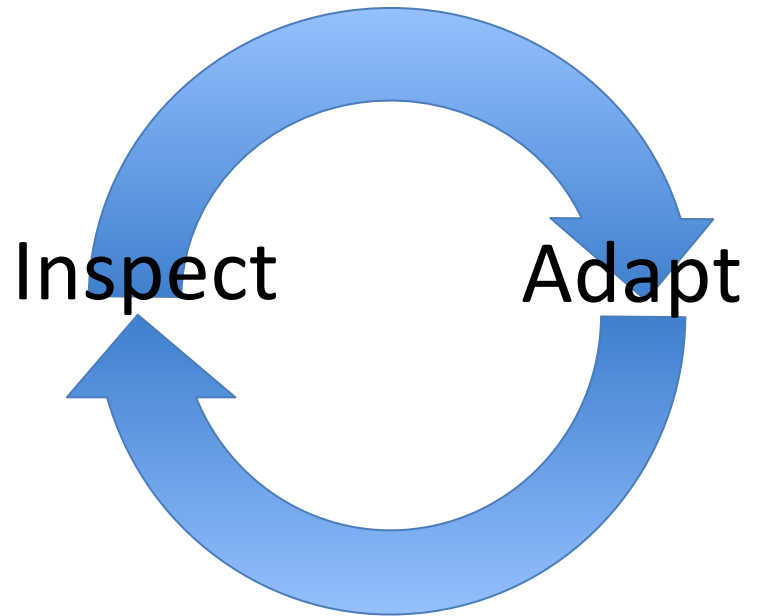
Empirical

Waterfall Product Development

Reqs

Design

Develop

Test

Deploy

Maintain
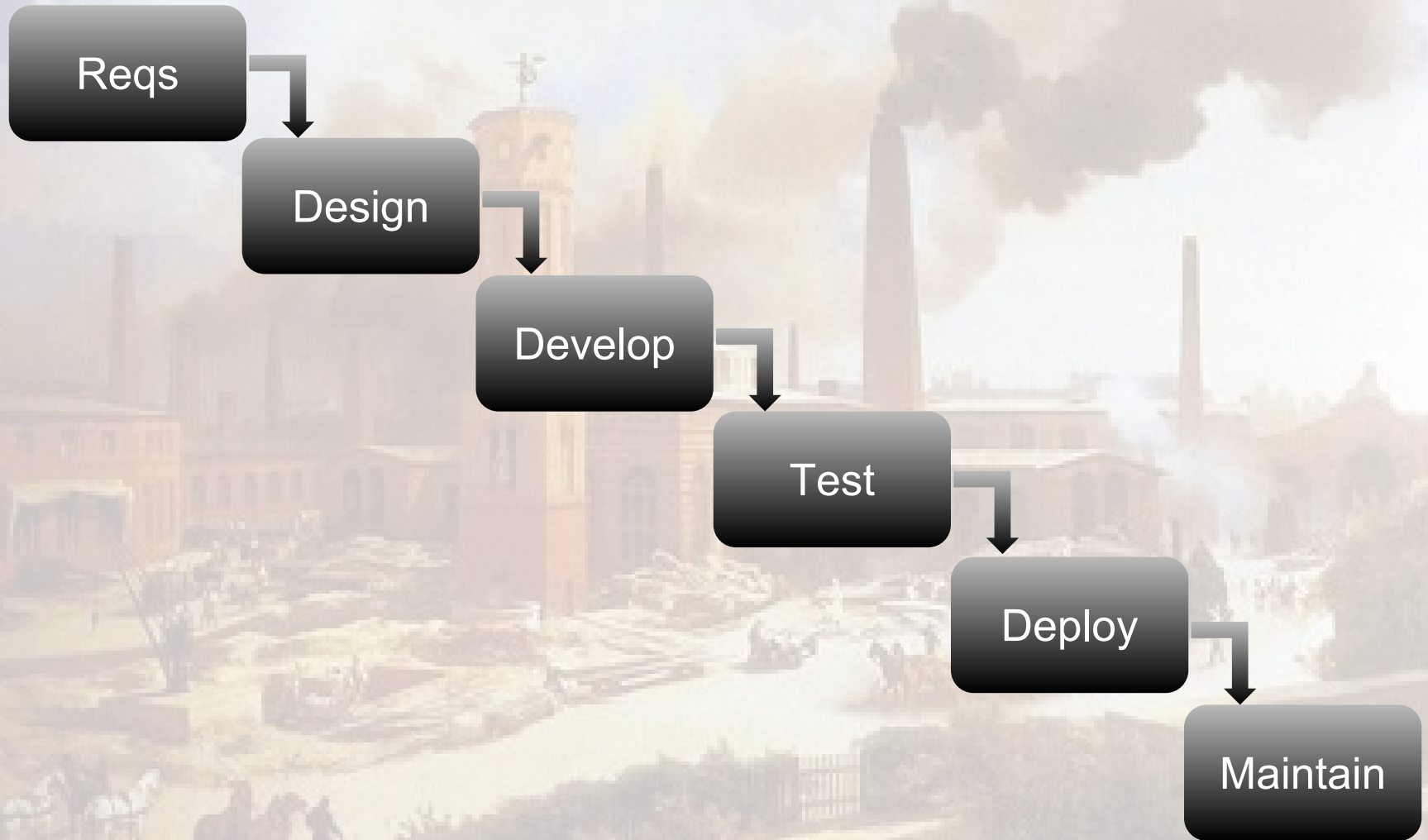
Waterfall For Software

In the 1960 and 70s, the software development industry inherited the manufacturing process model

Today: A New Environment

- One constant: Requirements, Technologies, & Customer Preferences now change faster than ever.

- Businesses are in a race to be first to market.

- Teams need to be able to **adapt** and **respond to change** to be competitive and fast to market.

- Organizations need to be able to reduce waste to meet their goals.

Agile & Lean:
Software & Beyond

- Organizations started focusing on reducing waste. (Toyota, MIT, Lean thought leaders)

- Software experts and business thought-leaders started exploring frameworks that were more lightweight and responsive to change.

- **The result:** Agile, Lean, & Value-Driven Approaches

Agile: The Values (Software)

- **Individuals and interactions** over processes and tools

- **Working software** over comprehensive documentation

- **Customer collaboration** over contract negotiation

- **Responding to change** over following a plan

While we value the items on the right, we value the items on the left more

*Source: agilemanifesto.org

Agile: The Principles

The Agile Manifesto is based on 12 Principles*

Collaboration

Timeboxing

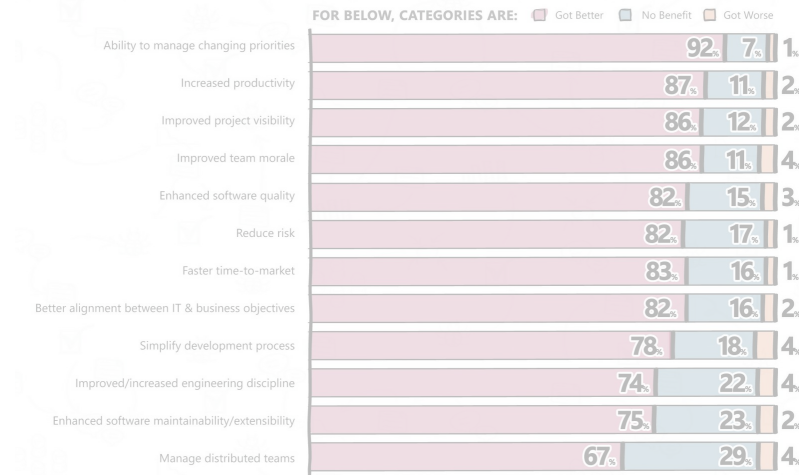**Continuous Feedback & Improvement**

SELF-ORGANIZING TEAMS

Retrospectives

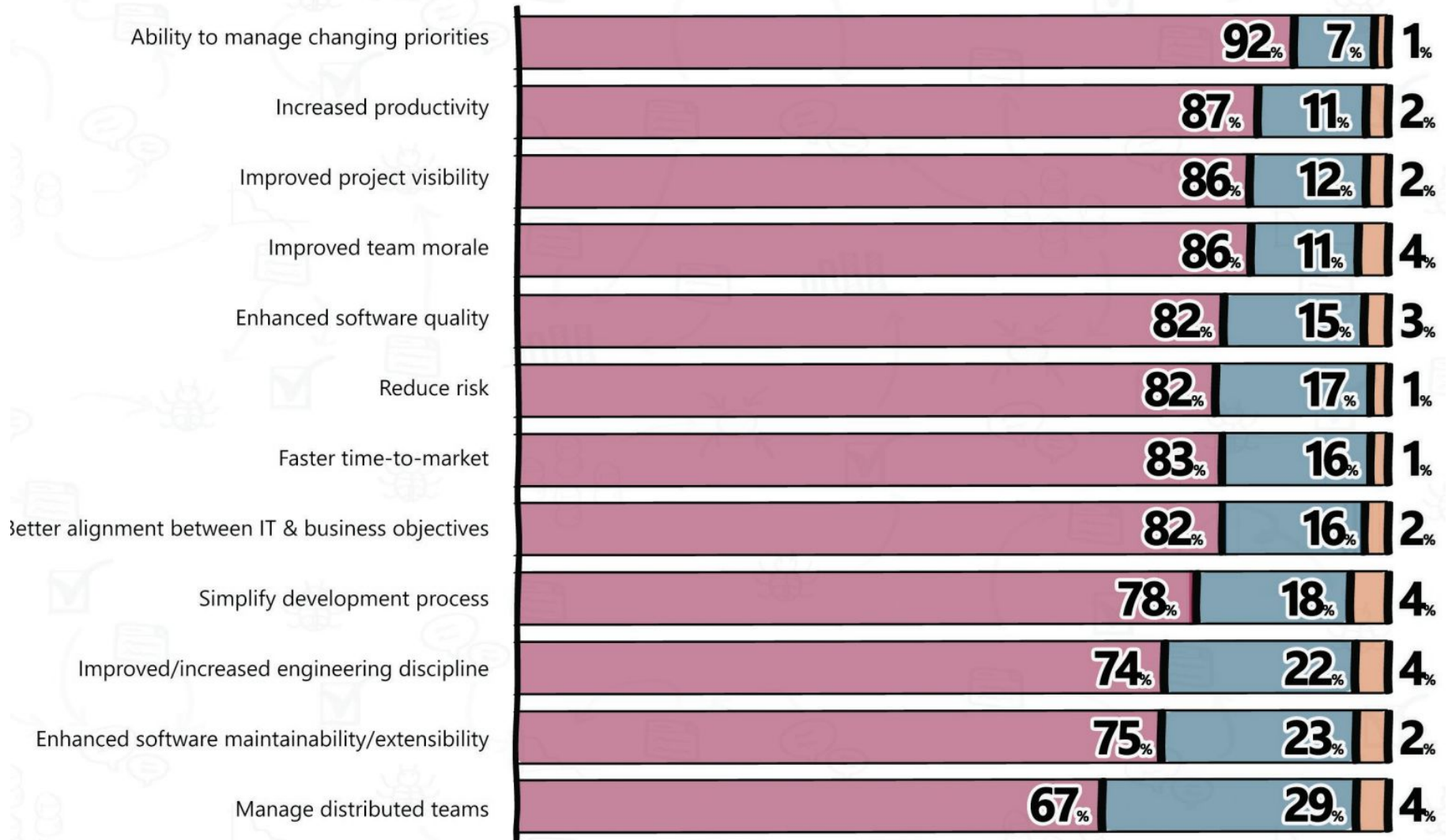**Frequent & Continuous Delivery**

Inspect-Adapt Cycle
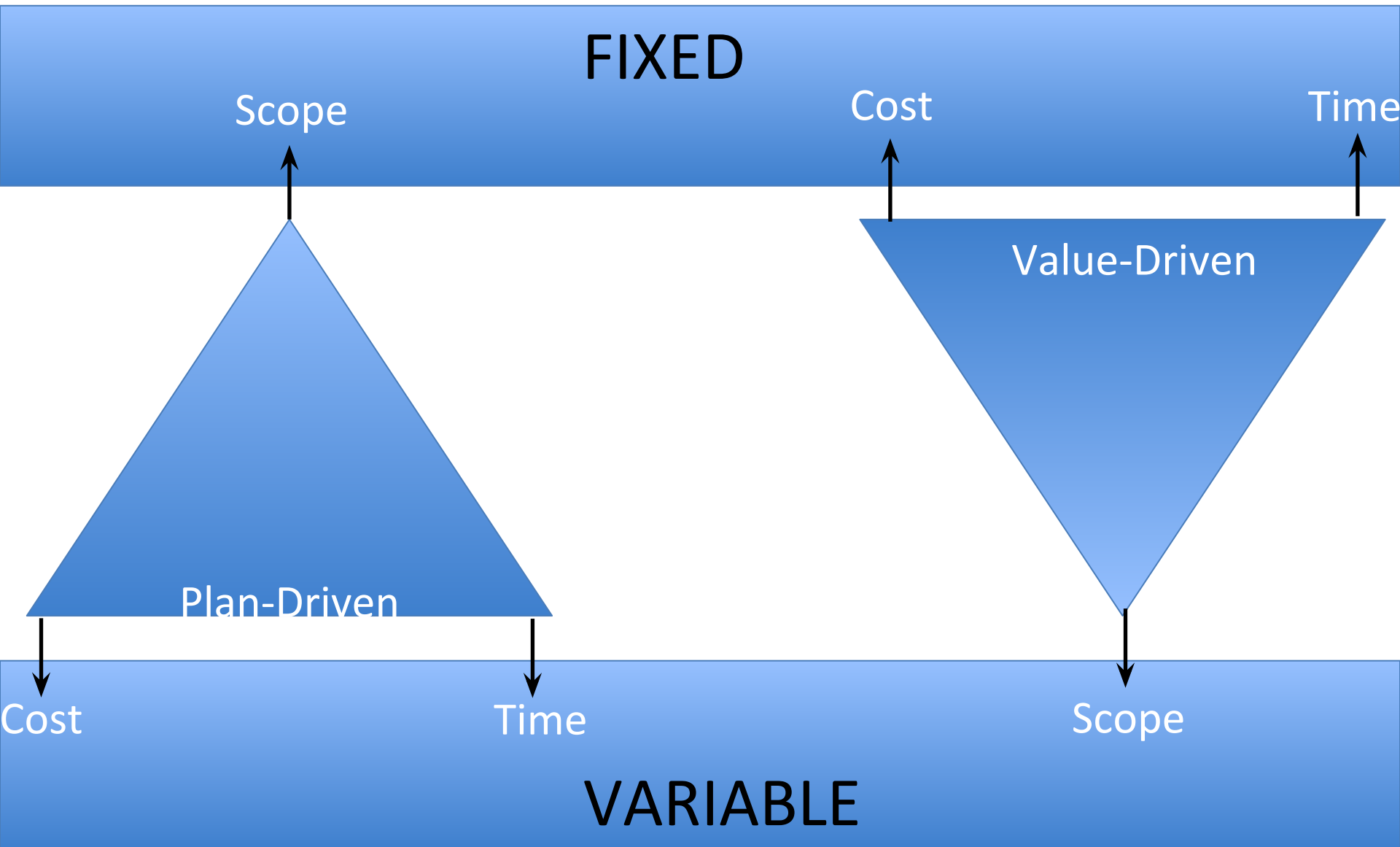
SUSTAINABLE PACE

* Source: agilemanifesto.org/principles.html

	Got Better	No Benefit	Got Worse
Ability to manage changing priorities	92%	7%	1%
Increased productivity	87%	11%	2%
Improved project visibility	86%	12%	2%
Improved team morale	86%	11%	4%
Enhanced software quality	82%	15%	3%
Reduce risk	82%	17%	1%
Faster time-to-market	83%	16%	1%
Better alignment between IT & business objectives	82%	16%	2%
Simplify development process	78%	18%	4%
Improved/increased engineering discipline	74%	22%	4%
Enhanced software maintainability/extensibility	75%	23%	2%
Manage distributed teams	67%	29%	4%

AGILE:
BENEFITS & DELIVERING VALUE

© 2015 **27Sprints** @HalaSaleh1

Agile: What Are The Benefits?

FOR BELOW, CATEGORIES ARE: ☐ Got Better ☐ No Benefit ☐ Got Worse

Category	Got Better	No Benefit	Got Worse
Ability to manage changing priorities	92%	7%	1%
Increased productivity	87%	11%	2%
Improved project visibility	86%	12%	2%
Improved team morale	86%	11%	4%
Enhanced software quality	82%	15%	3%
Reduce risk	82%	17%	1%
Faster time-to-market	83%	16%	1%
Better alignment between IT & business objectives	82%	16%	2%
Simplify development process	78%	18%	4%
Improved/increased engineering discipline	74%	22%	4%
Enhanced software maintainability/extensibility	75%	23%	2%
Manage distributed teams	67%	29%	4%

© 2015 27Sprints @HalaSaleh1

Plan-Driven vs. Value-Driven

FIXED

Scope   Cost   Time

Value-Driven

Plan-Driven

Cost   Time   Scope

VARIABLE

Agile Value Triangle

FIXED

Quality

Value

Constraints

VARIABLE

Deadline

HIGHEST VALUE

Business Value Delivered

Time

© 2015 27Sprints @HalaSaleh1
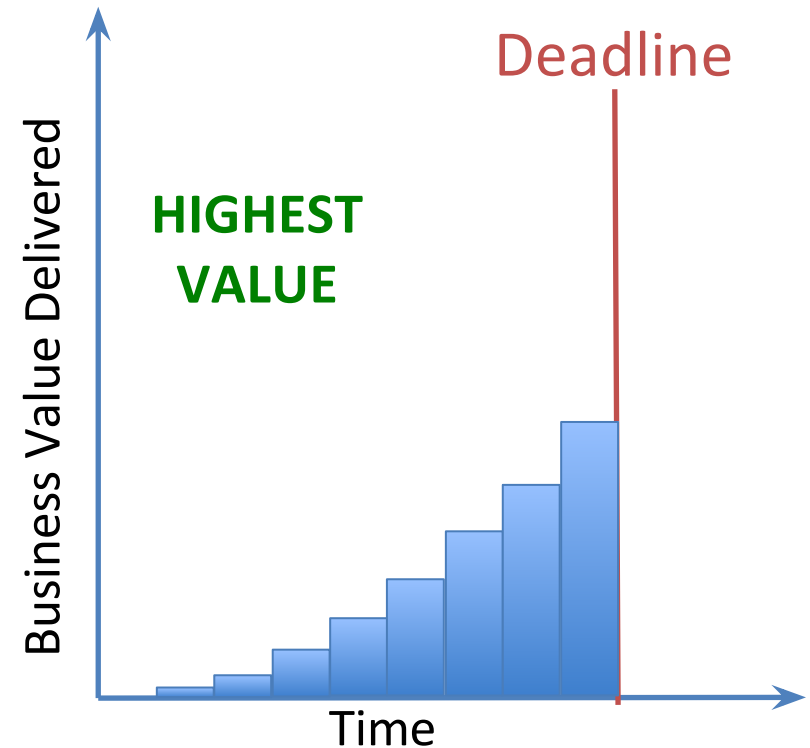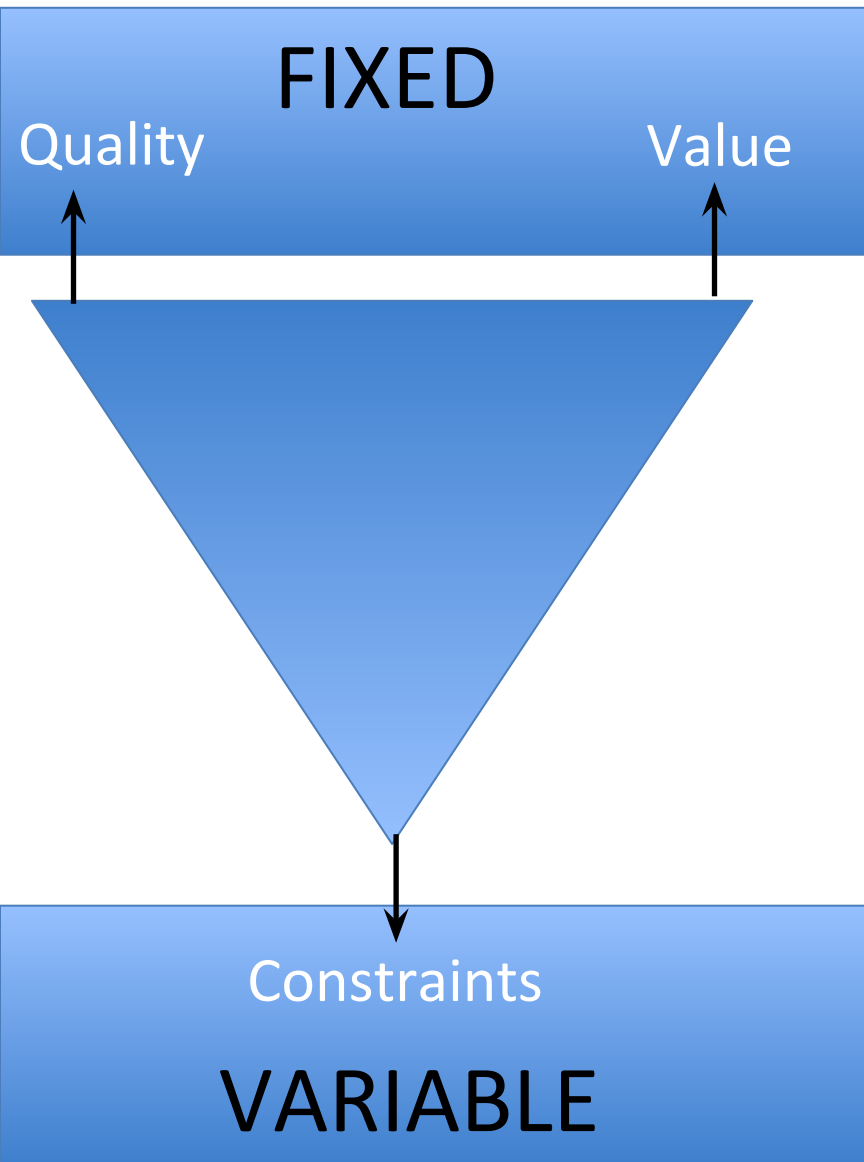
AGILE:
COMMON FRAMEWORKS

Common Agile Frameworks

Iterative & Incremental
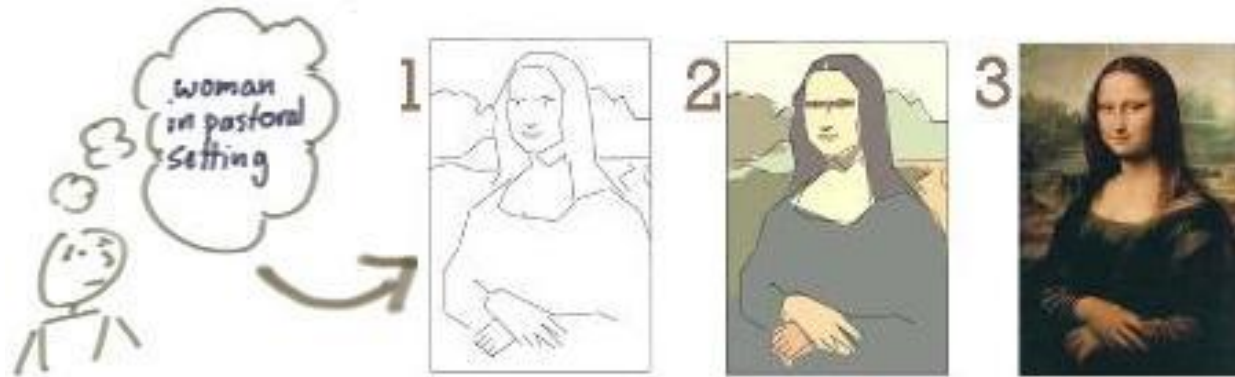
Image source: InfoQ: Jeff Patton Iterative & Incremental Mona Lisa

SCRUM:
AN INTRODUCTION

Scrum Framework
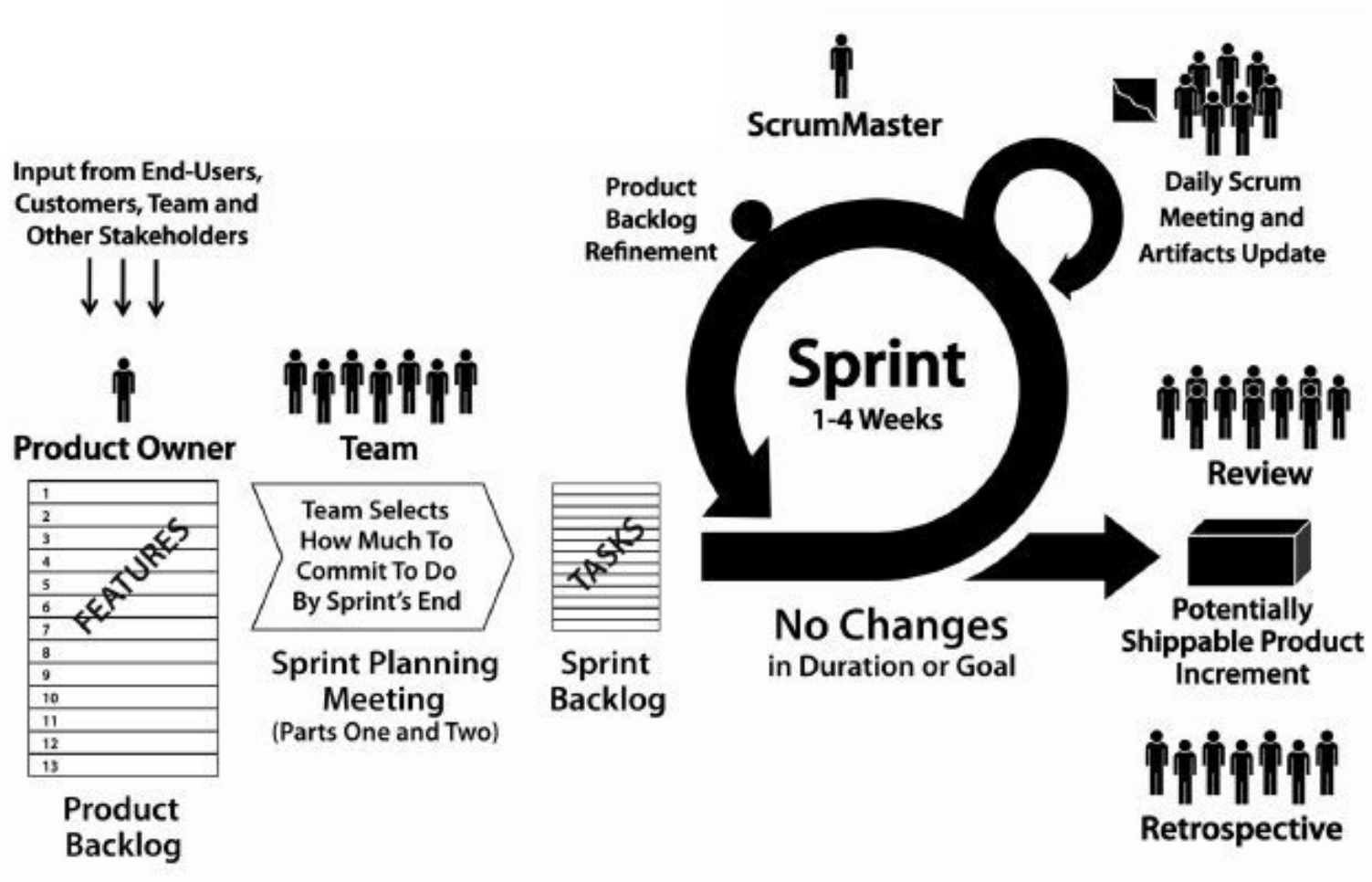
How Do I Incrementally + Iteratively Build Functionality?

Bare Necessity	what is essential functionality necessary to demonstrate the simplest use of the feature?
Capability, Flexibility	what would make this feature useful in more situations? What would allow the feature to be used in alternative ways?
Safety	what would make this feature safer for me to use for the user or other stakeholders? These are characteristics that help prevent errors or validate data to prevent bad information from entering the system.
Usability, Performance, Sex Appeal	what would make this feature more desirable to use? This includes adjustments to improve usability, visual and interaction design improvements to esthetic appeal, and improvements that help users do their work faster including architectural improvements or features such as short cuts or speed keys.

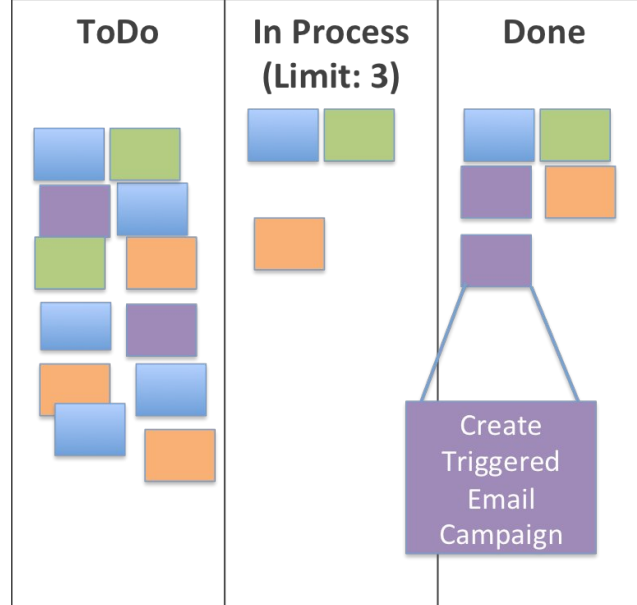# Scrum Framework:
An Overview

Roles

- Scrum Master
- Product Owner
- Team

Activities

- Release planning
- Sprint planning
- Daily scrum
- Sprint review
- Sprint retrospective
- Backlog refinement

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts
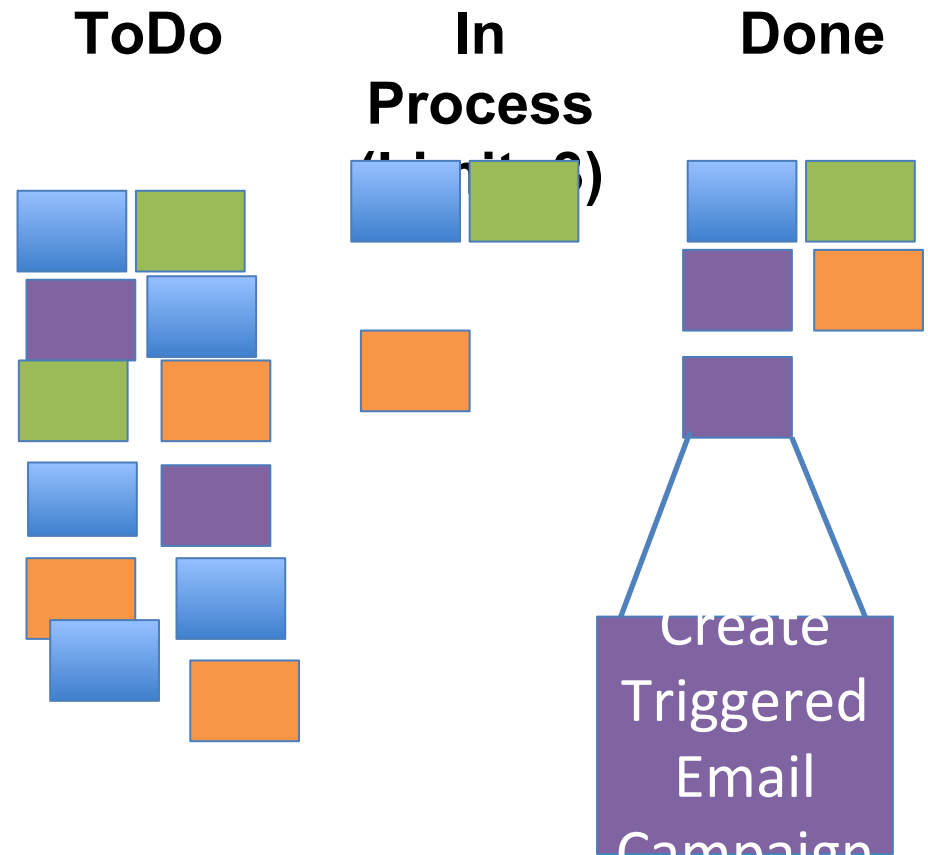- Product increments
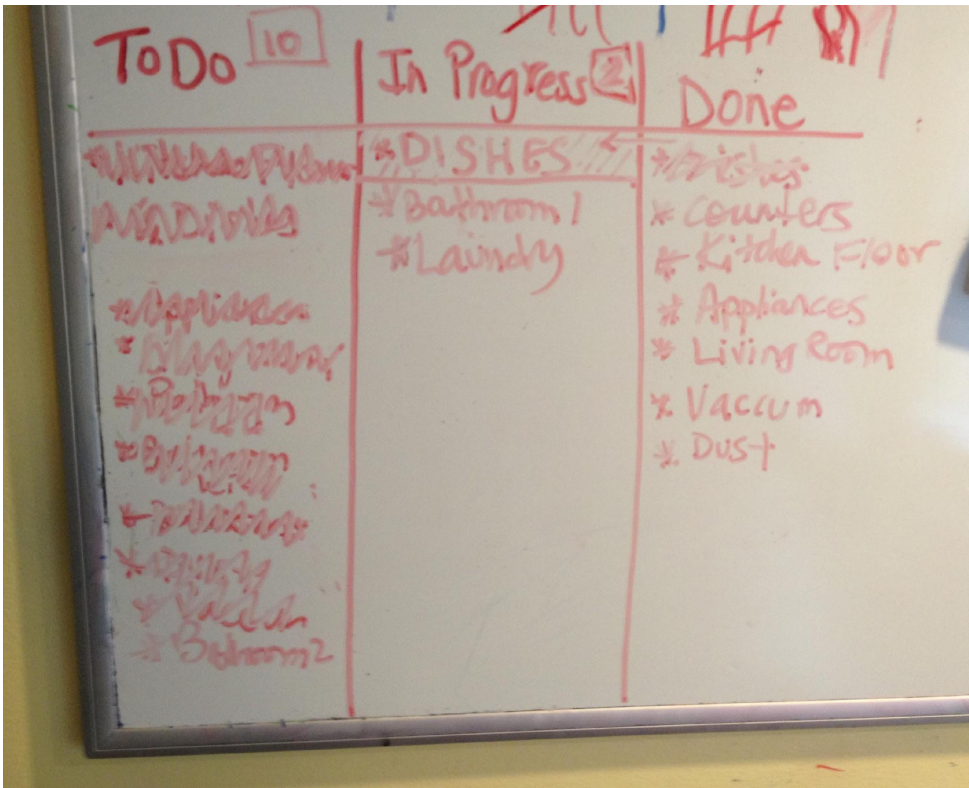- Team Task Board

KANBAN:
AN INTRODUCTION

Kanban: A Flow System

Kanban Principles:

1. Visualize Workflow

2. Limit Work in Process

3. Focus on Flow

4. Continuous Improvement

ToDo

In Process (Limit 3)

Done

Create Triggered Email Campaign

Visualize: Why?

1. Transparency
2. See bottlenecks: Wasteful processes, Procrastination
3. Focus: Focus on tasks vs. remembering
4. COLLABORATION!

Exercise: Sources of Waste

What are some sources of waste and ineffectiveness for businesses?

Limit WIP: Why?

1. Multitasking: It doesn't work.

2. Getting to "Done": Putting limits on "Doing" actually leads more stuff to get to "Done"

Keys For Successful Visual Tools

1. Simple: Visual tools should be simple with low overhead

2. Make key information visible to all: Easy to view the key information (avoid information-overload or not enough information)

3. Identify impediments to progress: Easy to see where there are issues (use color-coding, placement of post-its or other tools/ methods)

4. Keep team focused on outcome

5. Make a difference: If the tool doesn't increase effectiveness, efficiency, or quality (or other goals you have for using it), should be modified.

Is Agile For Me/ Us?

1. Delivery vs. Discovery

    a) Agile is about Predictable delivery when you're mostly confident you're building the right thing

    b) If your biggest risk is determining whether you're building the right thing, focus on fast learning cycles.

2. Focus on Agile Values, & Process Will Follow

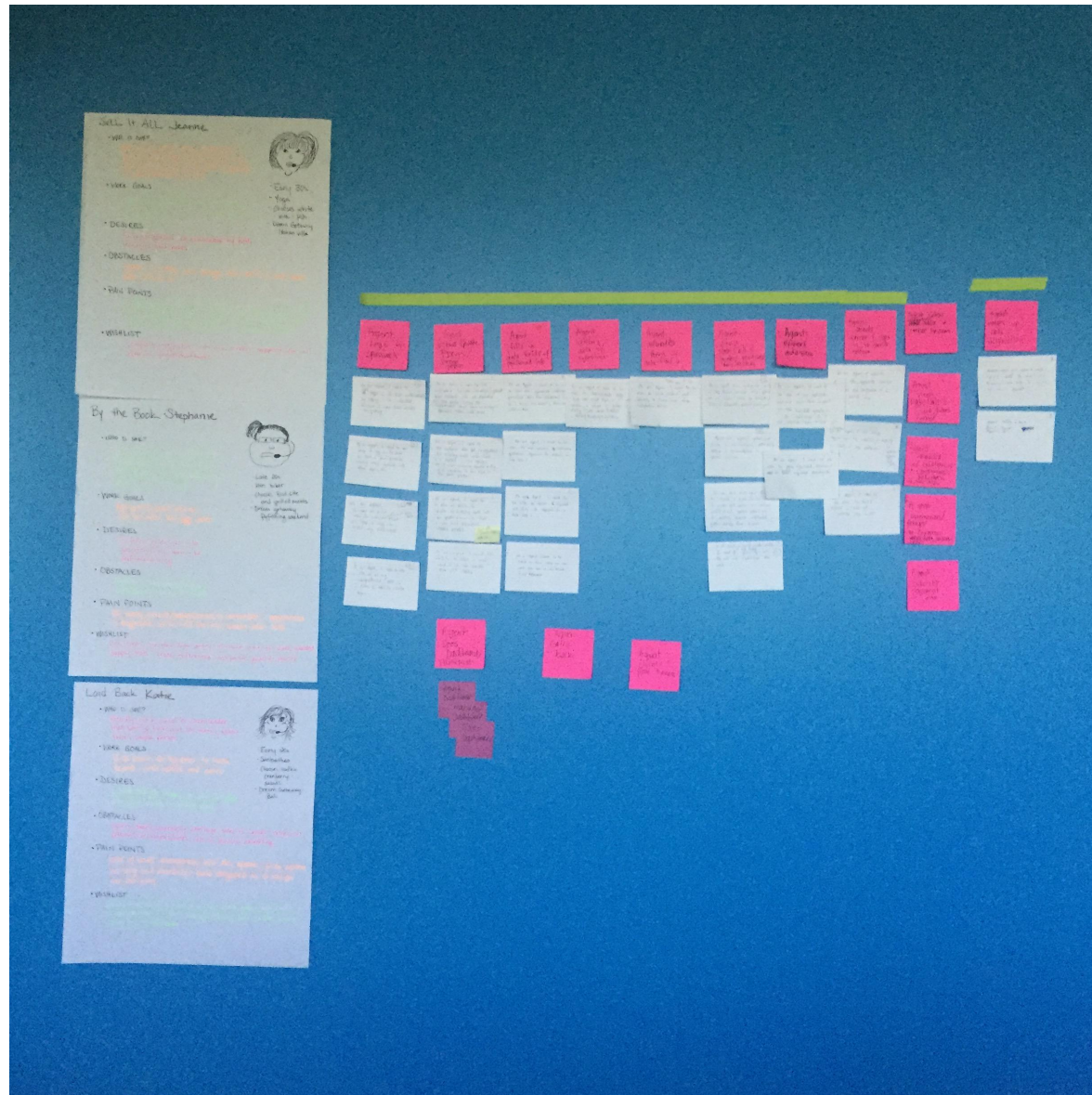Example: Nordstrom Innovation Lab

#1: There is NO Silver Bullet

#2: There is NO Monopoly on Good Ideas

#3: People != Resources

#4: Visualize #FTW

#5: Timeboxing solves EVERYTHING

© 2015 27Sprints @HalaSaleh1

#6: Multitasking is a Myth