

Kanban Training

DIEGO PACHECO

About me...



- Cat's Father
- Principal Software Architect
- Agile Coach
- SOA/Microservices Expert
- DevOps Practitioner
- Speaker
- Author

 diegopacheco

 @diego_pacheco

 <http://diego-pacheco.blogspot.com.br/>

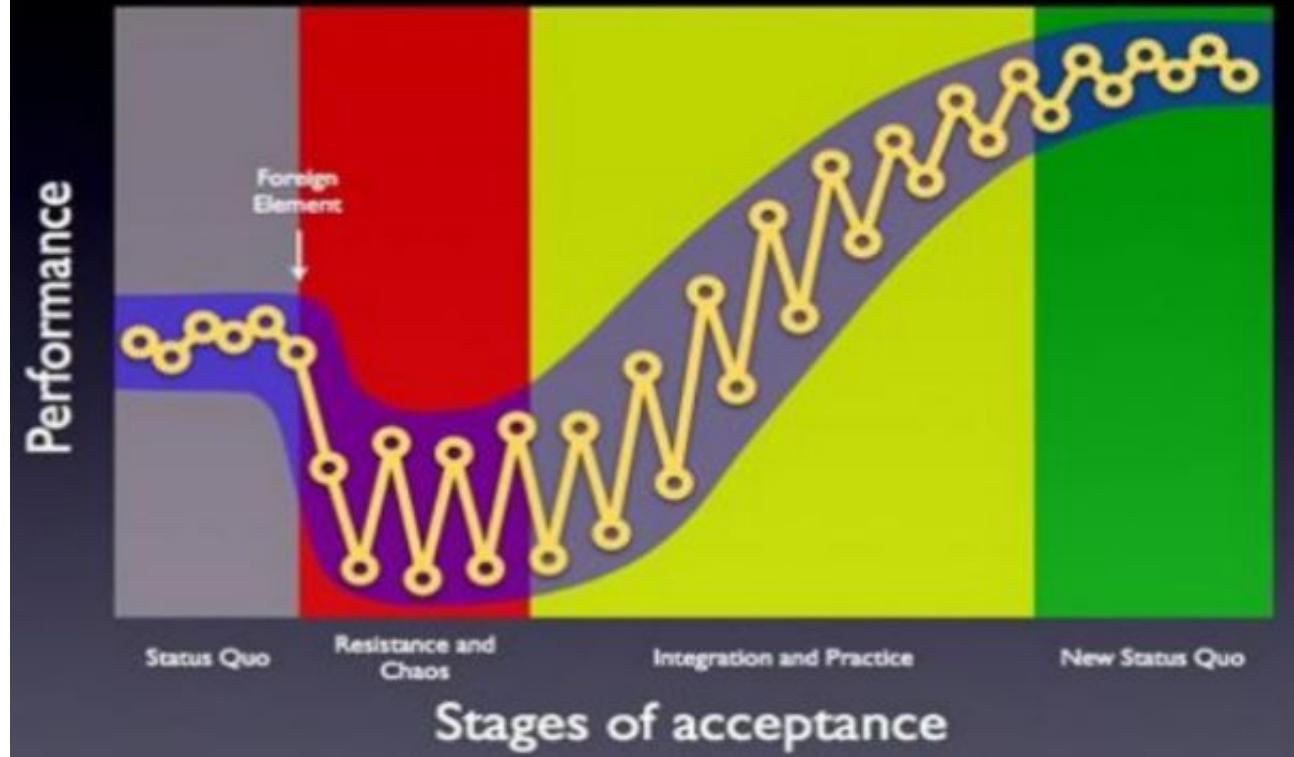


<https://diegopacheco.github.io/>



DISCLAIMER

Satir Change Model



**QUE O DESAPEGO
ESTEJA COM VOCÊ**



EEEETA MAINHA





Pastelaria

do
Beicola

Av. Borges de Medeiros | 3497 | Centro | Gramado | RS

ATENDIMENTO: Sábado e Domingo 11:30 às 02h
Segunda a Sexta 16h às 02h

TELE ENTREGA

54 3286 4290
9 9225 1826

ATENDIMENTO
terças à domingos 11h. à 0h.
segundas 16h. à 0h.

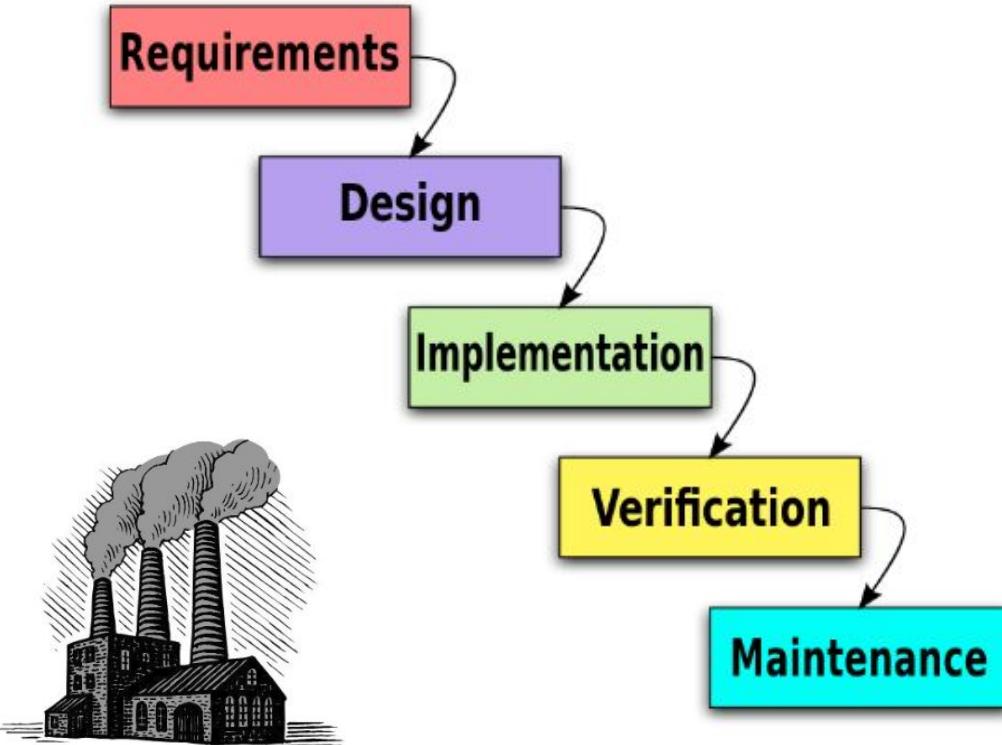
Leve a qualidade da Pastelaria do Beicola
para sua Casa e Evento...

A cartoon character of a woman with glasses and a yellow shirt, holding a piece of food, is positioned above the logo.

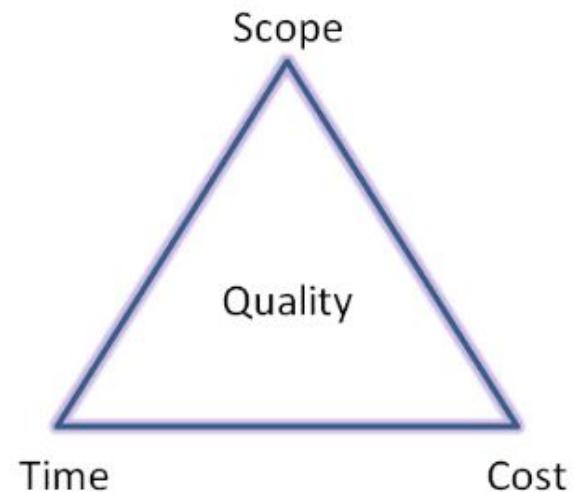
A basket of pastries is shown in the bottom right corner.



Waterfall



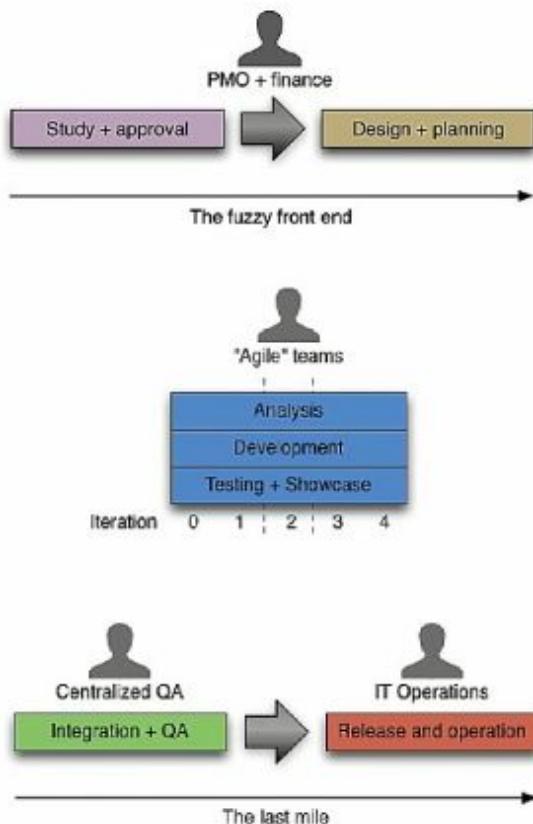
- What is software?
- Buy X Sell
- ALL FIXED
- Low Feedback
- BDUP
- Dev X Ops X ...



Agile

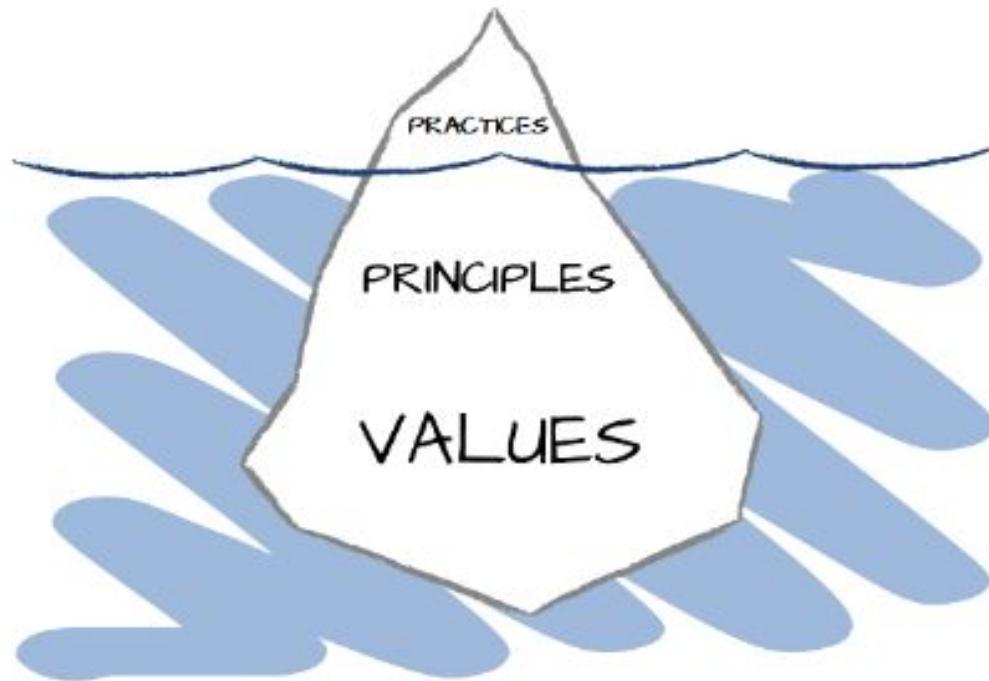


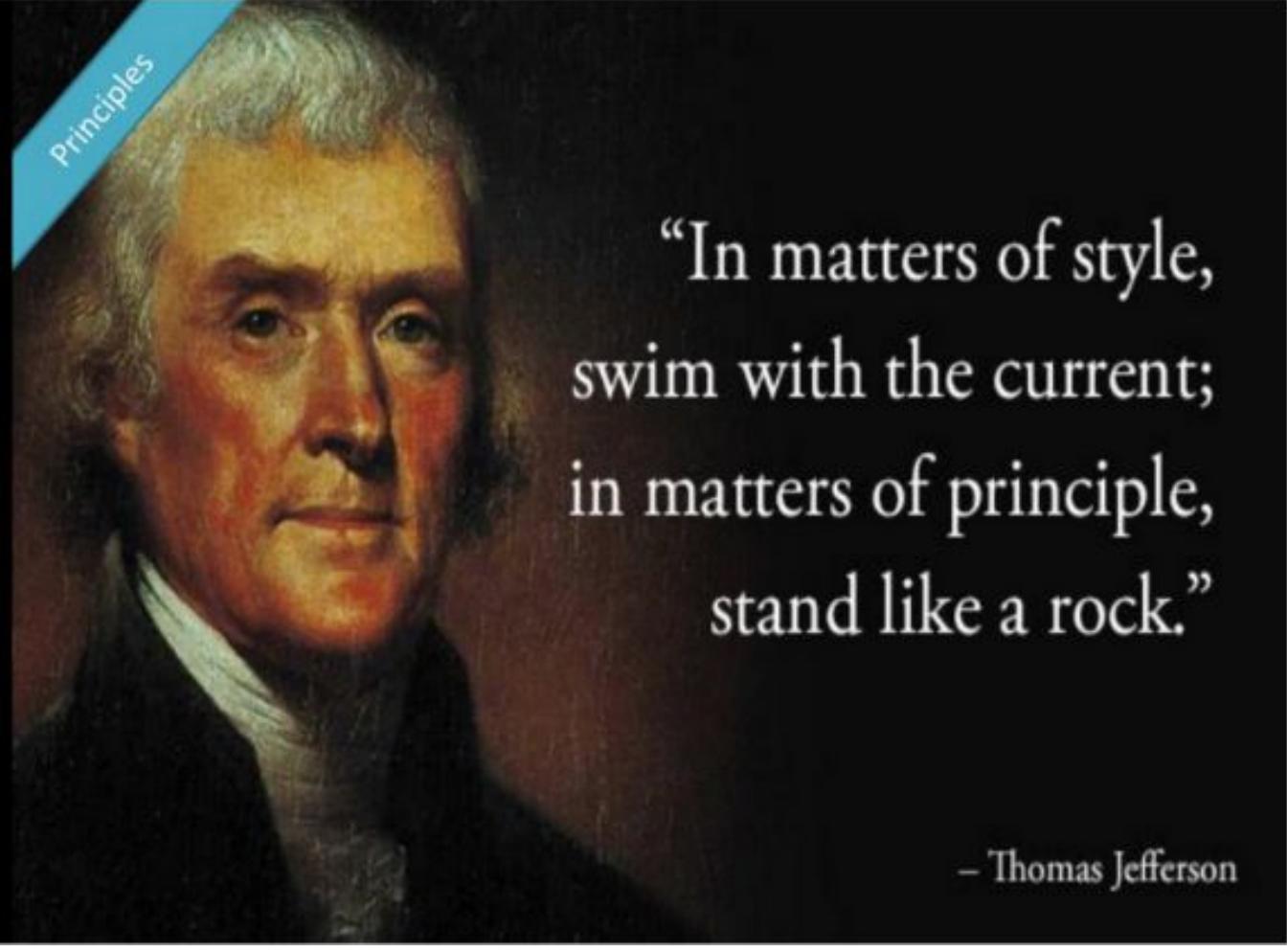
Water-Scrum-Fall



water-
scrum-
fall





A portrait painting of Thomas Jefferson, the third President of the United States. He is shown from the chest up, wearing a dark blue or black coat over a white cravat and a white waistcoat. His hair is powdered and grey. The background is dark and indistinct.

Principles

“In matters of style,
swim with the current;
in matters of principle,
stand like a rock.”

– Thomas Jefferson

A photograph of a diverse group of approximately ten people of various ages and ethnicities, all wearing headsets with microphones. They are gathered around a large, round conference table in what appears to be a modern office or meeting room. The lighting is soft and warm, creating a collaborative and focused atmosphere. The people are looking down at something on the table, possibly a device or a document, and seem to be engaged in a team discussion or a video conference.

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work 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.

The AGILE Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work 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.

Ken 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 Meller Ken Schwaber Jeff Sutherland Dave Thomas

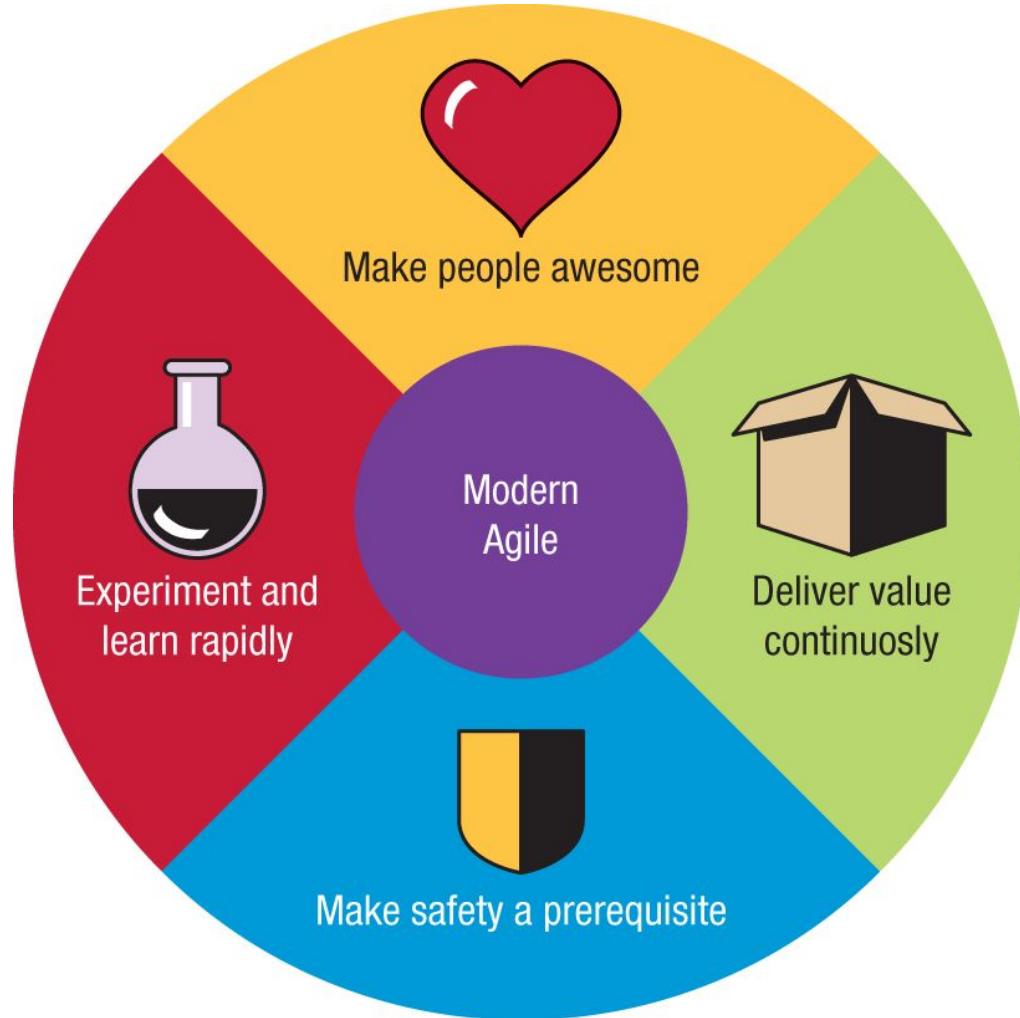


12 Principles of Agile Software

- 01** Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- 02** Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- 03** Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- 04** Business people and developers must work together daily throughout the project.
- 05** Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- 06** Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- 07** Working software is the primary measure of progress.
- 08** The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- 09** Continuous attention to technical excellence and good design enhances agility.
- 10** Simplicity—the art of maximizing the amount of work not done—is essential.
- 11** The best architectures, requirements, and designs emerge from self-organizing teams.
- 12** At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

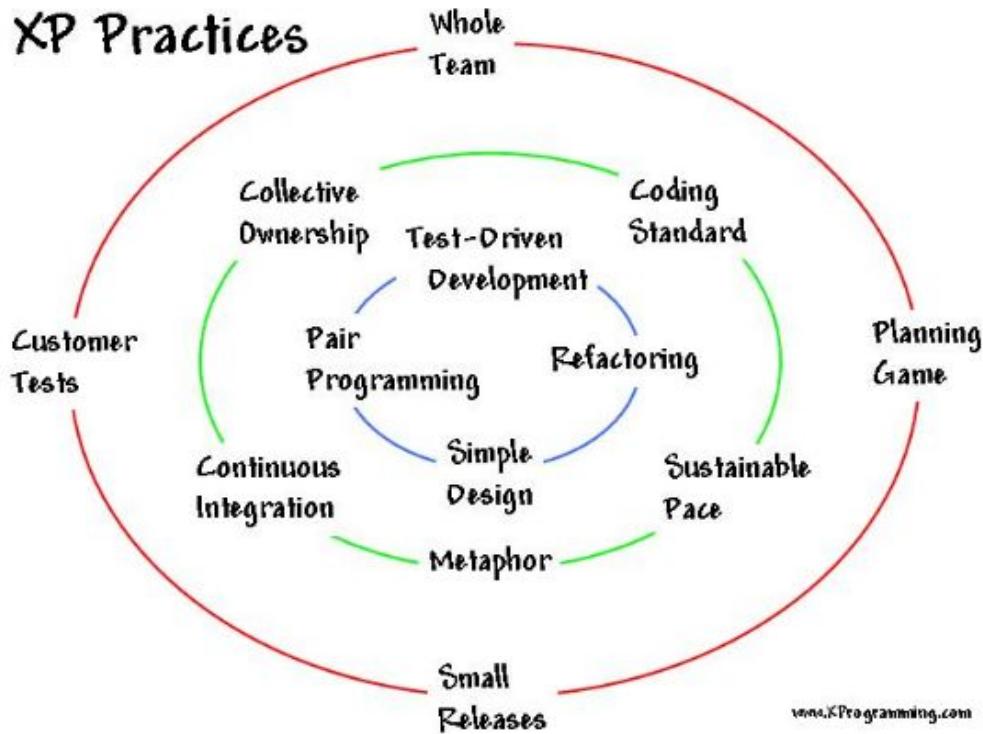
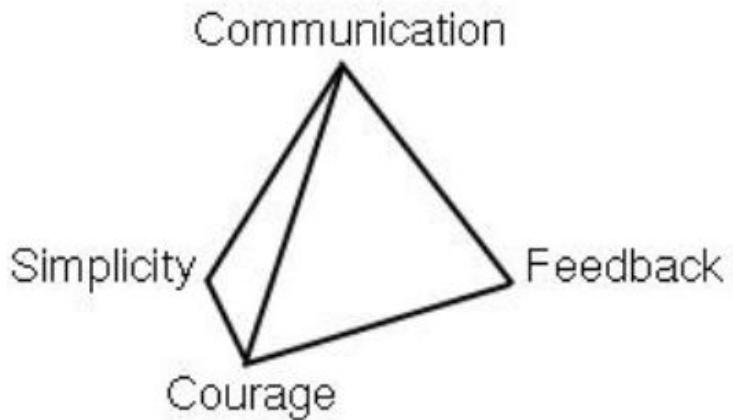
**“Discipline is doing
what you don’t want
to do when you
don’t want to do it.”**

Posted in the wrestling room at
Graham High School, St. Paris, OH

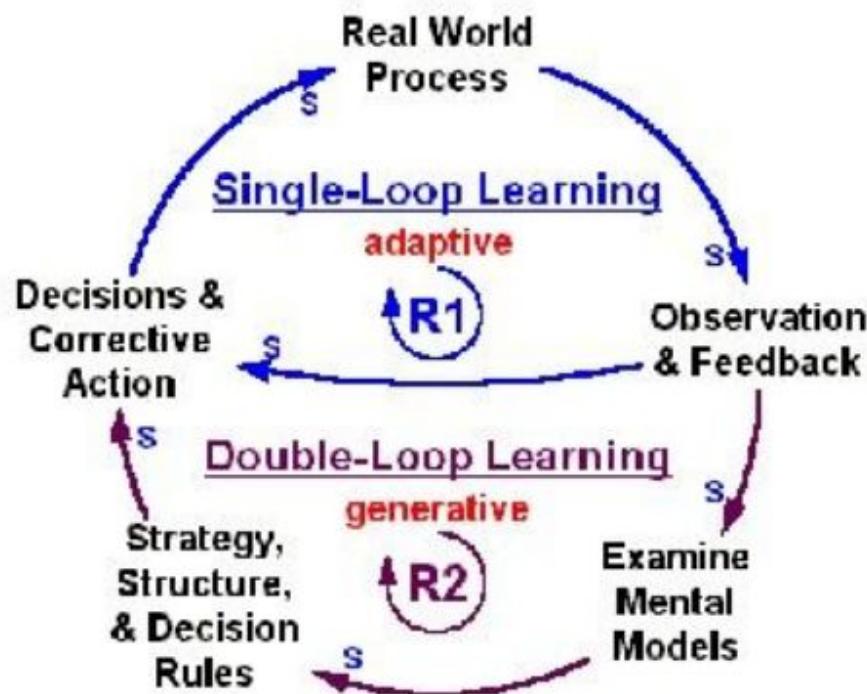
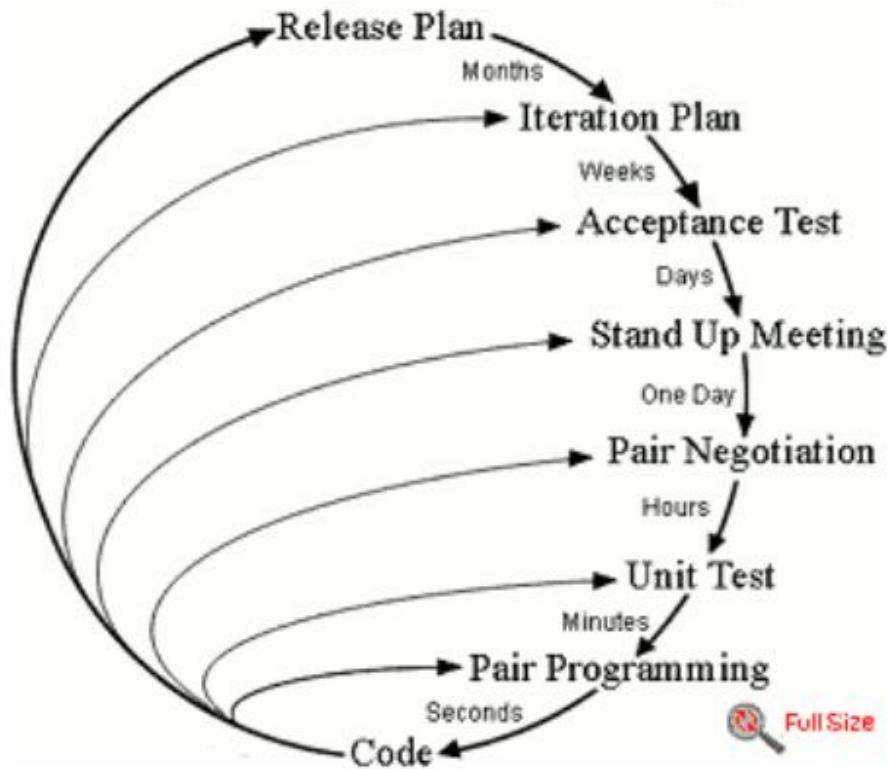


XP





Planning/Feedback Loops



XP Dimensional Quality

XP Evolutionary Approach



Chão Batido



Paralelepípido



Autoestrada



pair programming

driver

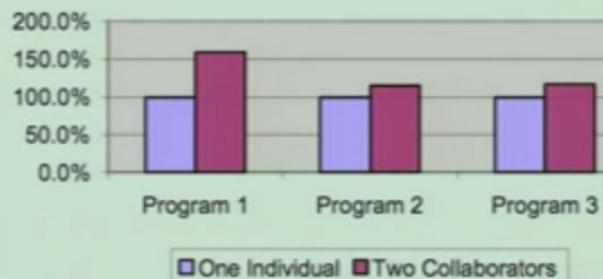
- typing
- micro-concerns
- syntax
- formatting
- line-by-line
- getting the test to pass

navigator

- thinking
- macro-concerns
- right class?
- refactor?
- algorithm?
- using design patterns

pair programming studies

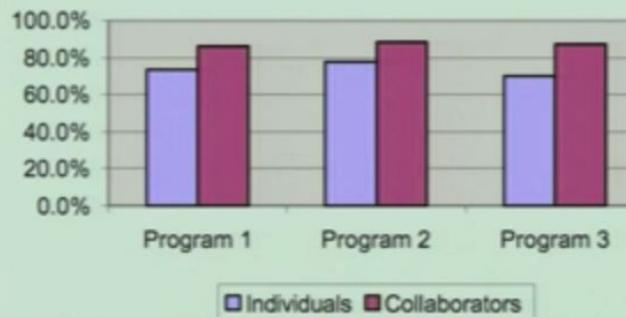
Relative Time: One Individual vs Two Collaborators



after adjusting, pairs produced code 15% more slowly than individuals...

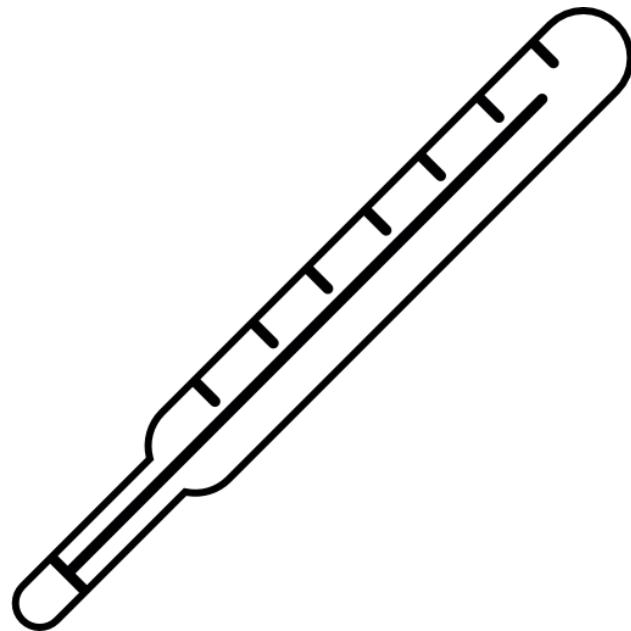
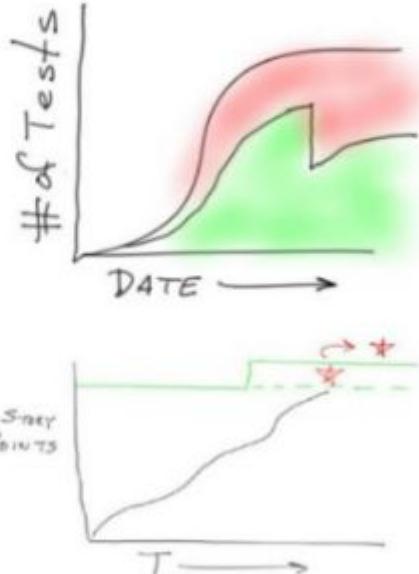
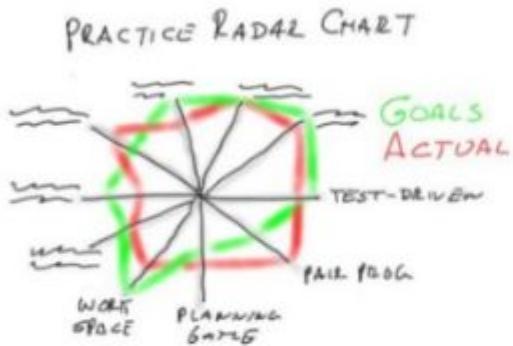
pair programming studies

Post Development Test Cases Passed



...with 15% fewer defects

Big Visible Charts



As who I want
what so that
why



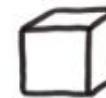
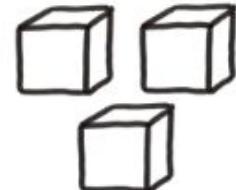
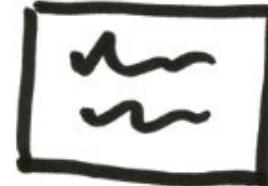
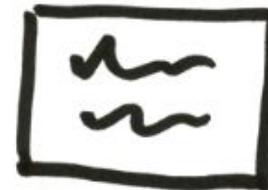
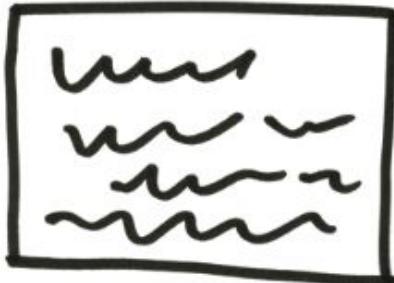
As who I want
what so that
why



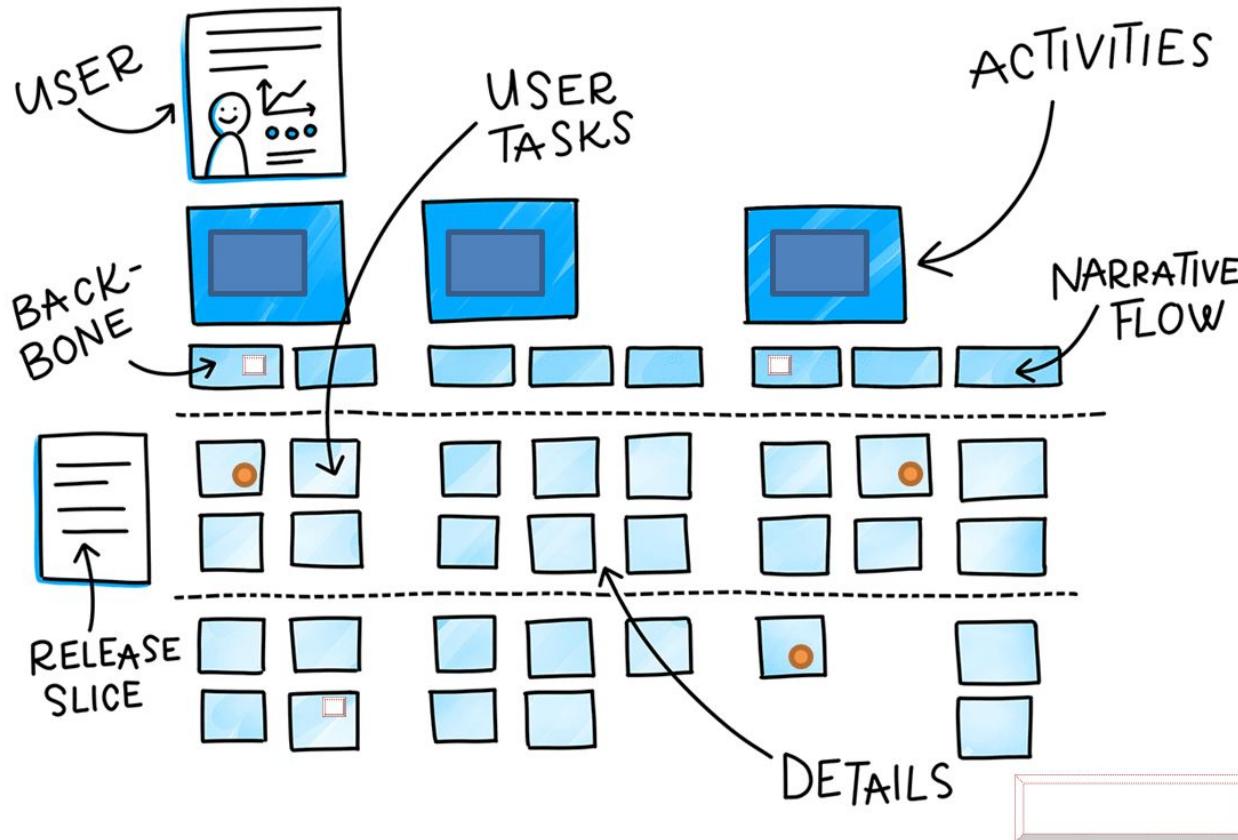


themes epics user stories features

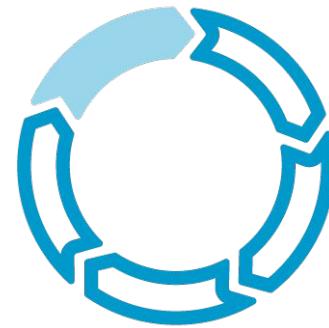
order
processing



USER STORY MAPPING

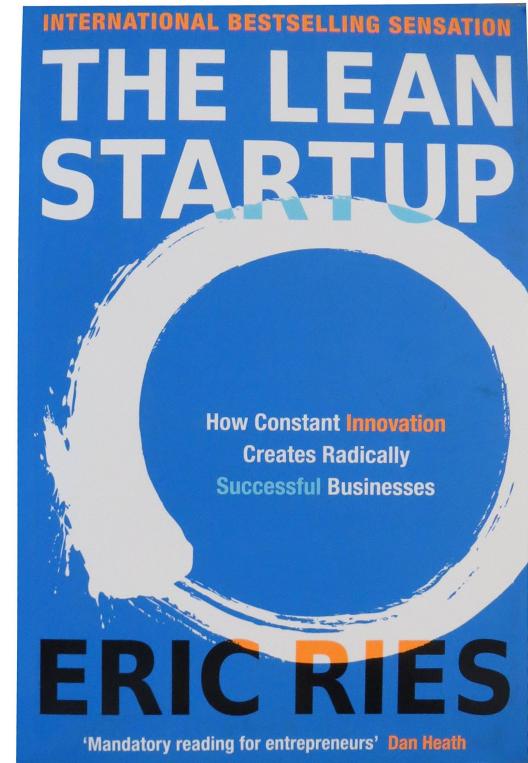
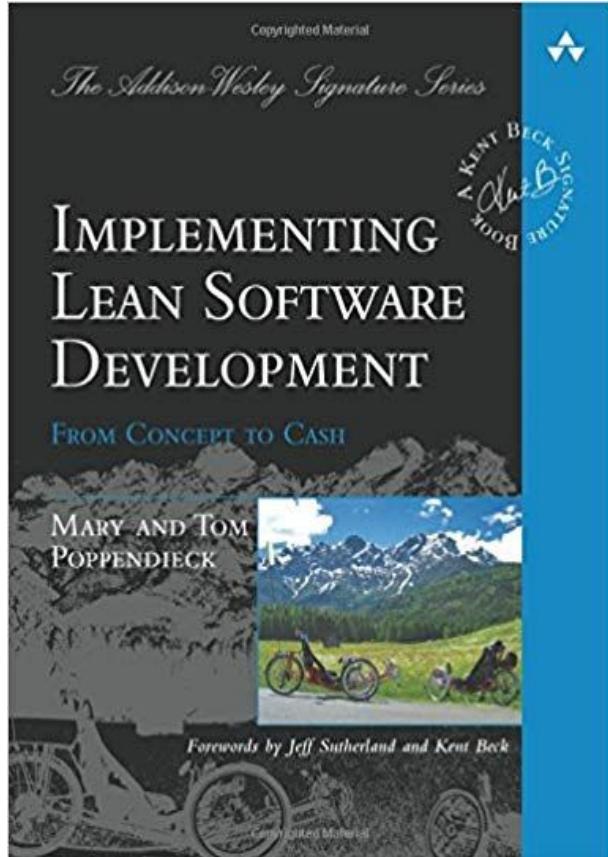
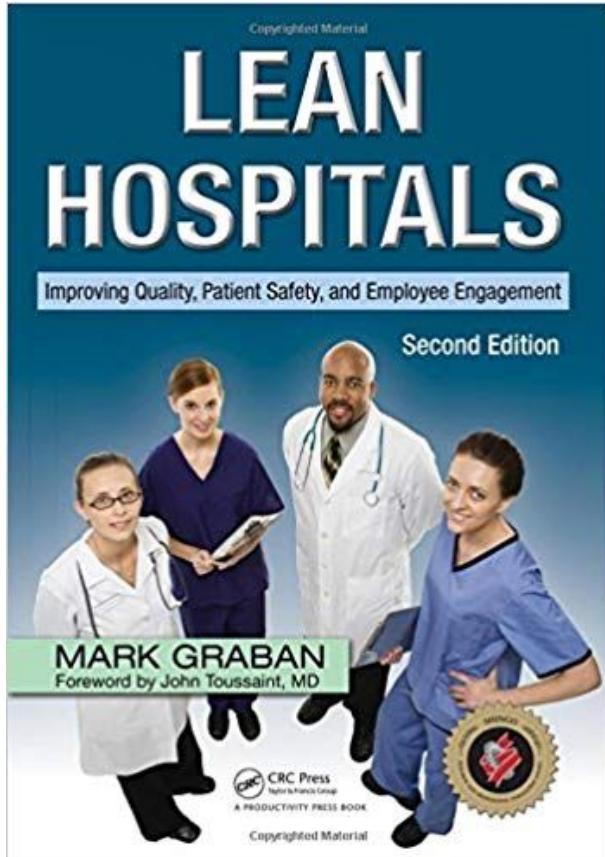


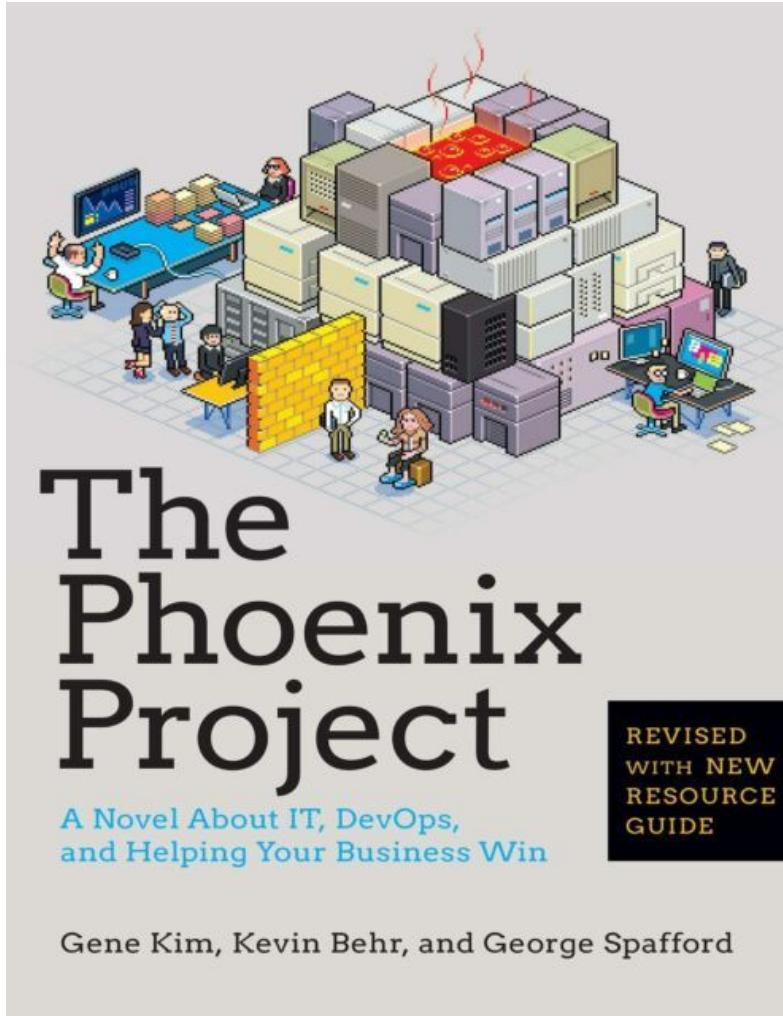
Lean



LEAN MANUFACTURING







#1 book about Devops.

DevOps == LEAN



vs



Assumption 1: A mature organization looks at the whole system; it **does not** focus on optimizing disaggregated parts.

Assumption 2 A mature organization focuses on learning effectively and empowers the people who do the work to make decisions.





Why do it at all ?
Remove Waste

#1 Built-in Quality



#2 Empower the Team



#3 Deliver as Fast as possible



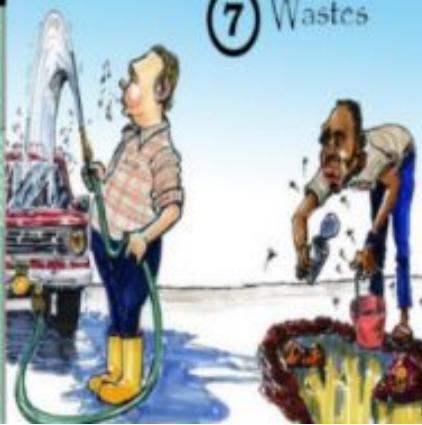
#4 Decide as Late as Possible



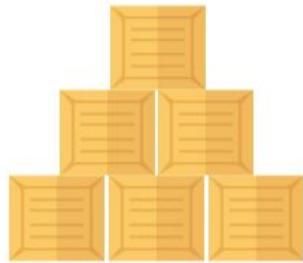
#5 Optimize the Whole



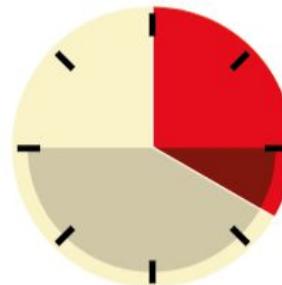
⑦ Wastes



7 Wastes of Lean



Inventory



Waiting



Defects



Overproduction



Motion



Transportation

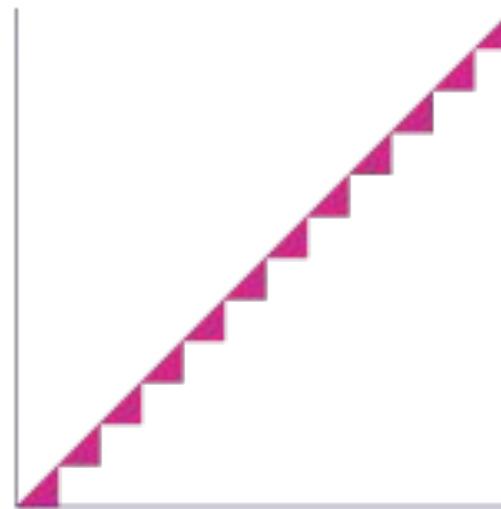


Over-processing

Large Batches

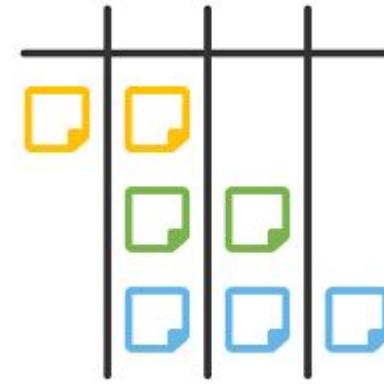


Small Batches



 Queue Size

Kanban



The Way I See It # 168

The human brain is the only object in the known universe that can predict its own future and tell its own fortune. The fact that we can make disastrous decisions even as we foresee their consequences is the great, unsolved mystery of human behavior. When you hold your fate in your hands, why would you ever make a fist?

-- Daniel Gilbert
Psychology professor and author of
Stumbling on Happiness.

Decaf

Shots

Syrup

Milk

Custom

Drink



This is the author's opinion, not necessarily that of Starbucks.
To read more or respond, go to www.starbucks.com/wayiseit.

Made with
10% post-consumer
recycled fiber.

Careful, the beverage you're
about to enjoy is extremely hot.

NO 422PCF 0000 0-00 20.5oz (605.8ml) SPECIAL DESIGN HOT CUP
MKT-002-168V2006 MADE IN THE USA C F M 10

Daniel Gilbert



“Kanban is an approach to change how the manager think / behave”

1. Visualize the Workflow
2. Limit WIP
3. Manage Flow
4. Make Process Policies Explicit
5. Improve Collaboratively (using Models and the Scientific Method)

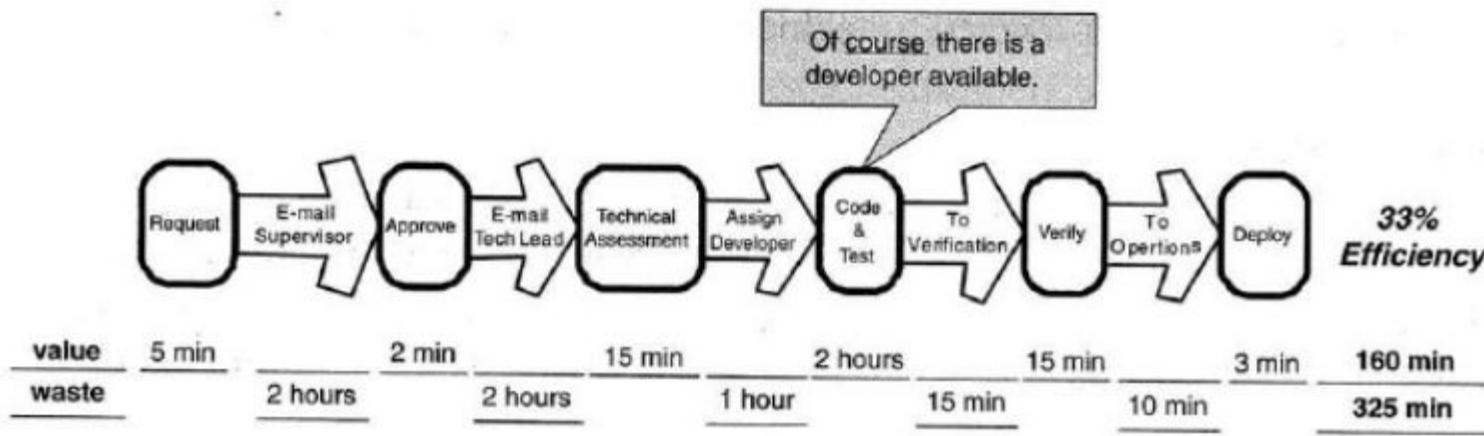
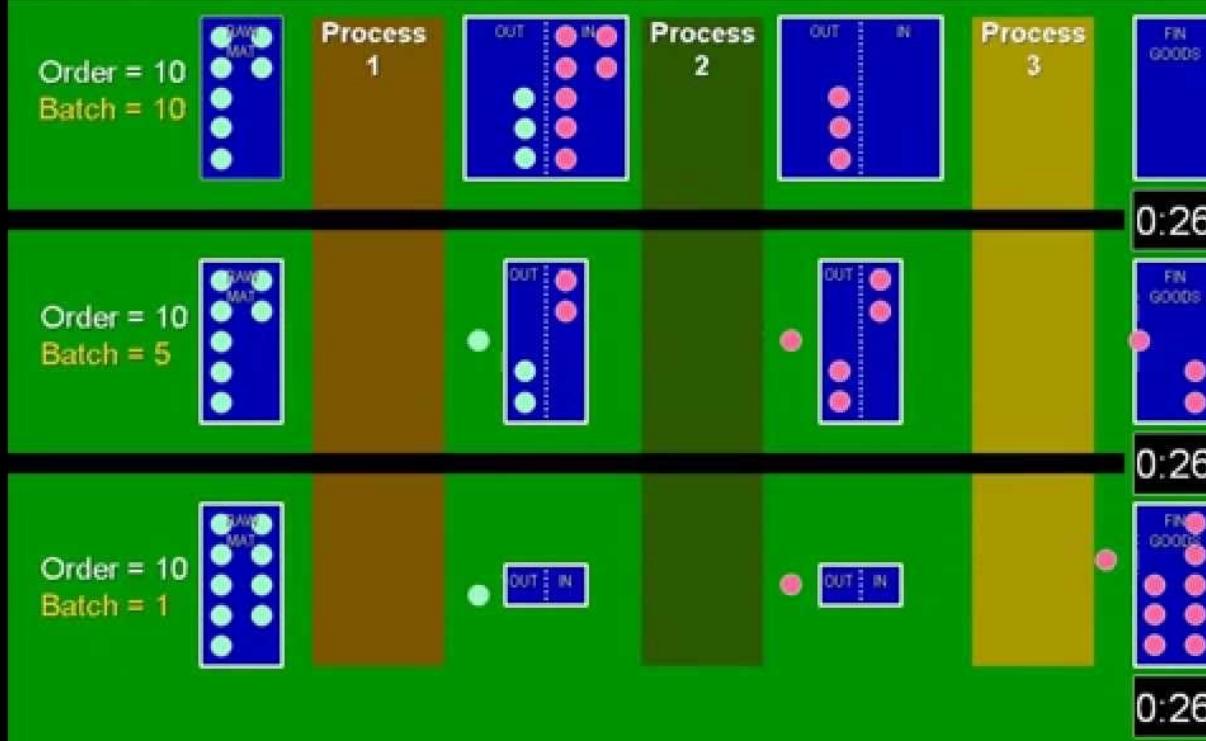


Figure 4.3 *Value stream map of a small, high-priority feature change request—Organization A*

HELLO
my name is

Names Experiment

Flow – One piece flow versus Batch Production



Courtesy: BMGI

<https://www.youtube.com/watch?v=JoLHKSE8sfU>





			
<p>Wip Max</p>	<p>Bug Hunt</p>	<p>Dojo</p>	<p>Wip 2</p>
		 One Piece Flow	 Dedo duro da semana
<p>Boi de Piranha</p>	<p>Pomodoro</p>		
	 One Piece Flow +  Pomodoro	 TDD Ping Pong	 Fireman

Visualized workflow



Inbox	Specification	Ready for Development	Development			Code review	Test locally	Test on PreProduction	Ready for Release (3 pm tuesday)
5	2	2	Planned	In progress	Done	in progr.	Done	in progr.	Done
Write Start Date	Accept Criteria!	Plan pairing	Refactor	TDD					Released: -Remove tickets -Write end date -Review deploy - Update CFD, Defect rate and Cycle time
									Measure flow

WIP limit = 2

Explicit policies

Cover:
Unitest.
Int.. Test
Code Coverage
Depl. issue

Tester and Product Owner need 10 min. preparation

Only Core Functionality

Kanban Metrics: Cumulative Flow Diagram



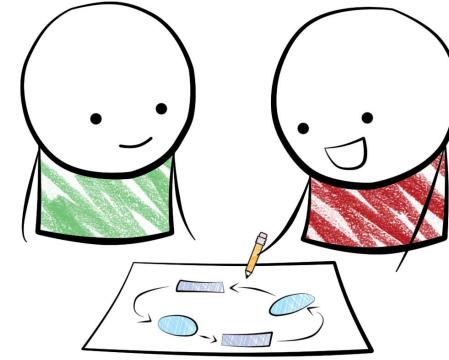
2014 Kanban Litmus Test

1. Have managers changed their behavior?
2. Has the customer interface changed?
3. Has the customer contract changed?
4. Has the service delivery business model changed?

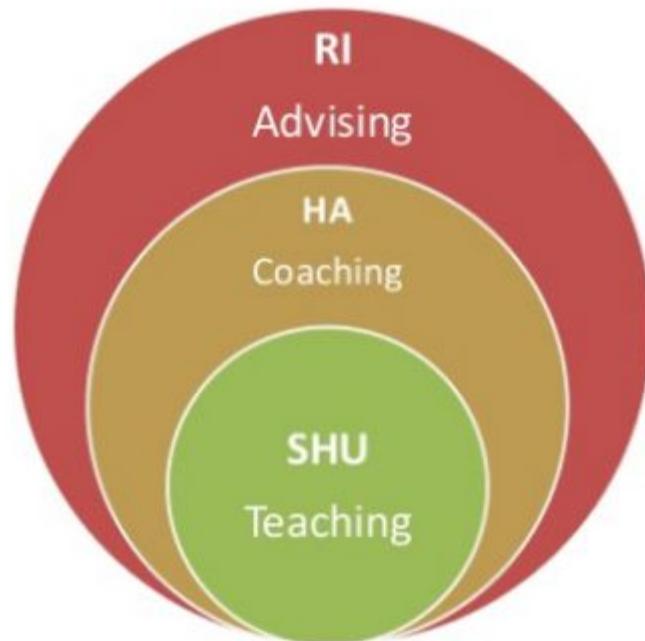
If you can't answer yes to at least 2 of these questions you aren't doing Kanban yet!



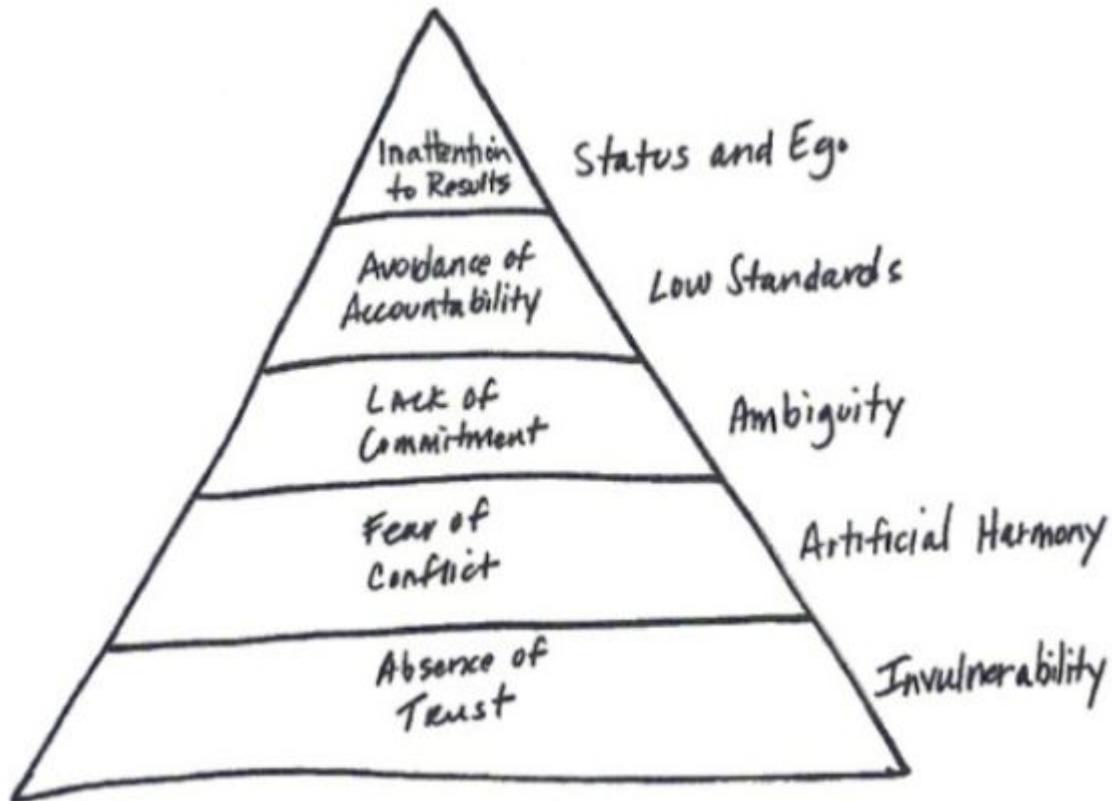
Agile Coaching

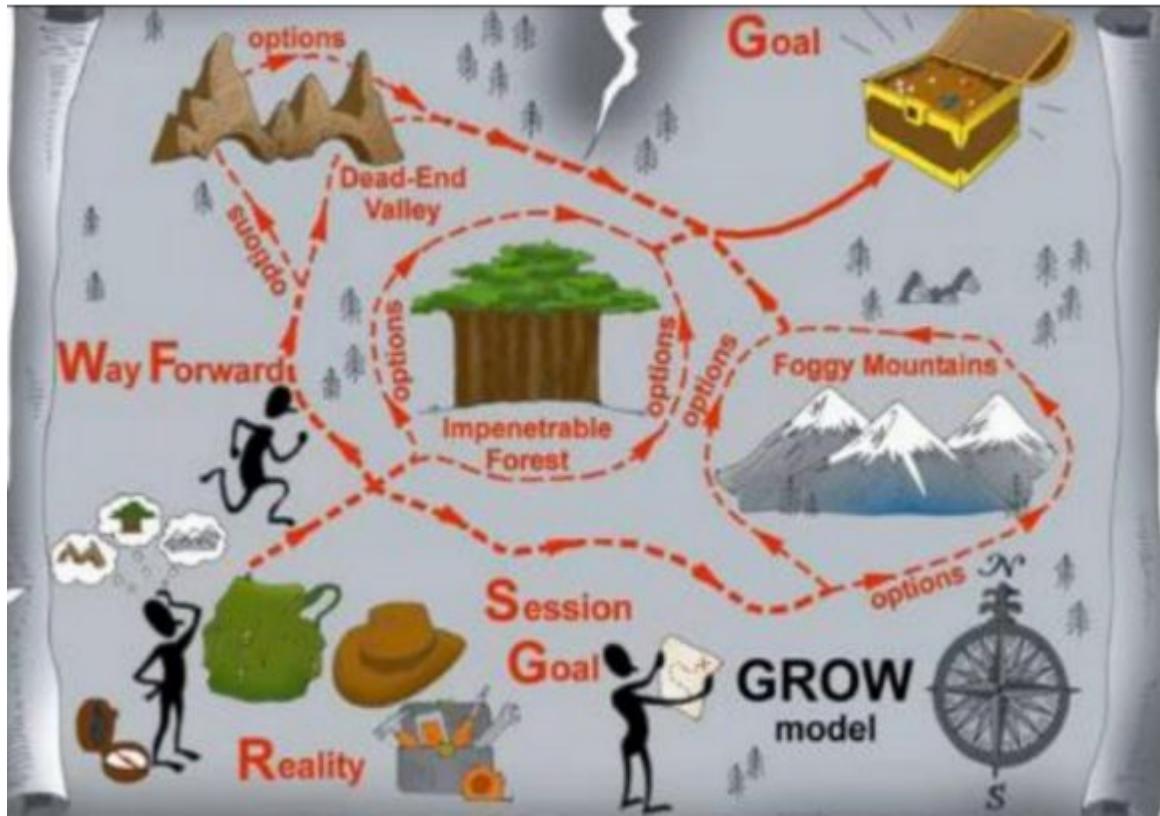


守破離









Id	Start	End	Diff Date	Type	Points	Backlog	Blocked	Design	Test&Code	CR	CR Ajustes	Acceptance	Real bug	Root cause
[US-1] [Jasper] [Reporting] [CSV]	8/11/2014	8/11/2014	1	Bug	1	0	0	0	4:00	0	0	0	Y	Endpoint configuration
[US-2] [Jasper] [Reporting] [PDF]	8/11/2014	8/11/2014	1	Bug	1	0	0	0	8:00	0	0	0	N	Not a bug. A feature
[US-3] [Jasper] [Reporting] [PDF]	8/12/2014	8/15/2014	4	Bug	2	0	0	0	14:00	1	0	0	Y	Issue on jasper
[US-4] [Jasper] [Reporting] [PDF]	8/14/2014	8/14/2014	4	Bug	1	0	0	0	7:00	0	0	0	Y	Environment instability
[US-5] [Jasper] [Reporting] [PDF]	8/14/2014	8/14/2014	1	Bug	1	0	0	0	2:00	0	0	0	N	Cache not reflecting DB (outdated)
[US-6] [Jasper] [Reporting] [PDF]	8/15/2014	8/18/2014	2	Bug	2	0	0	0	7:00	0	0	0	N	Not Reproducible
[US-7] [Jasper] [Reporting] [PDF]	8/20/2014	8/20/2014	1	Tech Debt	0,5	0	0	0	2:00	0	0	0	-	-
[US-8] [Jasper] [Reporting] [PDF]	8/20/2014	8/20/2014	7	Tech Debt	5	0	0	0	52	0	0	0	-	-
[US-9] [Jasper] [Reporting] [PDF]	8/22/2014	8/22/2014	11	Bug	5	0	0	0	80	0	0	0	Y	-
[US-10] [Jasper] [Reporting] [PDF]	8/13/2014	8/23/2014	8	Bug	3	0	0	10:00	0	0	0	Y	Some data is missing, NPE thrown	
[US-11] [Jasper] [Reporting] [PDF]	8/15/2014	8/20/2014	11	Bug	3	0	0	0	5:00	0	0	0	Y	Environment instability
[US-12] [Jasper] [Reporting] [PDF]	8/22/2014	8/29/2014	6	Bug	1	0	0	0	1	0	0	0	Y	Not Reproducible
[US-13] [Jasper] [Reporting] [PDF]	8/25/2014	8/29/2014	5	Bug	1	0	0	6	14	0	0	0	Y	Data corrupted
[US-14] [Jasper] [Reporting] [PDF]	8/25/2014	8/29/2014	5	Bug	1	0	0	6	14	0	0	0	Y	Data corrupted
[US-15] [Jasper] [Reporting] [PDF]	8/15/2014	9/1/2014	12	Story	3	0	0	10	40	1	0	0	-	-
[US-16] [Jasper] [Reporting] [PDF]	8/28/2014	9/1/2014	3	Story	3	0	0	0	14	0	0	0	-	-
[US-17] [Jasper] [Reporting] [PDF]	8/25/2014	9/3/2014	8	Bug	3	0	0	5	10	0	0	0	Y	-
[US-18] [Jasper] [Reporting] [PDF]	8/18/2014	9/1/2014	12	Story	3	0	0	10	40	1	0	0	-	-
[US-19] [Jasper] [Reporting] [PDF]	8/2/2014	9/8/2014	5	Story	3	0	0	8	22	0	0	0	-	-
[US-20] [Jasper] [Reporting] [PDF]	9/8/2014	9/8/2014	1	Bug	0,5	0	0	3	0	0	0	0	N	-
[US-21] [Jasper] [Reporting] [PDF]	9/1/2014	9/9/2014	7	Story	3	0	0	2	48	0	0	0	-	-
[US-22] [Jasper] [Reporting] [PDF]	8/25/2014	9/8/2014	12	Story	5	0	0	8	70	0	0	0	-	-
[US-23] [Jasper] [Reporting] [PDF]	9/10/2014	9/12/2014	3	Story	1	0	0	2	9	0	1	0	-	-
[US-24] [Jasper] [Reporting] [PDF]	9/8/2014	9/19/2014	10	Story	3	0	0	48	24	0	0	0	-	-
[US-25] [Jasper] [Reporting] [PDF]	9/8/2014	9/19/2014	10	Story	3	0	0	48	24	0	0	0	-	-
[US-26] [Jasper] [Reporting] [PDF]	9/3/2014	9/15/2014	9	Story	0	0	0	40	5	0	0	0	-	-
[US-27] [Jasper] [Reporting] [PDF]	9/11/2014	9/19/2014	7	Story	3	0	0	5	30	0	0	0	Y	Logging
[US-28] [Jasper] [Reporting] [PDF]	9/19/2014	9/23/2014	3	Story	2	0	0	16	0	0	0	0	-	-
[US-29] [Jasper] [Reporting] [PDF]	9/9/2014	9/23/2014	11	Bug	0	0	0	10	0	0	0	0	N	-
[US-30] [Jasper] [Reporting] [PDF]	9/22/2014	9/23/2014	2	Bug	1	0	0	8	8	0	0	0	Y	Japanese encoding
[US-31] [Jasper] [Reporting] [PDF]	9/22/2014	9/25/2014	4	Bug	1	0	0	8	0	0	0	0	N	-
[US-32] [Jasper] [Reporting] [PDF]	9/26/2014	9/26/2014	1	Bug	1	0	0	3	4	0	0	0	Y	Wrong functionality
[US-33] [Jasper] [Reporting] [PDF]	9/24/2014	9/25/2014	2	Bug	3	0	0	0	1	0	0	0	N	-

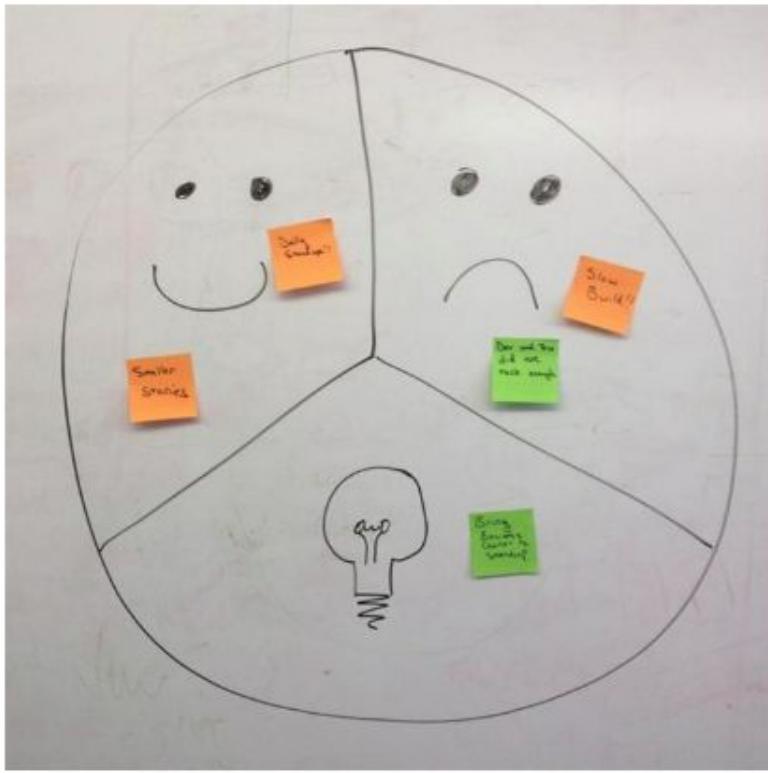


Setlists

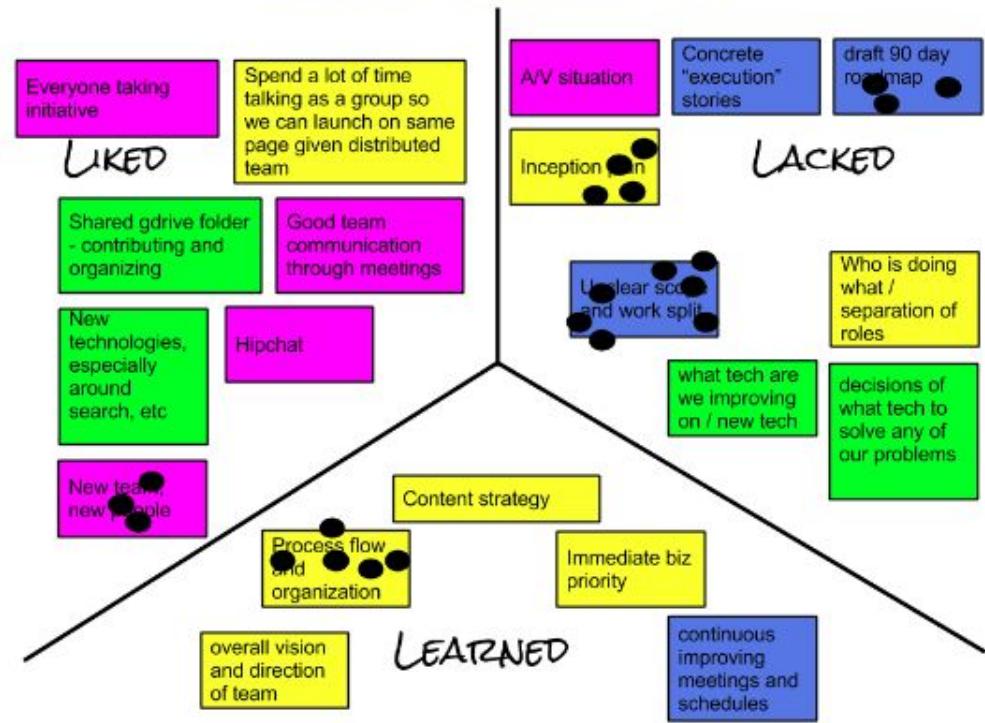
- Opening
- Ice-Breaker
- Process
- People
- Software
- Outside
- Aha-Moment
- ROTR



designed by freepik.com



the 3 Ls activity: Liked – Learned – Lacked





Speedboat Retrospective

Propellers: what moves us forward



Life Preserver: What can save or help us



Anchors: What holds us back



Rocks: Where can we crash

Success and Failure



It's not about the first-mover advantage;
It's about the fast-learner advantage.

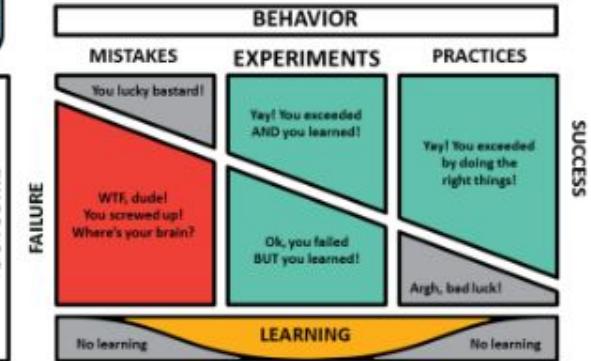


Don't change things;
run experiments.

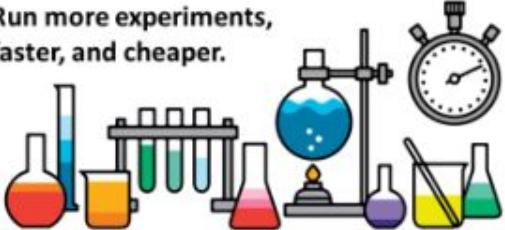


Create a safe-to-fail environment.

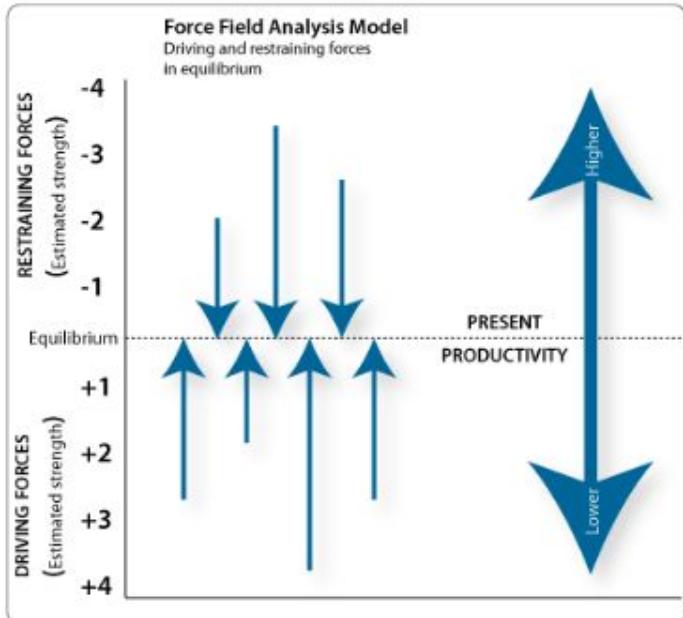
CELEBRATION GRID



Run more experiments,
faster, and cheaper.



Force Field Analysis Model
Driving and restraining forces
in equilibrium



Vision:

Current State	Target State	Who's Affected?
----------------------	---------------------	------------------------

Working with this change

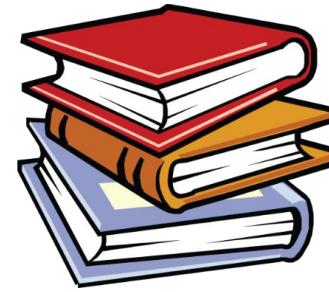


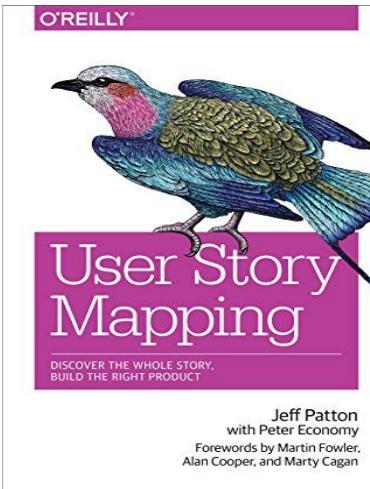
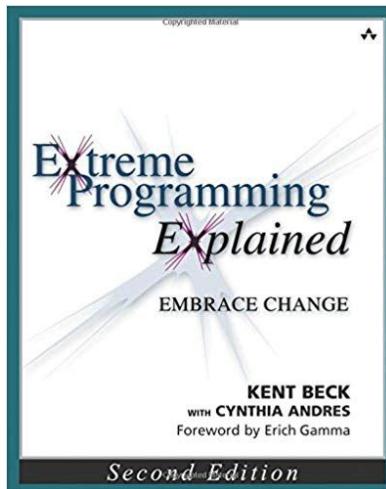
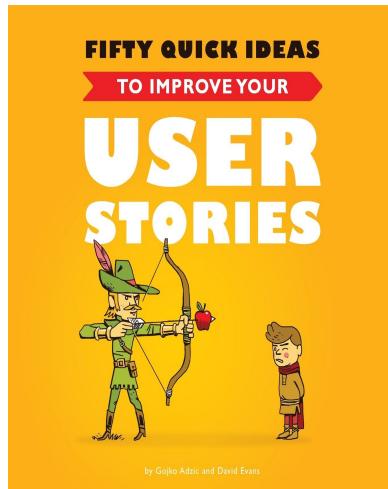
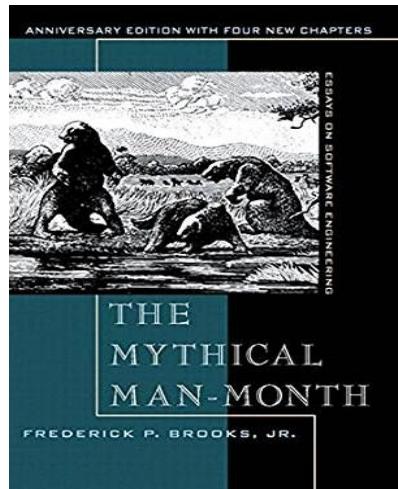
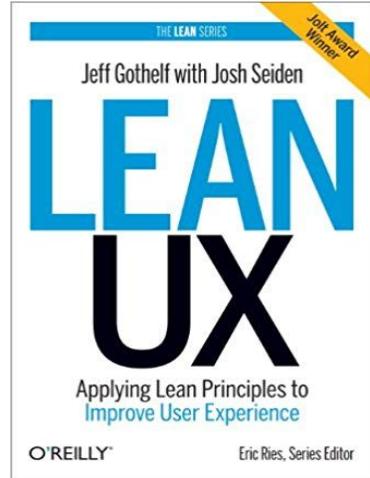
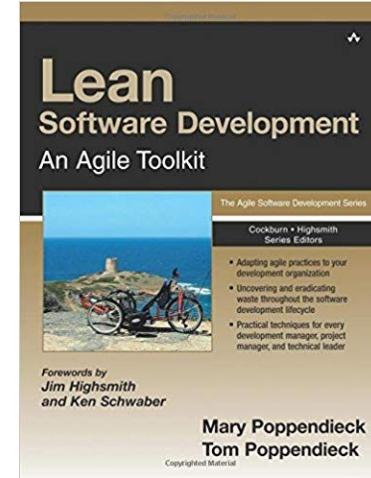
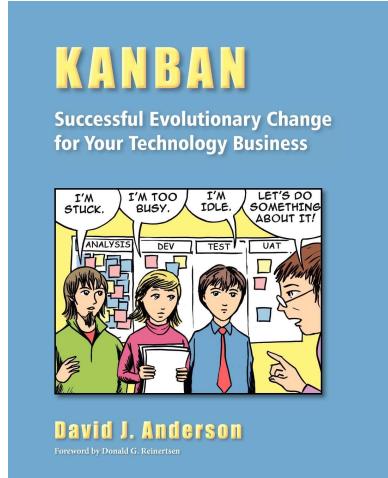
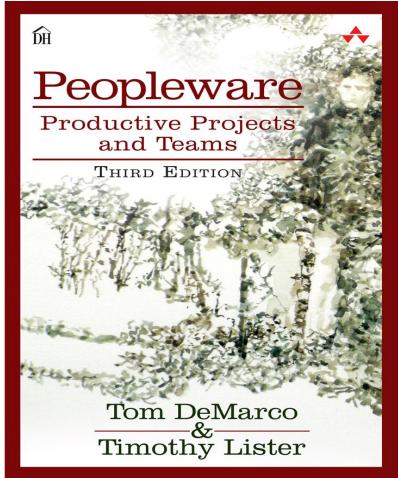
Working against this change

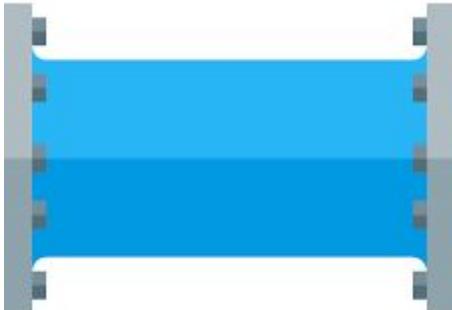


Backlog	Next	Prepare	In Progress	Review
----------------	-------------	----------------	--------------------	---------------

Books







Kanban Training

DIEGO PACHECO