

Transforming Planning: Agile and Flooding

Pamelyn Klepal Shefman, PhD
Director of Assessment and Planning
University of Houston

Why plan?

“Always plan ahead.
It wasn't raining when
Noah built the ark.”

~Richard Cushing



Overview

- Basics of Agile concepts
 - Not a deep dive
 - Disclaimer: no plans to create software developers after this session
- Share personal flood lessons related to agile
- Impart that everything we might need does not has to be from “our fields” to be useful

What is “Agile”?



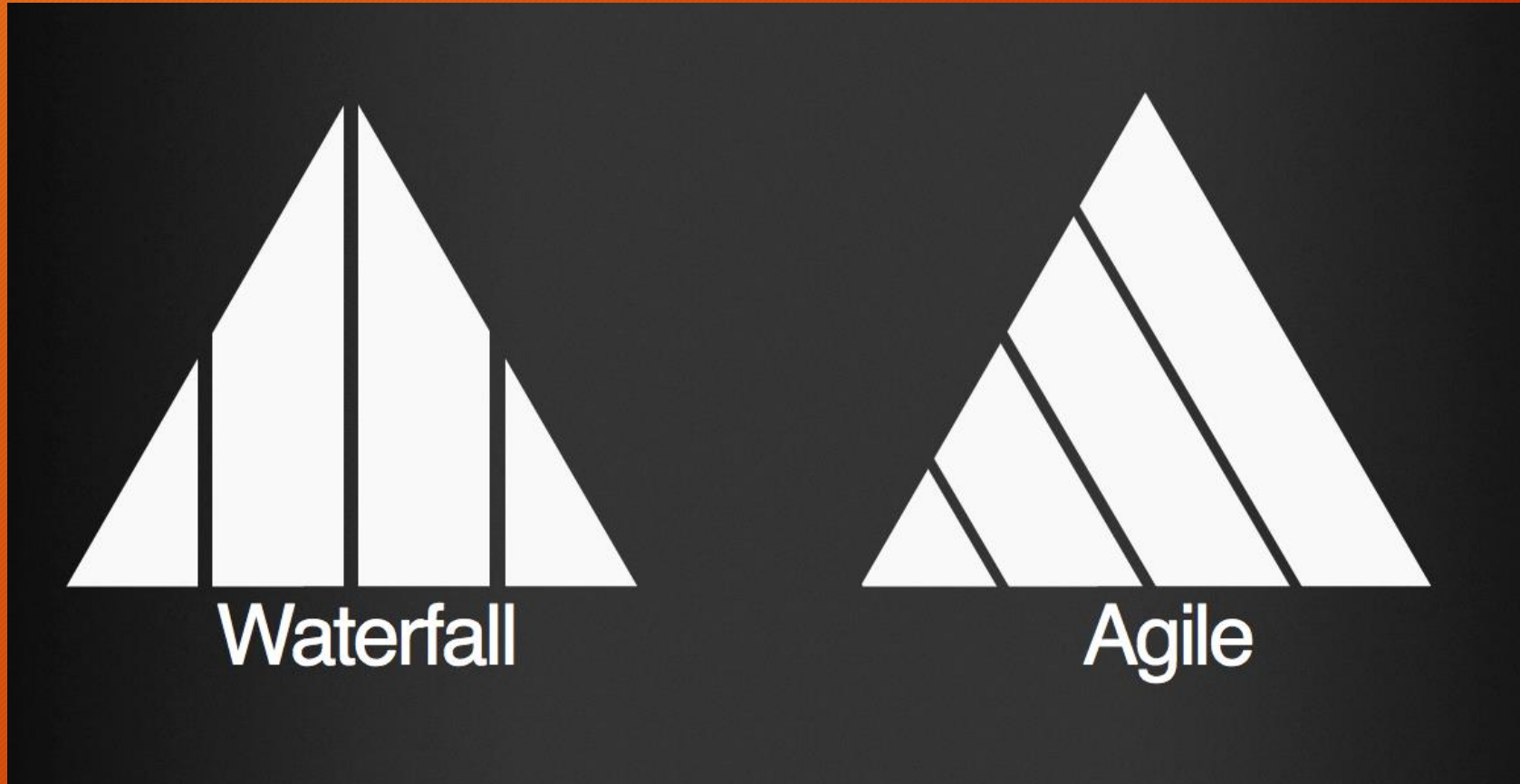
Agile software development describes a set of values and principles for software development under which **requirements and solutions evolve** through the collaborative effort of self-organizing cross-functional teams.

What is “Waterfall”?



The waterfall model is a **sequential design process** in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, production, and maintenance.

Waterfall verses Agile

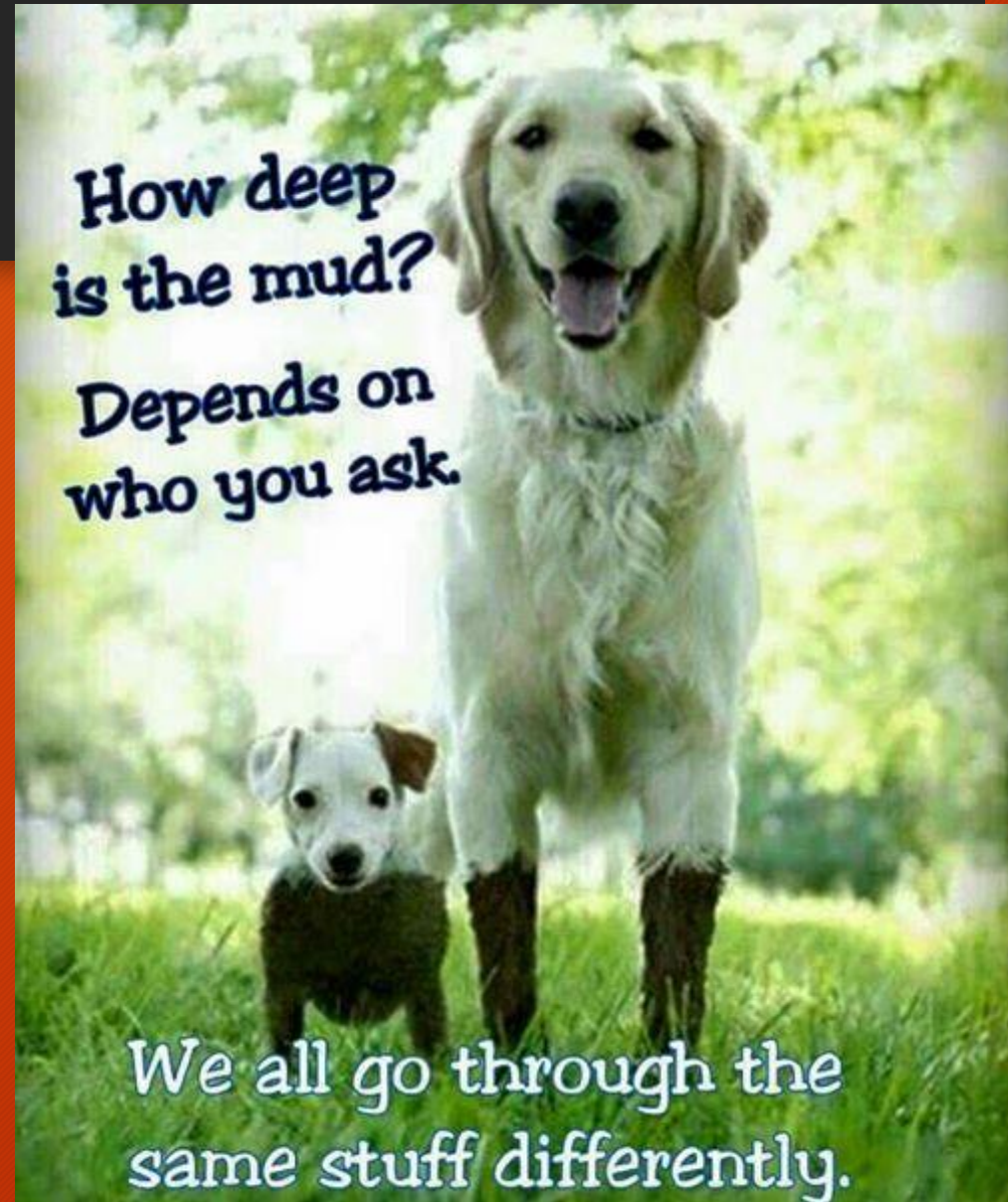


Who am I?

- Director of Assessment and Planning
- Futuristic * Strategic * Activator * Self-Assurance * Individualization
- Flood Participant in inches and gallons
 - 27” Memorial Day 2015 (162 Billion Gallons)
 - 13” Tax Day 2016 (240 Billion Gallons)
 - -5” Harvey 2017 (9 Trillion Gallons)
(Elevated house 1 day before - home office was impacted)

Flooding

Each story is unique, different,
and important



Room flood poll

- How many have been personally impacted in recent events?
- How many have witnesses rising water in your own home?
- Of those who had rising water did you have a “plan”?



Context of flooding

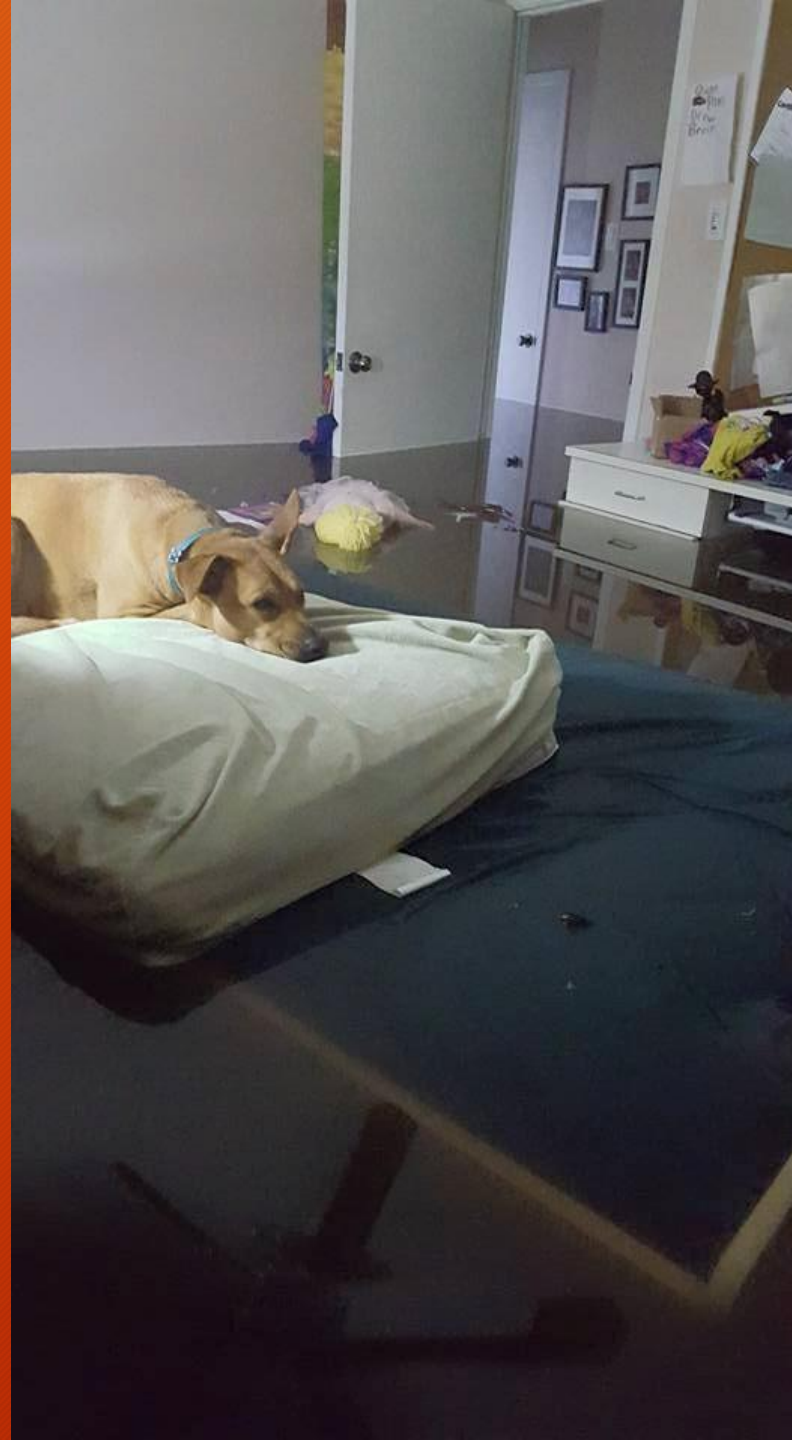
- Trigger warning:

Personal images of my flooded home are next and throughout.











Agile Concepts that you can learn from flooding

1. Backlog
2. Prioritized Backlog
3. YAGNI
4. Communications & Teams
 - Frequent Team Communication
 - Self-organizing, cross-functional teams
5. Processes
 - Iterations
 - Demo, retro, sprint planning

A backlog is a list of features or tasks which the team maintains and which, at a given moment, are known to be necessary to complete a project.

SOS - Save our stuff



Prioritized backlog

- How might you prioritize to Save Your Stuff (i.e. save our stuff)?
 - Floor -> Counters
 - Irreplaceable -> Common
 - Expensive -> Cheap
 - Useful -> Unnecessary
 - Nearest -> Farthest
 - Front of House -> Back of House



Creating a backlog in your planning

- Reminder: Backlog is a prioritized list of “functionality”
- What is “core” to what we do that we must do well?

Implicit verses Explicit Priorities

Implicit Priority

Ordering the backlog correlated toward easiness or convenience.

- Floor -> Counters
- Nearest -> Farthest
- Front of House -> Back of House

Explicit Priority

Ordering the backlog correlated with the value it delivers to the business.

- Irreplaceable -> Common
- Expensive -> Cheap
- Useful -> Unnecessary

YAGNI You Aren't Gonna Need It

Is a mantra that some functionality/capability that could be built for the future should not be done now because

"you aren't gonna need it"
(yet or maybe ever again).



YOU AIN'T GONNA NEED IT

Don't waste resources on what you *might* need.

There were some things that were not worth saving.



Know what you NEED...



Define your goals based on needs



Define your goals based on needs



Communications & Teams

- Self-organizing, cross-functional teams
- Frequent Team Communication

Know your team and communications

Self-organizing

A group of motivated individuals, who **work together** toward a goal, have the ability and authority to make decisions, and readily adapt to changing demands.

Cross-functional

Cross-functional just means that the team as a whole has **all the skills** needed to build the product, and that each team member is willing to do more than just their own thing.

Know your team



Know your team and communications



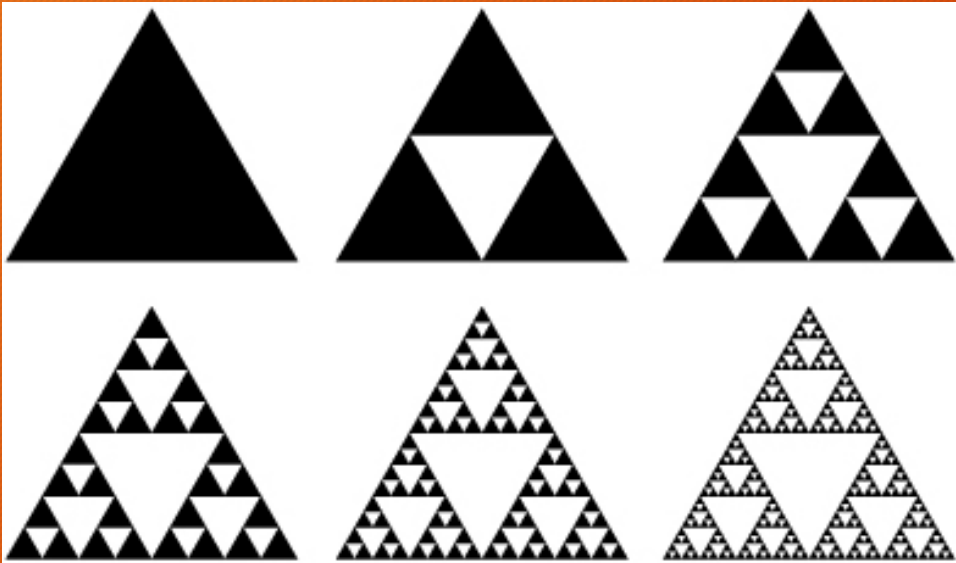
Know your team and communications



Processes

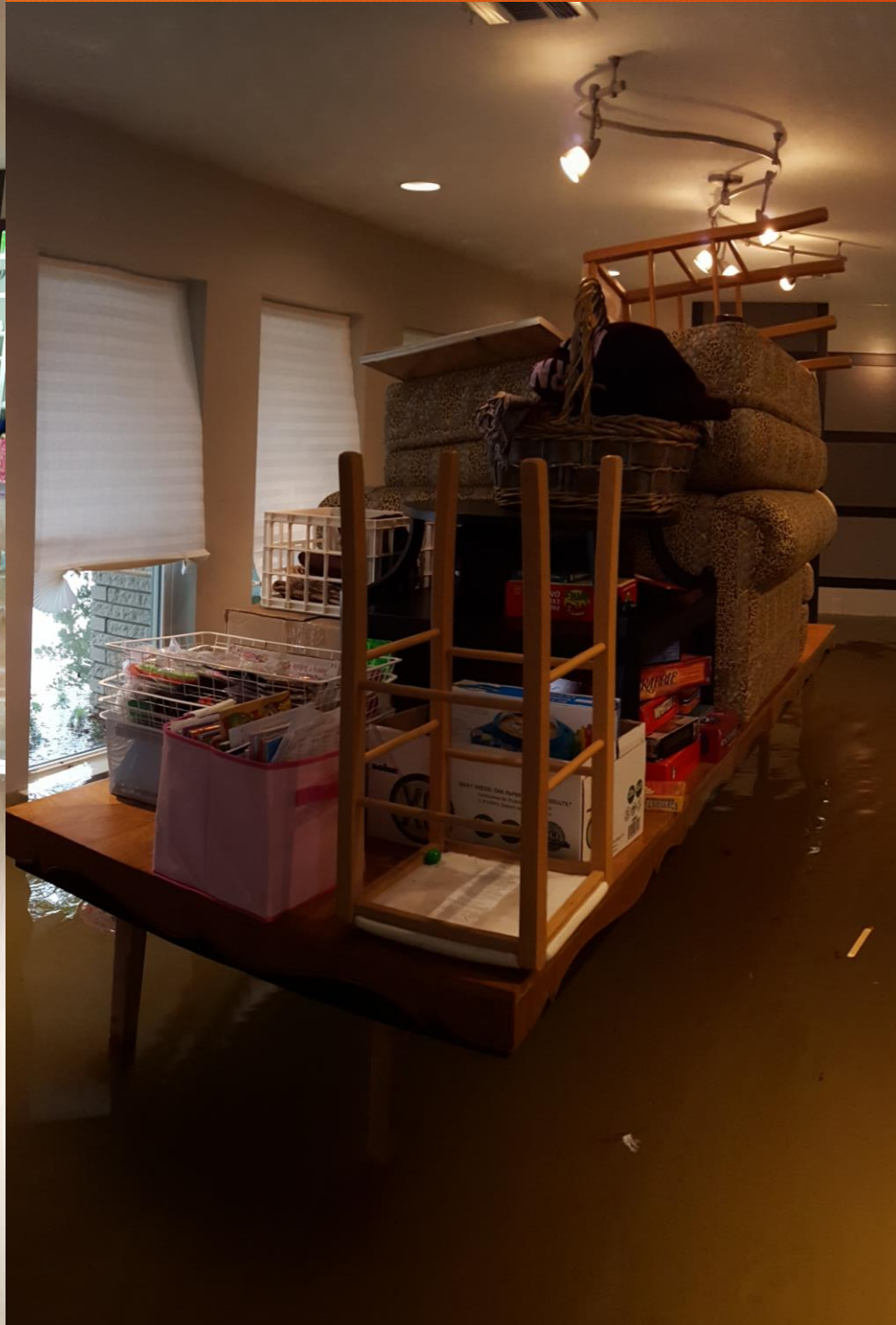
- Iterations
- Not covered in flooding but terms that are good to know for planning:
 - Demo, retro, sprint planning

Iterations



A repeating set of work to get done before reflecting on the progress, objectives, and external influences.

Iterations are not final products but are “done enough” to add value.



Other Agile Concepts

- Sprints
 - Commitment to highest priorities
- Demo
 - Promotion of discussion across stakeholders through demonstration
- Retro
 - Reflection, continuous improvement

PROS

Waterfall is **well-disciplined**



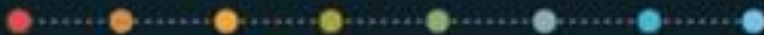
It has to start with **complete requirements**



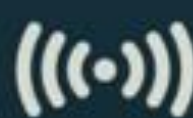
Great for projects with **defined phases**



WATERFALL METHODOLOGY (PREDICTIVE MODEL)



CONS



Early feedback is **absent**



Response to change is **slow**



Risk of **high cost** if a requirement is **missing**

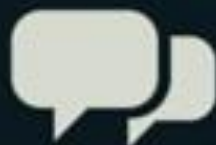




Focuses more on
**customer
collaboration**



Upholds individuals &
interactions



It advances towards a
**working
software**



It puts emphasis on
Fast delivery



AGILE METHODOLOGY (ADAPTIVE MODEL)



Lack of
documentation



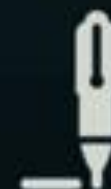
Difficulty in building
great design



Lack of emphasis in
necessary **designing**



Can easily get off track
if goals are **unclear**



Difficulty in assessing
efforts required at the
beginning

April 22, 2016 - 4 days after TD flooding

Your present plans will be
successful.



5014

5014

5014

5014



5014

KOLTER
COUGARS

ARKITEKTURA
OF INTERIORS, INC.
(281) 957-9192
FEMA ELEVATION
CONTRACTOR

