
SOFTWARE PROJECT MANAGEMENT - 2

WEEK 3 – 09/11/19

THINGS TO DO

- Submit recitation 1 before your recitation this week.
- Project Milestone 1 will be posted this Friday. Due on Friday, September 27

RECITATION GUIDELINES

- Attendance is mandatory for ALL recitations to be eligible for credit
- The exercises assigned in the recitation have to be completed and submitted on Canvas within the due date to receive full credit
- Even if you complete your exercise during the recitation, please upload the completed work on Canvas
- There is no partial credit for recitation. Its either full credit (100) or no credit (0)
- In case you miss a recitation due to an unforeseen circumstance, do let us know beforehand
- If you're unsure where you stand with a situation w. r. t. recitation, ASK to avoid confusion

- What is unacceptable?
 - Not attending recitation and submitting the exercises on Canvas
 - Attending the recitation, not completing the baseline task
 - Attending recitation, completing baseline task and not submitting the rest of the work on Canvas

CHALLENGES WITH WATERFALL



A successful project demands a detailed and accurate Work Breakdown Structure early-on



The scope of the project is locked down at that time, and should remain unchanged until solution delivery



Changes to scope, even when well-managed, will extend project timeline and costs



Once requirements are defined, customers are much less engaged during the design and construction phases



Customers must wait until the project is over to enjoy any tangible benefit from the project

WHAT IF?



What if we don't know?



What if they changed their minds?



What if their demand changes, but won't let me change the budget or timeline?

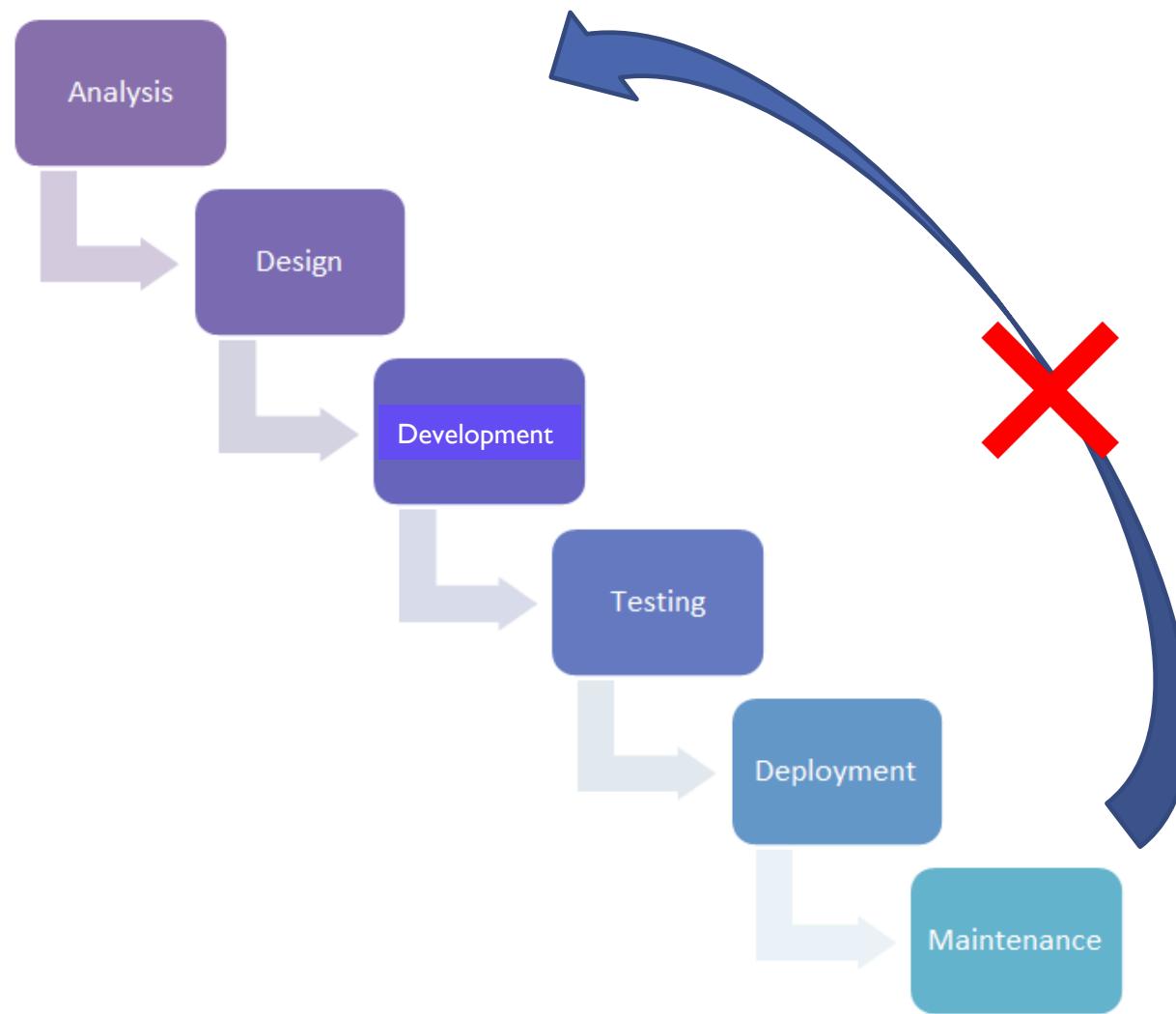


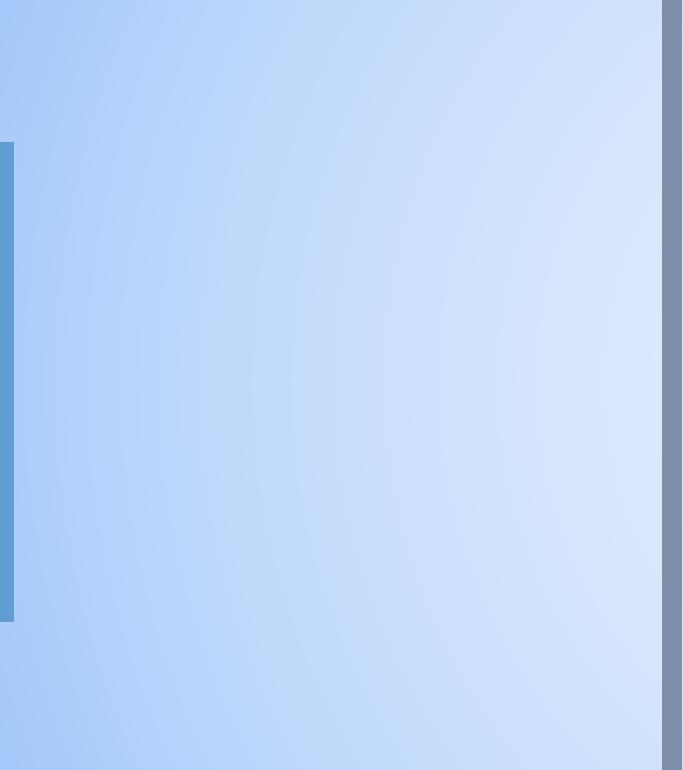
What if the customers are agitated about being out of the loop?



What if they want to see results before we are ready?

WHAT IS LACKING IN “WATERFALL” ?





HOW DO WE
FIX THAT ?

AGILE



AGILE



INDIVIDUALS AND INTERACTIONS

over

PROCESSES AND TOOLS



WORKING SOFTWARE

over

COMPREHENSIVE DOCUMENTATION



CUSTOMER COLLABORATION

over

CONTRACT NEGOTIATION



RESPONDING TO CHANGE

over

FOLLOWING A PLAN

12 KEY PRINCIPLES OF AGILE

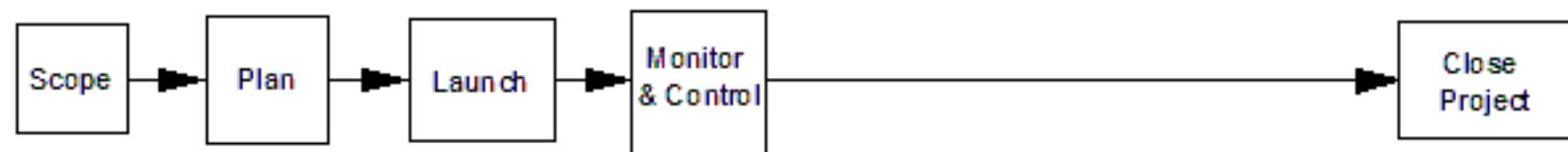
1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.
4. Business people and developers work together daily throughout the project.
5. Build projects around motivated individuals, give them the environment and support they need
6. The most efficient and effective method of conveying information with and within a development team is face-to-face conversation.

12 KEY PRINCIPLES OF AGILE

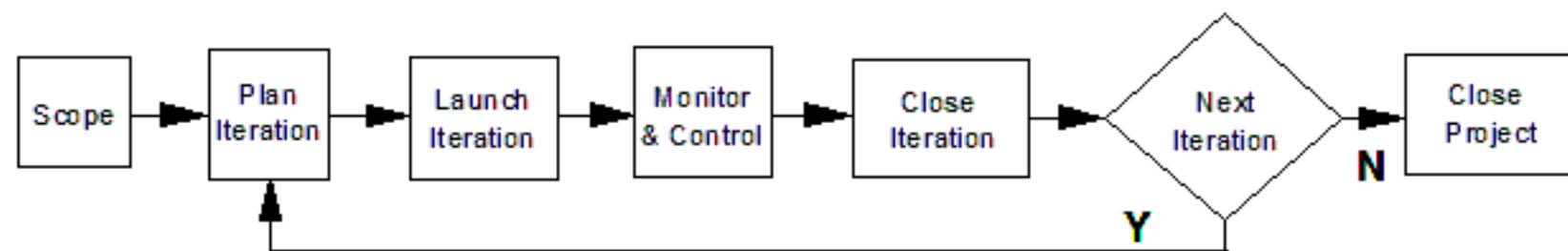
7. Working software is the primary measure of progress
8. Agile processes promote sustainable development. The sponsors, developers and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility
10. Simplicity—the art 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

WATERFALL VS. AGILE

Waterfall – linear model



Agile – iterative model



PROTOTYPING

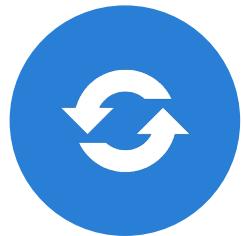
Build it, assess it, modify it, repeat

1. Continuous integration, verification, and validation of the product.
2. Frequent demonstration of progress to increase the likelihood that the end product will satisfy customer needs.
3. Early detection of defects and problems.

AGILE MODEL



FOCUS ON
CUSTOMER VALUE



ITERATIVE AND
INCREMENTAL
DELIVERY



EXPERIMENTATION
AND ADAPTATION



SELF-
ORGANIZATION



CONTINUOUS
IMPROVEMENT

SCRUM METHOD

- The product is composed of “Features” as described by one or more “User Stories”
- A feature delivers functionality to the customer according to the function described in a user story
- Features and user stories are prioritized by their perceived highest value
- Team tackles the highest priorities first
- Priorities of “feature backlog” are re-evaluated after each iteration
- Each “sprint” produces fully functional features
- Each feature involves analysis, design, build and test

USER STORIES

A templated statement about a specific user's expectation of the system*

- As a [stakeholder role]
- I want to [do some task]
- So that I can [achieve a goal]

Example

- As a student
- I want to enroll in a degree
- So that I can land a job

*Note: The template describes three elements of a user story. The order is not prescribed and can be changed to improve readability.

ROLES



Product Owner

Acts on behalf of the customer
Keeper of the feature backlog
Keeps the team focused on priority objectives
Final authority on requirements
Decides when a feature is complete



Team

Builds the feature
5 – 9 people
Self-organizing
No roles or titles
Input to sprint planning



Scrum Master

Facilitator
Resolves issues
Keeps team on process
More “coach” than “manager”

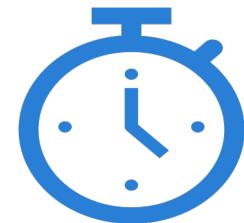
MEETINGS



Built around Sprints

A sprint runs a maximum of 4 weeks

A sprint team has 5-9 people



Daily Scrum

Same time & place, 15 minutes max

All Stand

Tasks & Blocks on white board

Questions

- ✓ What have you done since the last scrum?
- ✓ What will you do before the next scrum?
- ✓ What is blocking you?

SCRUM MASTER



- Does NOT assign daily tasks to team
- The team decides amongst themselves
- Master of the process, not of the team
- At the end of a sprint, team reviews the product/feature
- At the end of a sprint, team reviews the process in a “retrospective”

Product Backlog

Prioritized list of features/requirements

A prioritized list of work to be accomplished

Owned by the product owner

Sprint backlog

List of individual sprints

Tied to product backlog

Owned by the team

BACKLOG

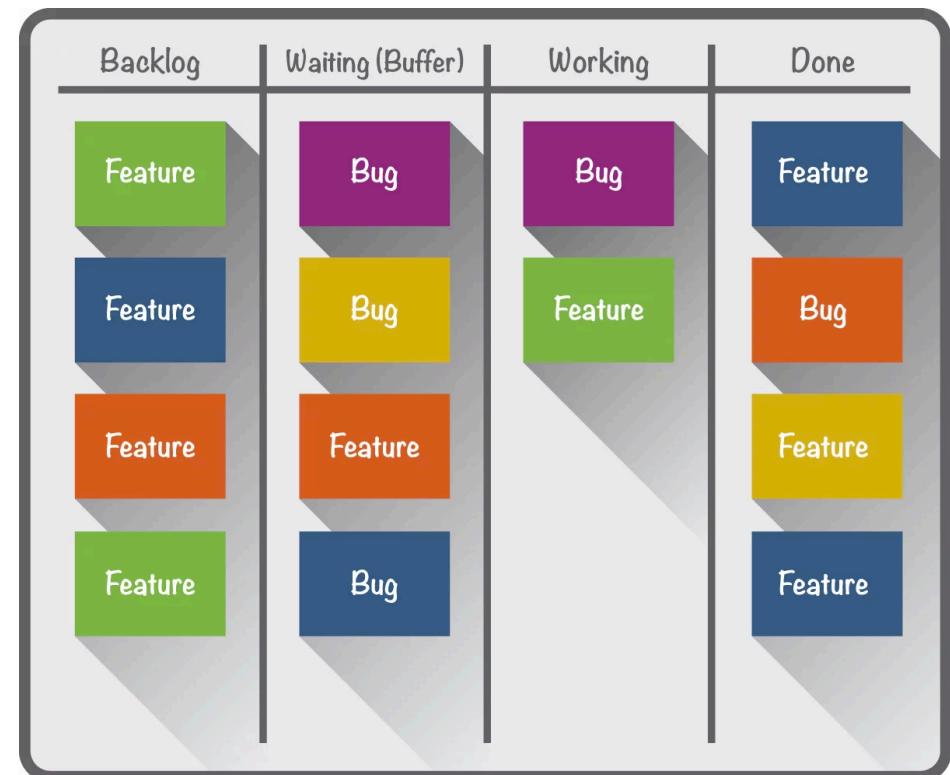


RESISTANCE TO AGILE

- Agile PM does not satisfy top management's desire for controlling budget, scope and schedule
- Many of the agile principles -- including self-organizing and intense collaboration – may run counter to corporate culture
- Requires a knowledgeable Scrum Master to keep teams focused on proper methods

KANBAN

- A method for managing work
- Visual, on a board
- Tracking work
 - Planned
 - In-Progress
 - Complete



SCRUM & KANBAN

- Both embrace principles of agile development
- Both encourage
 - early and frequent delivery
 - self-organized teams
 - continuous improvement
 - high quality
 - prioritizing of requirements based on business value

ADDITIONAL RESOURCES

Kanban overview:

- <https://www.youtube.com/watch?v=jf0tlbt9lx0>
- <https://www.youtube.com/watch?v=N3BoLRVXoI0>

Scrum overview:

- <https://www.youtube.com/watch?v=9TycLR0TqFA>

PROJECT MANAGEMENT TOOLS

Why a PM Tool?

- Document “deliverables” (a.k.a. features, requirements)
- Capture the WBS as features/tasks
- Plan the work
- Schedule the work
- Assign work to teams or individuals
- Plan sprints
- Track and report progress toward completion

Are you Agile? Your tool must

- support people over process
- facilitate collaboration

CREATING A PROJECT PLAN

GANTT chart

- WBS
- Predecessors/Successors
- Resource Assignments
- Duration
- Calendar



SOME PM TOOLS TO LOOK AT

- [Asana](#)
- [Freedcamp](#)
- [Trello](#)
- [Wrike](#)

(there are many more, but most cost \$\$ after a 30-day free trial)