

PMI-ACP Training



May 2020						
M	T	W	T	F	S	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Saturday 9th May 2020

Introductions

- Name
- Role
- Experience
- Prior exposure to Agile/Scrum



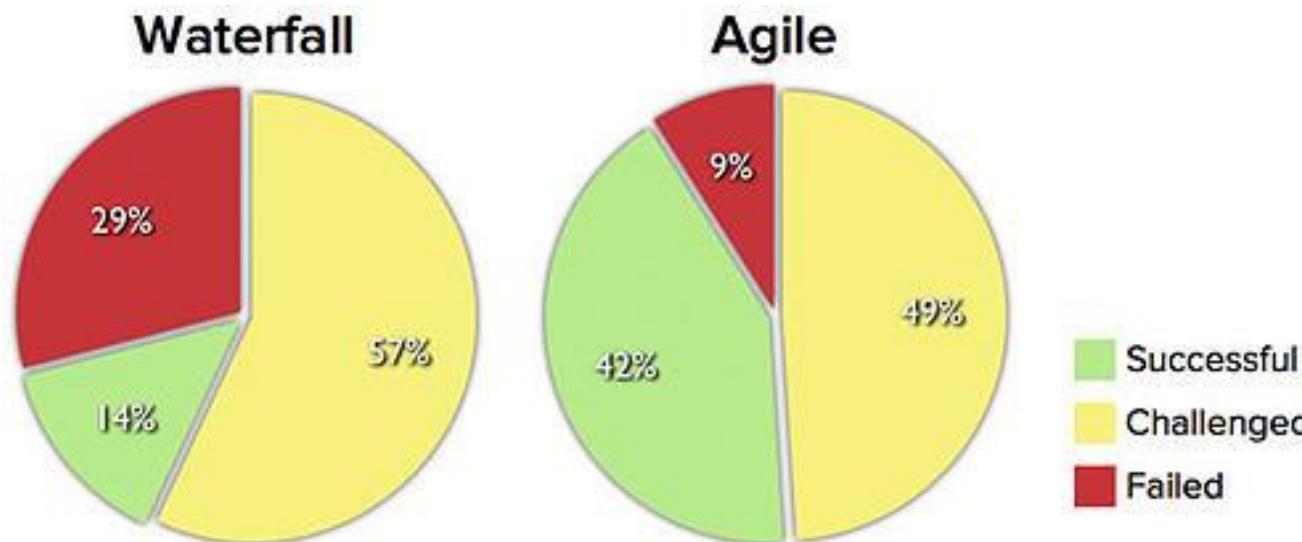
GROUND RULES

- Start on time
- End on time
- Laptops & Mobile Off/Silent Mode
- No Cross Conversation





Agile is 3 Times More Successful



Source: The CHAOS Manifesto, The Standish Group, 2012.

Agile projects are successful three times more often than non-agile projects, according to the CHAOS report from the Standish Group. The report goes so far as to say, **"The agile process is the universal remedy for software development project failure. Software applications developed through the agile process have three times the success rate of the traditional waterfall method and a much lower percentage of time and cost overruns."** The Standish Group defines project success as on time, on budget, and with all planned features. The study is based on projects from 2002 - 2010.

STATE OF SCRUM report

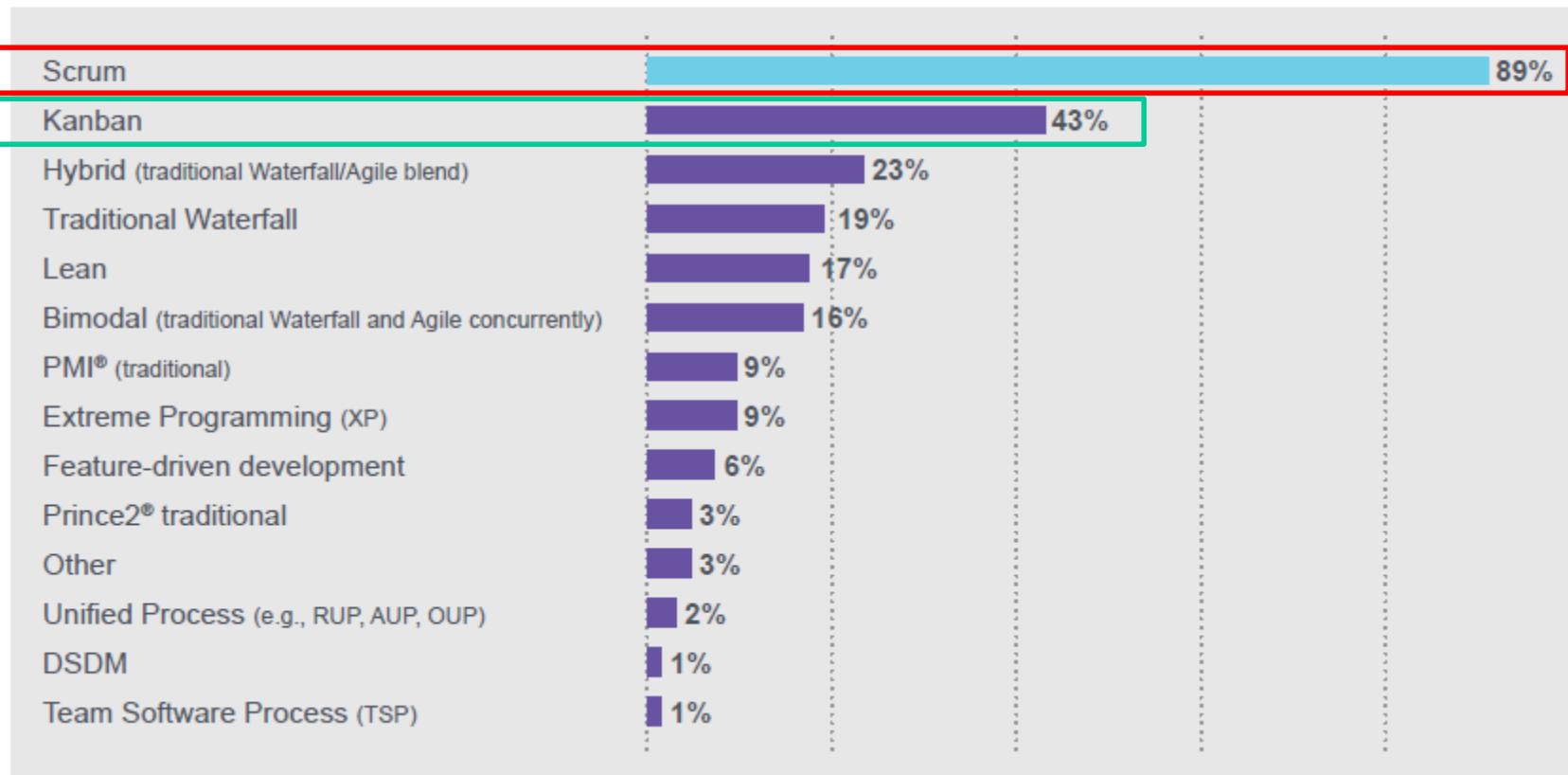
How the world is successfully applying
the most popular Agile approach to projects



2. Which Agile approach is your organization using?

(Multiple answers allowed)

Most respondents — 89% — report that Scrum specifically is the Agile approach or at least one of the Agile approaches used in their organization. (Ninety-two percent of respondents use any form of Scrum, including scaling Scrum frameworks, among their approaches.) Kanban is the second most common, followed by a hybrid approach, then traditional Waterfall. Lean, previously in third place at 21% in 2015, fell to 17% in 2016.



"The ability to prioritize, the ability to collaborate with a team, all these necessary skills and ancillary skills of facilitating and coaching will become more important in each work environment," he says. "And all of these will, over the course of the next five to 10 years, show up more in university or college education."

Indeed, that is already happening. There are K–12 schools in the United States, Canada, and Europe where children learn Agile practices. U.S. universities, such as Rutgers, the University of Virginia, and Northwestern University, offer courses in Agile or Agile project management. So do universities in Great Britain, Australia, and the Czech Republic.

But whether in education or business, it's adopting Agile practices, such as innovation and collaboration, that can make or break a successful Agile transformation.

"Companies are going to continue to see that their ability to innovate in order to compete is going to rely on them figuring out how to experiment in a very short time frame," Orrell says. "Those experiments will sometimes succeed and sometimes fail, but that's the nature of innovation."

"It's that innovation that's going to keep companies competitive."

The Future of Scrum

98% of respondents say they plan to use Scrum moving forward.

Copyrighted Material

SCRUM The Art of Doing Twice the Work in Half the Time

JEFF SUTHERLAND
Co-creator of Scrum
J.J. SUTHERLAND

Copyrighted Material

Project Life-cycle Approaches

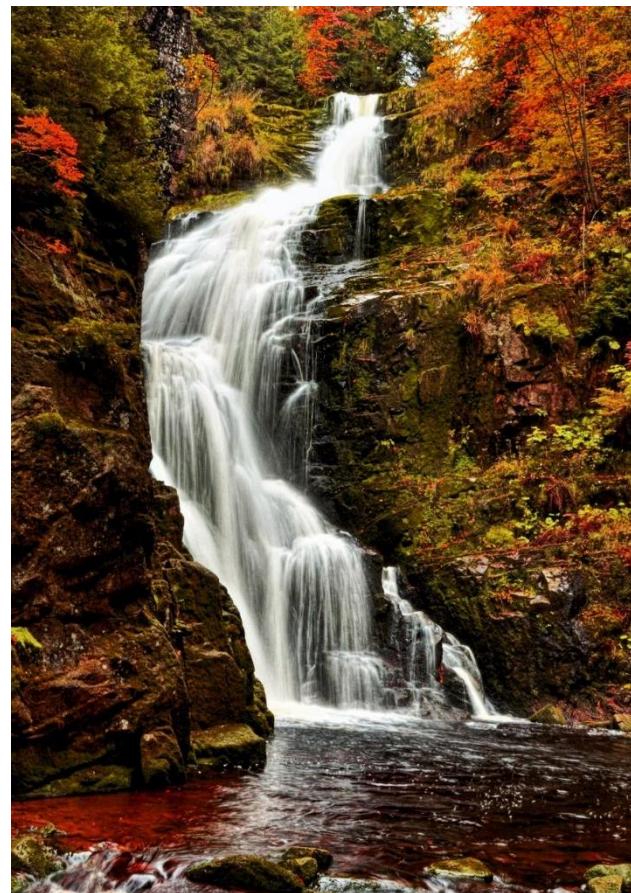
1. Predictive

2. Adaptive

3. Hybrid

1. Predictive life cycle

Requirements
clearly known
in advance



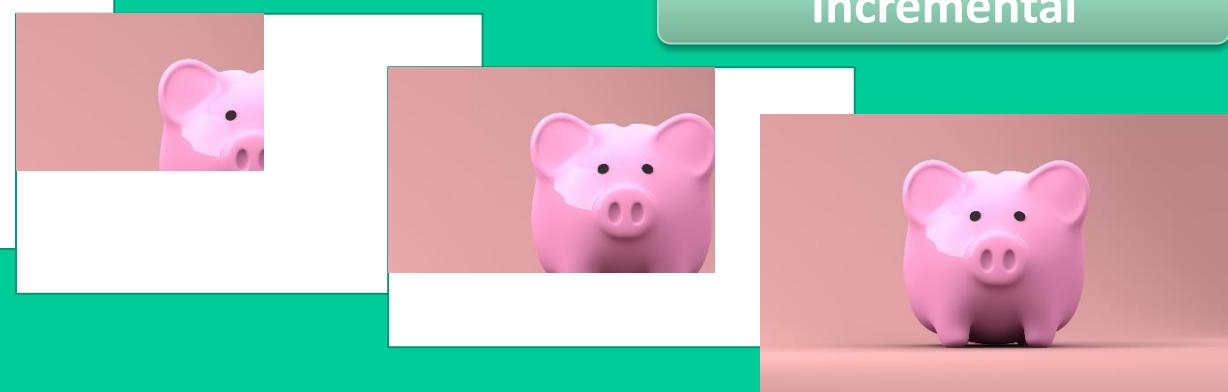
2. Adaptive life cycle

When requirements are not known in advance

Iterative



Incremental



Iterative Life Cycle

2. Iterative life cycle, the project scope is generally determined early in the project life cycle, but time and cost estimates are routinely modified as the project team's understanding of the product increases. Iterations develop the product through a series of repeated cycles, while increments successively add to the functionality of the product.

Incremental Life Cycle

3. In an **incremental life cycle**, the deliverable is produced through a series of iterations that successively add functionality within a predetermined time frame.



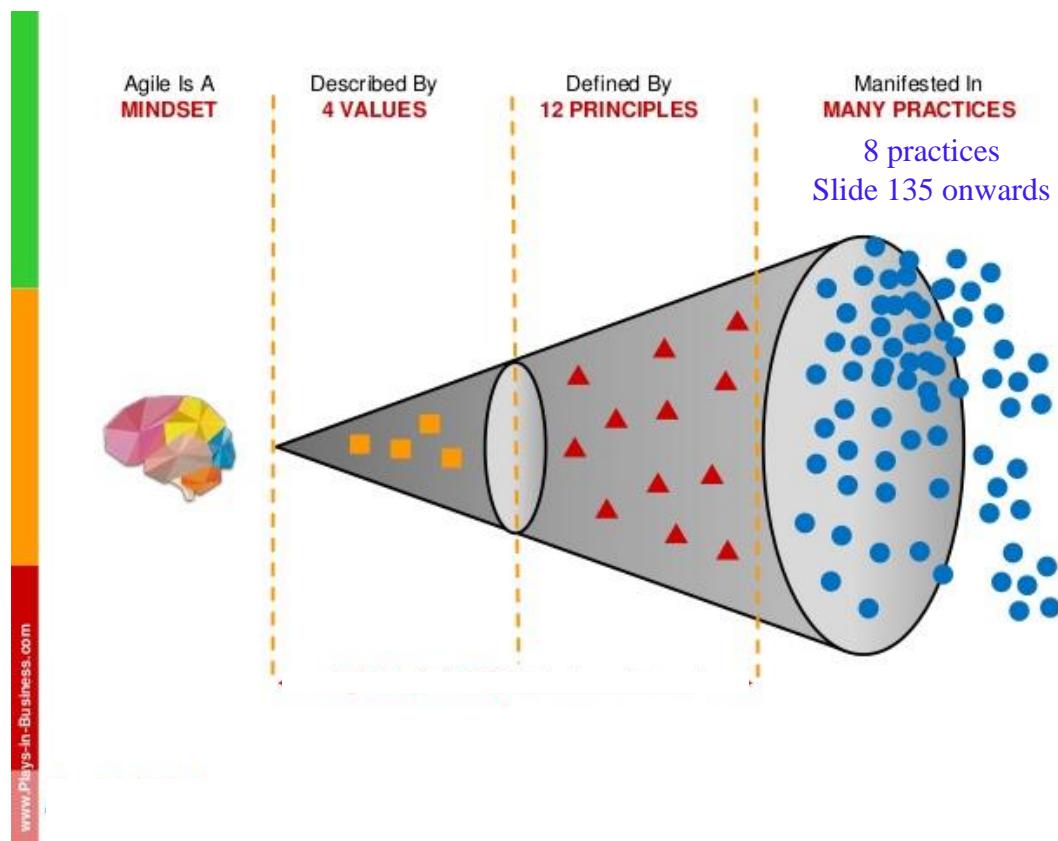
Concept: Iterative-Incremental

3. Hybrid life cycle

3. A hybrid life cycle is a combination of a predictive and an adaptive life cycle. Those elements of the project that are well known or have fixed requirements follow a predictive development life cycle, and those elements that are still evolving follow an adaptive development life cycle.



Agile is a Mindset





THE
AGILE
Manifesto

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

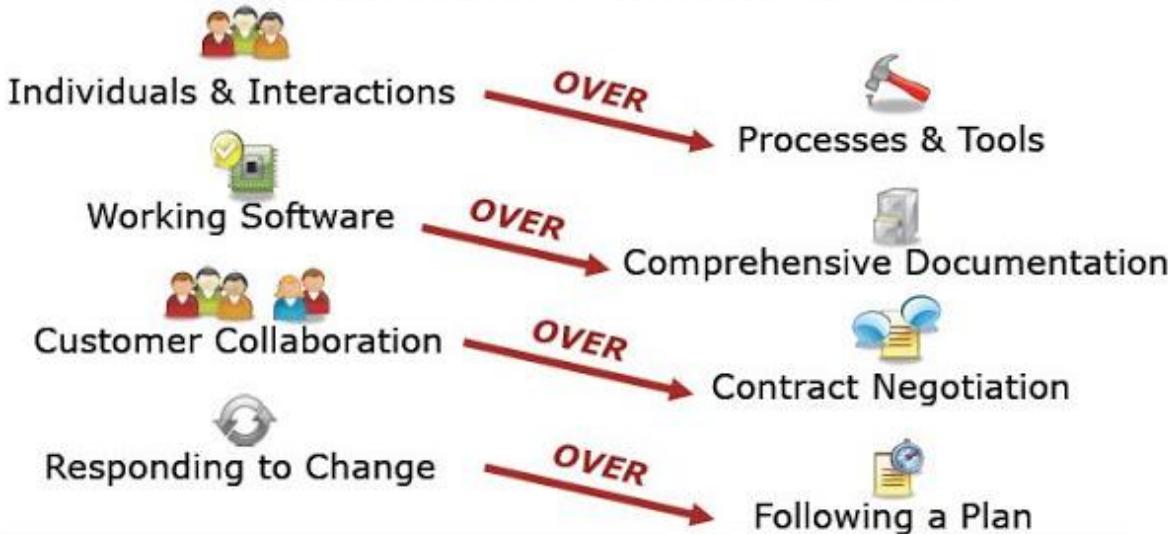
Responding to change over following a plan

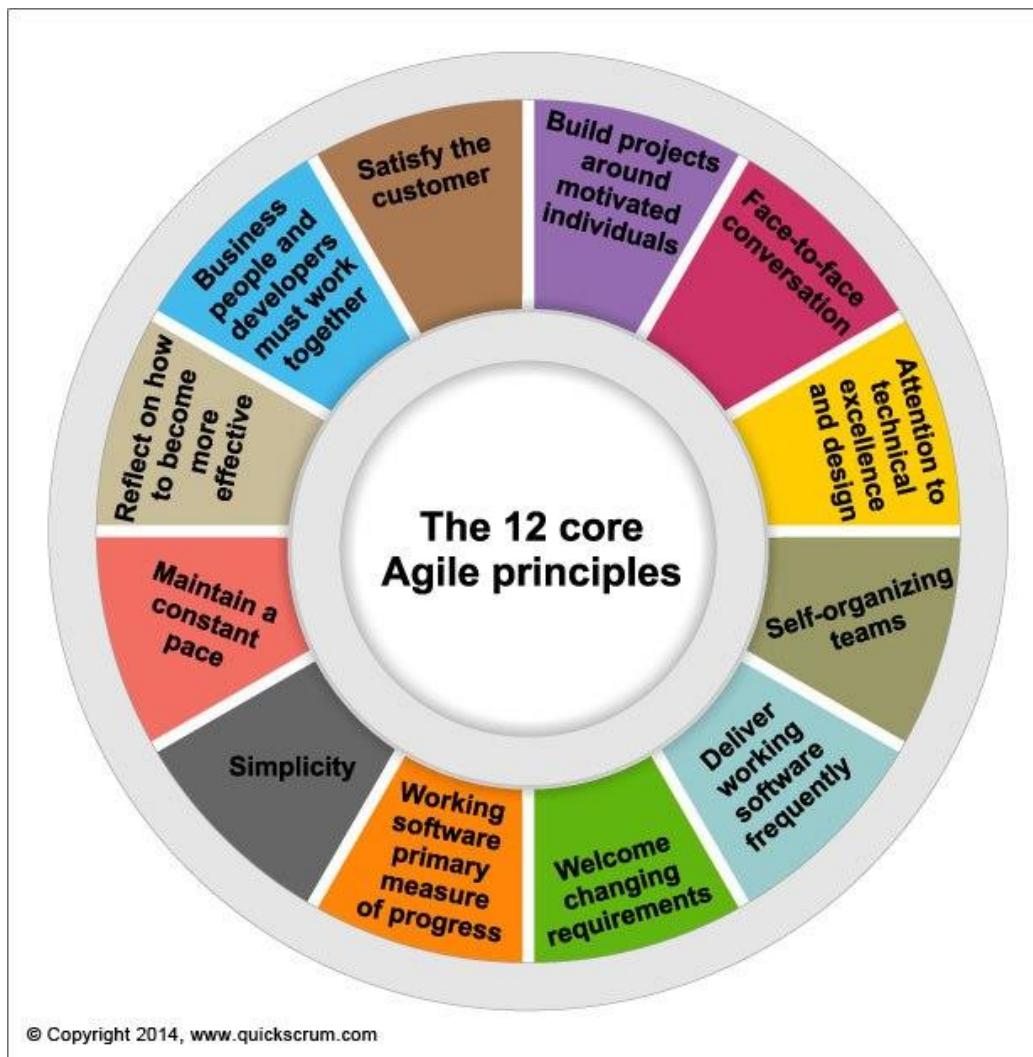
That is, while there is value in the items on the right, we value the items on the left more

The Agile Manifesto*... we know it well

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

Through this work we have come to value:





Agile Principles

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 to the shorter timescale.
4. Business people and developers must work together daily throughout the project.

Agile Principles

5. Build project around motivated individuals. Give them the environment and the support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
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.

Agile Principles

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.

QUIZ
1

Agile Manifesto states, "Responding to change _____ ?"

- a. over contract negotiation
- b. over following a plan
- c. over dictating a solution
- d. over comprehensive documentation

QUIZ
2

What does the Agile Manifesto principle “Build projects around motivated individuals” mean?

- a. Only engage in projects that motivate individuals
- b. The best architectures emerge from self-organizing teams
- c. Trust the team to get the job done
- d. Do projects that will ‘excite’ the customers

QUIZ
3

Which of the following is NOT seen as a benefit of Agile?

- a. Emphasis on collaboration and frequent demonstrations of progress
- b. Accurate predictions of project completion
- c. Lightweight, relying on whiteboards, index cards, and facilitation techniques
- d. Implicitly focusing on 'push vs. pull'

QUIZ
4

Agile does NOT recommend Big Design upfront, because:

- a. Too much emphasis on design limits the amount of requirements gathering
- b. The best designs usually emerge from writing a clean code over a period of time
- c. Agile does not recommend design as it constrains development velocity
- d. Designing up-front wastes time that can be better spent on other activities

QUIZ 5

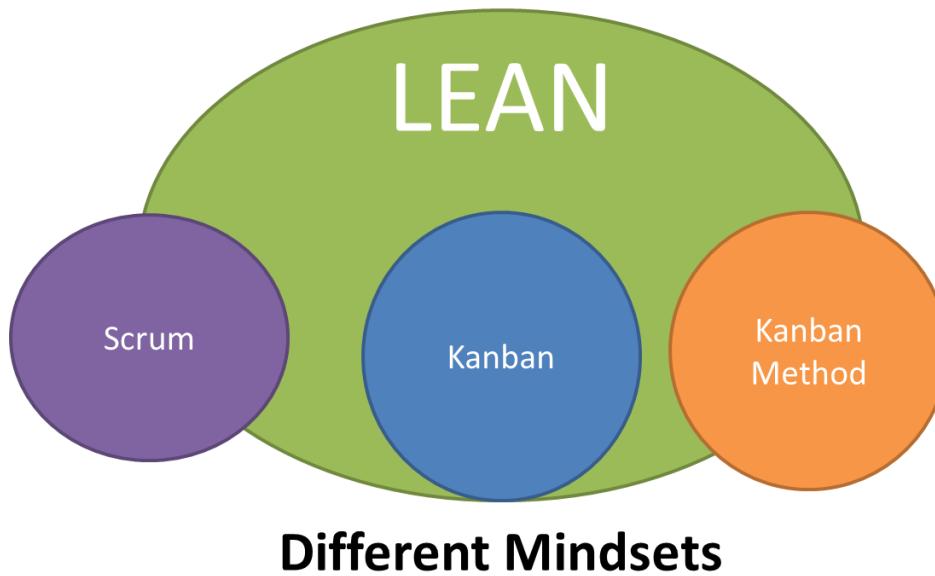
Which of the following Agile Manifesto values is incorrect?

- a. Working software over comprehensive documentation
- b. Responding to plans over following a plan
- c. Individuals and interactions over plans and practices
- d. Customer collaboration over contract negotiation

Summary

- Agile evolved in the late 1990s, in response to the burdens of heavy documentation and frequent requirements change.
- The Agile Manifesto was signed in 2001, by 17 software developers, which captured the values and principles of the Agile movement.
- Agile Manifesto highlights 12 principles.
- As a lightweight project management approach, Agile avoids big design, heavy documentation, and top-down management or control.

Lean thinking is a business methodology that aims to provide a new way to think about how to organize human activities to deliver more benefits to society and value to individuals while eliminating waste.



Agile approaches and *agile methods* are umbrella terms that cover a variety of frameworks and methods.

Agile is a Blanket Term for Many Approaches

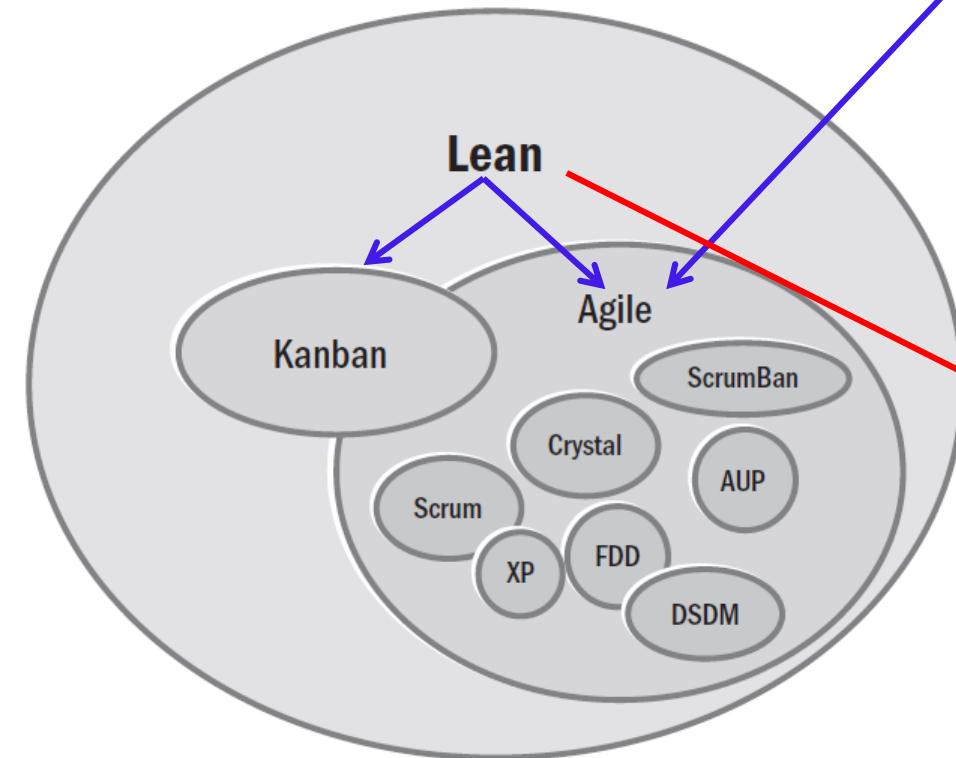
Is agile an approach, a method, a practice, a technique, or a framework?

Any or all of these terms could apply depending on the situation.

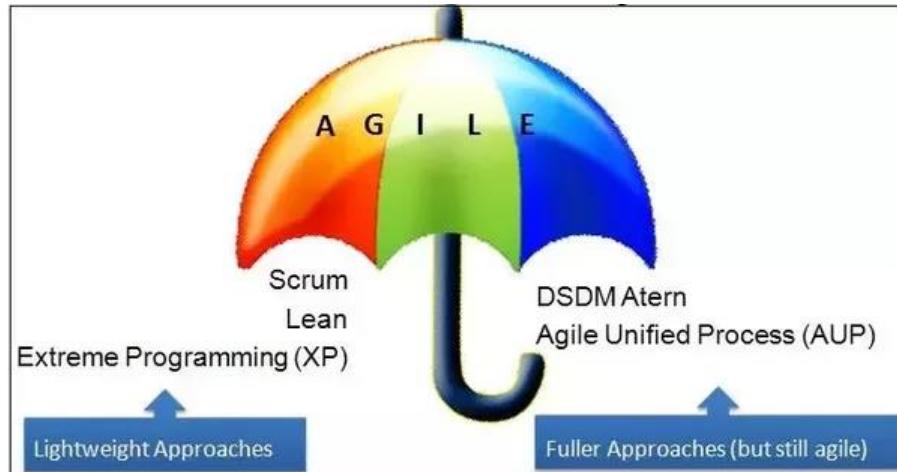
This practice guide, uses the term “approach” unless one of the other terms is obviously more correct

Lean traditionally focuses on the elimination of the eight kinds of waste/*muda*:

1. defects,
2. over-production,
3. waiting,
4. non-utilized talent,
5. transportation,
6. inventory,
7. motion,
8. extra-processing.



Agile approaches and *agile methods* are umbrella terms that cover a variety of frameworks and methods. **Agile is a Blanket Term for Many Approaches**



Is agile an approach, a method, a practice, a technique, or a framework?

Any or all of these terms could apply depending on the situation.

This practice guide, uses the term “approach” unless one of the other terms is obviously more correct

A number of core Agile methodologies share the same philosophy expressed in the Agile Manifesto. However, they have different practices, processes, and techniques.

Some of the Key Agile Methodologies are:

Scrum

Extreme Programming
(XP)

Crystal

Dynamic Systems
Development Method
(DSDM Atern)

Feature Driven
Development (FDD)

Agile Project
Management (APM)

Lean Kanban

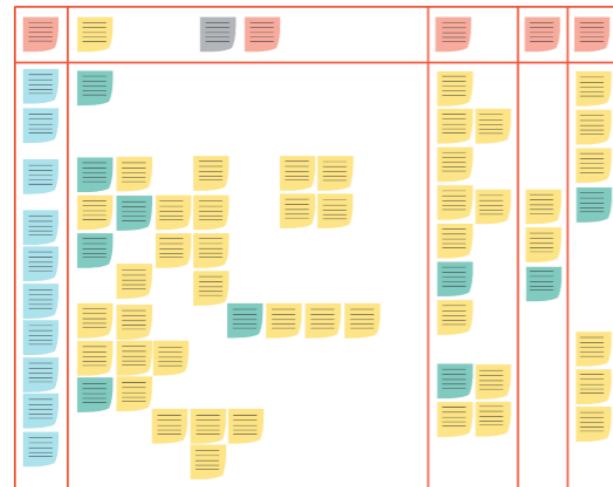
OpenUP



www.shutterstock.com · 430516177

Kanban is a concept related to Lean and Just-In-Time (JIT) production.

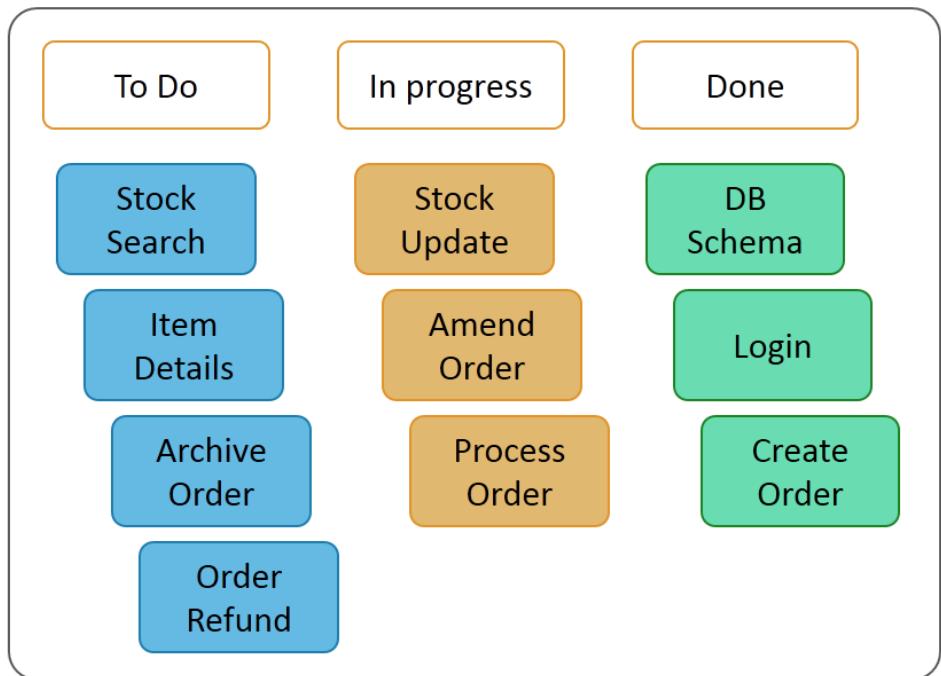
- The board is divided into segments reflecting key activities.
- The stories are represented by index cards or post-it notes.
- The status of a card is represented by its location on the board.
- A Kanban board helps the team realize how they are working and what has to be done next.



A simple Kanban board or a task board has three columns:

1. To Do
2. In Progress
3. Done

Tasks are represented by cards and status of the cards are posted under one of the three columns.



QUIZ
5

How does Kanban help a software project?

- a. By managing the throughput of a process by identifying bottlenecks
- b. By ensuring visibility of the overall software development process
- c. By providing an inexpensive mechanism to manage requirements
- d. By providing a continuous improvement process for projects

Kanban and JIT

Just-in-Time (JIT) has been very popular in the manufacturing industry, where the production system develops, delivers, or consumes only what is required at a particular point in time, instead of stacking the workflow.

- Kanban is a Pull/JIT system, wherein the flow of resources is controlled by replacing only what has been consumed.
- By visualizing and showcasing the user stories that will be addressed in the upcoming sprint through the Kanban boards, Product Owners can ensure the user stories are meeting the 'Definition of Ready' in a JIT manner.
- This ensures the team has complete understanding of the requirement and acceptance criteria on a JIT basis.

Earned Value—Planning Parameters

Budget calculations can be made using the Agile parameters as follows:

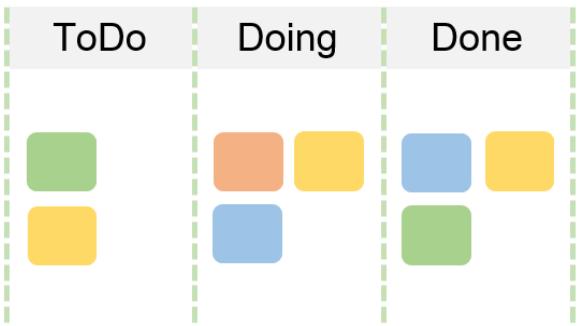
- Budget at completion can be calculated by multiplying the product backlog and the total cost to deliver each story point.

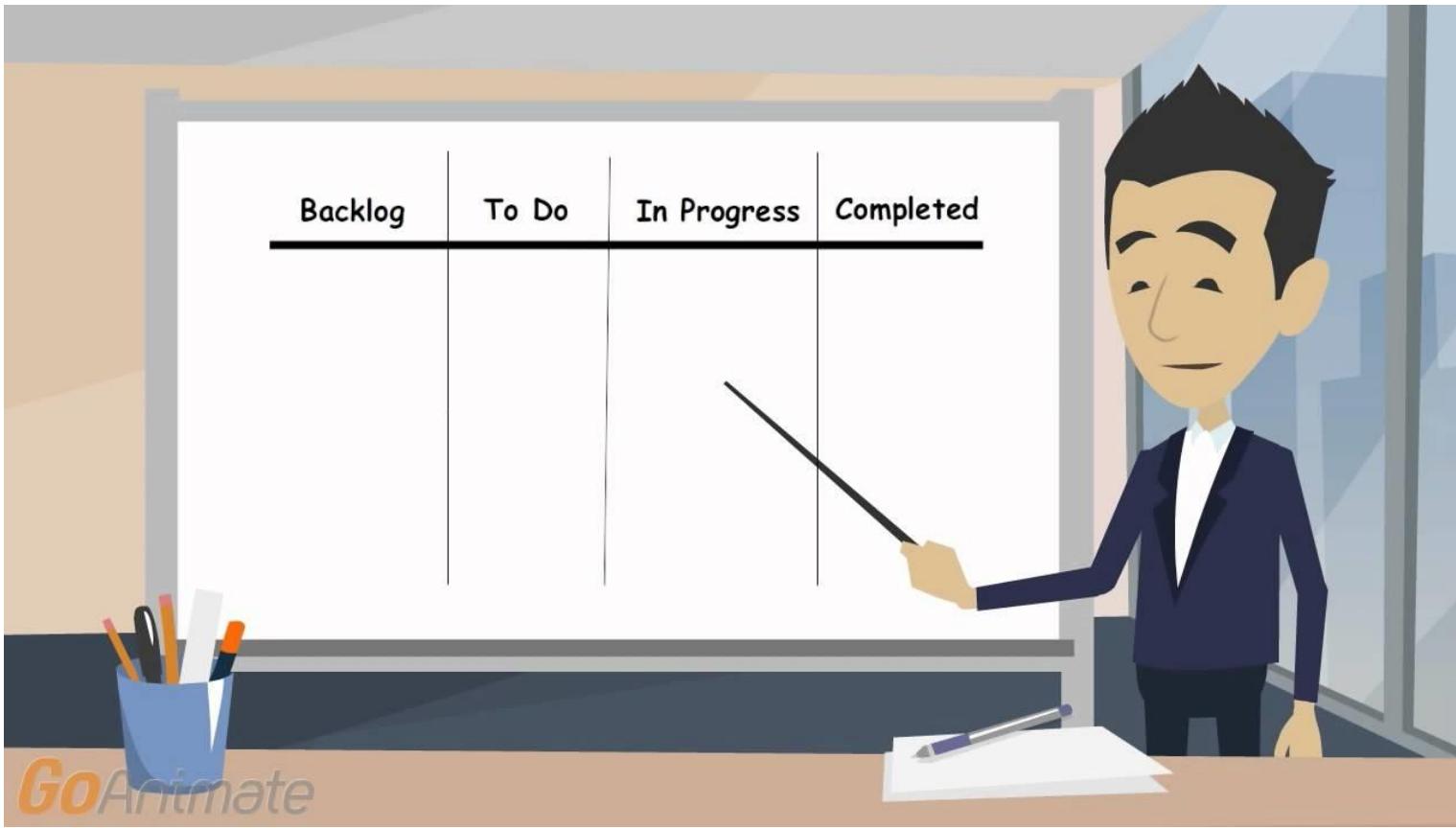
$$BAC = \text{Product Backlog} \times \text{Cost Per Point}$$

- Planned number of iterations is calculated by dividing the product backlog by the team's velocity.

$$\text{Planned Number of Iterations} = \frac{\text{Product Backlog}}{\text{Baseline Velocity}}$$

Kanban





PROJECT/TEAM: <i>Awesome Scrum Team</i>						© Liza Wood socketsandlightbulbs.com	
Backlog		To-Do		In Progress		In Review/QA	Done!
User Story 1							
User Story 2							
User Story 3							
User Story 4							
User Story 5							
User Story 6							
User Story 7							

SCRUM

 Work is done within time-boxed sprints, generally of 2-4 weeks. The goal is to produce a potentially shippable product after each sprint.

 Product is released on a particular cadence, which is determined by the sprints' length. So a team may release after 3 sprints, or every 6 weeks.

 There's a heavy focus on cross-functionality. Teams have no specified roles; everyone is a "marketer."

 Sprint kickoffs, daily standups, sprint reviews, and sprint retrospectives are vital rituals within the Scrum process.

KANBAN

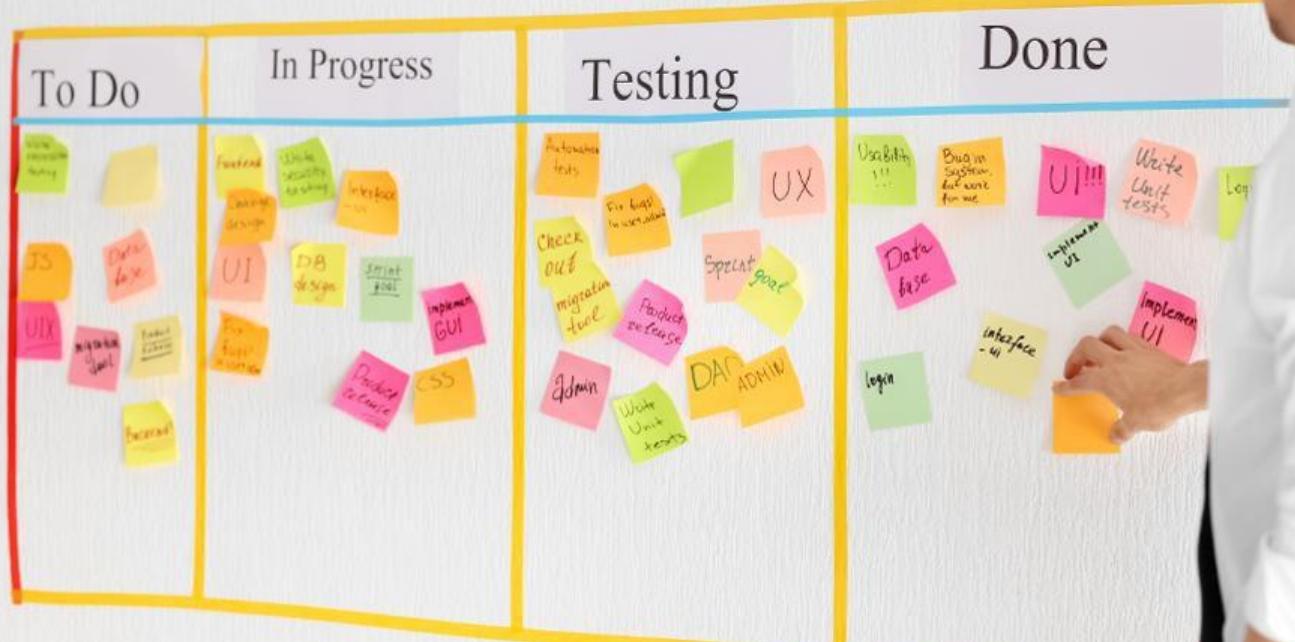
 There are no fixed-length sprints. Instead teams pull tasks from a prioritized backlog of things that need to be done.

 Releases occur continuously, or whenever there is a shippable product created.

 Team members can specialize and pull tasks related to their area of expertise, but too much specialization will reduce a team's effectiveness.

 There is an emphasis on continually improving processes, but no standardized regular meetings or rituals.

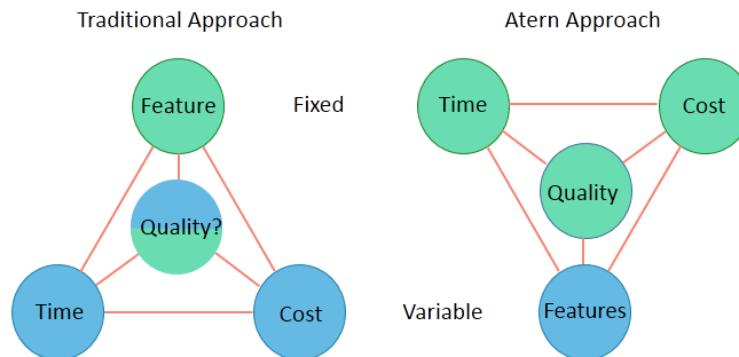
Kanban



Dynamic Systems Development Method (DSDM) was developed in the 1990s to provide more discipline to Rapid Application Development (RAD). The latest version is called Atern.

Features of DSDM are:

- It uses a prioritization technique called MoSCoW (Must, Should, Could, and Won't) to determine the requirements to be included in a release or iteration.
- Atern methodology fixes the schedule, cost, and quality.



The formulas to calculate future value and present value are:

Future Value

$$FV_N = PV(1 + i)^N$$

Present Value

$$PV = \frac{FV}{(1 + i)^N}$$

Where,

PV = Present value of a sum of money

FV = Future value of a sum of money

N = Number of years

i = Interest rate

An example for calculating future value is:

Q

If you have \$100 today and wish to invest it for three years for an interest rate of 10%, how much will you have earned by the end of the third year?

A

The formula for calculating future value is:

$$FV_N = PV(1 + i)^N$$

$$FV_N = 100 (1 + 10\%)^3 = 100 (1.331) = 133.10$$

Therefore, \$100 today is worth \$133.10 in three years with a 10% interest rate compounded annually.

An example for calculating of IRR is:

Q

Project A has an investment of \$200,000 and generates an IRR of 27%. Project B has an initial investment of \$100,000 and an IRR of 43%. Which is the best project to choose from?

A

The IRRs of Projects A and B are 27% and 43%, respectively. Project B must be selected, as it has a higher IRR than Project A.

QUIZ
2

Project A has NPV of 330, Project B has NPV of 300, and Project C has NPV of 280. Which of these projects would you select?

- a. Project A
- b. Project B
- c. Project A and Project B
- d. Project C

QUIZ
3

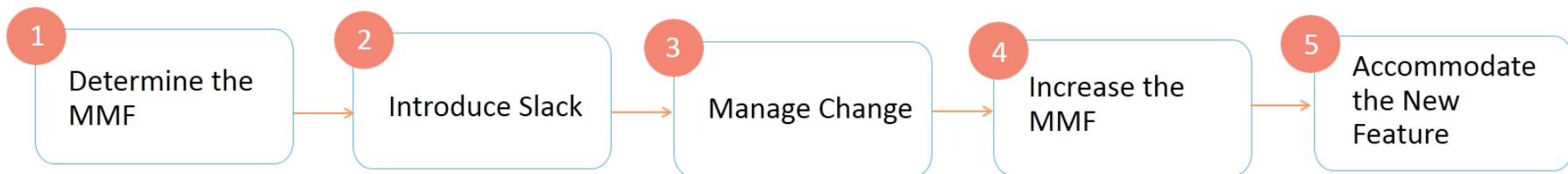
If you want to invest \$200 for 5 years at the interest rate of 20%, how much would you have at the end of the second year?

- a. \$220
- b. \$88
- c. \$112
- d. \$288

A Minimum Marketable Feature (MMF) is the smallest set of functionality that provides maximum value to the project, in line with Pareto's law (80% of benefits come from 20% of requirement.)

- By focusing on MMF, maximum value can be delivered early in the project cycle.
- MMF helps improve predictability and flexibility of a project.

The steps involved in project planning using MMF are as follows:



!

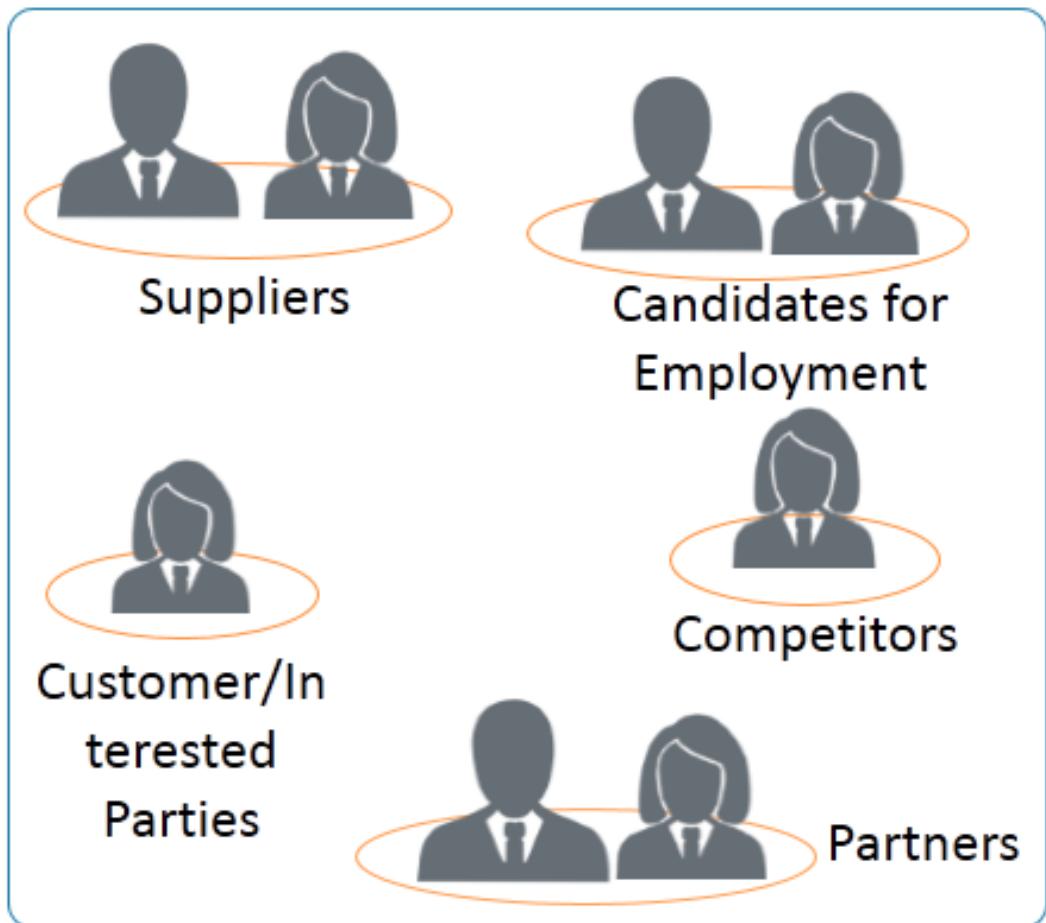
Incrementally delivering the features, which are part of MMF, provide maximum value to stakeholders.

Stakeholder Management

A stakeholder is anyone:

- Who has a stake in a project,
- Whose interest is positively or negatively impacted by the project's outcomes, or
- Who can impact the project positively or negatively.

Effective management of stakeholders is one of the most important determinants of project success, and it can be achieved through effective and timely communications.



Project Charter

Project Charter is a key reference document listing the stakeholders actively involved in the project. Although the purpose of a Charter is to provide an overview of the project in terms of goals, costs, budget, deadlines, deliverables, communication channel and frequency, sponsor, and so on, it also provides a ready reckoner of the stakeholders who are interested in the project outcome.

An Agile Charter is typically documented on a whiteboard. A chartering session helps a team:

- understand the parameters of team work and its context within the project;
- make well-informed decisions;
- identify the value the project will deliver to the business; and
- develop the trust and confidence needed in the project.

Understanding Stakeholder Needs

Agile methods acknowledge the semantic gaps that always exist between the development team who convert customer requirements into IT solutions, and the needs of the end customer. Many artifacts are used in Agile projects to address such gaps. They help to facilitate discussion, identify the best possible solution, and ensure constant collaboration.

Artifacts used to ensure knowledge sharing at the early stages of a project are:

- Wireframes
- Personas

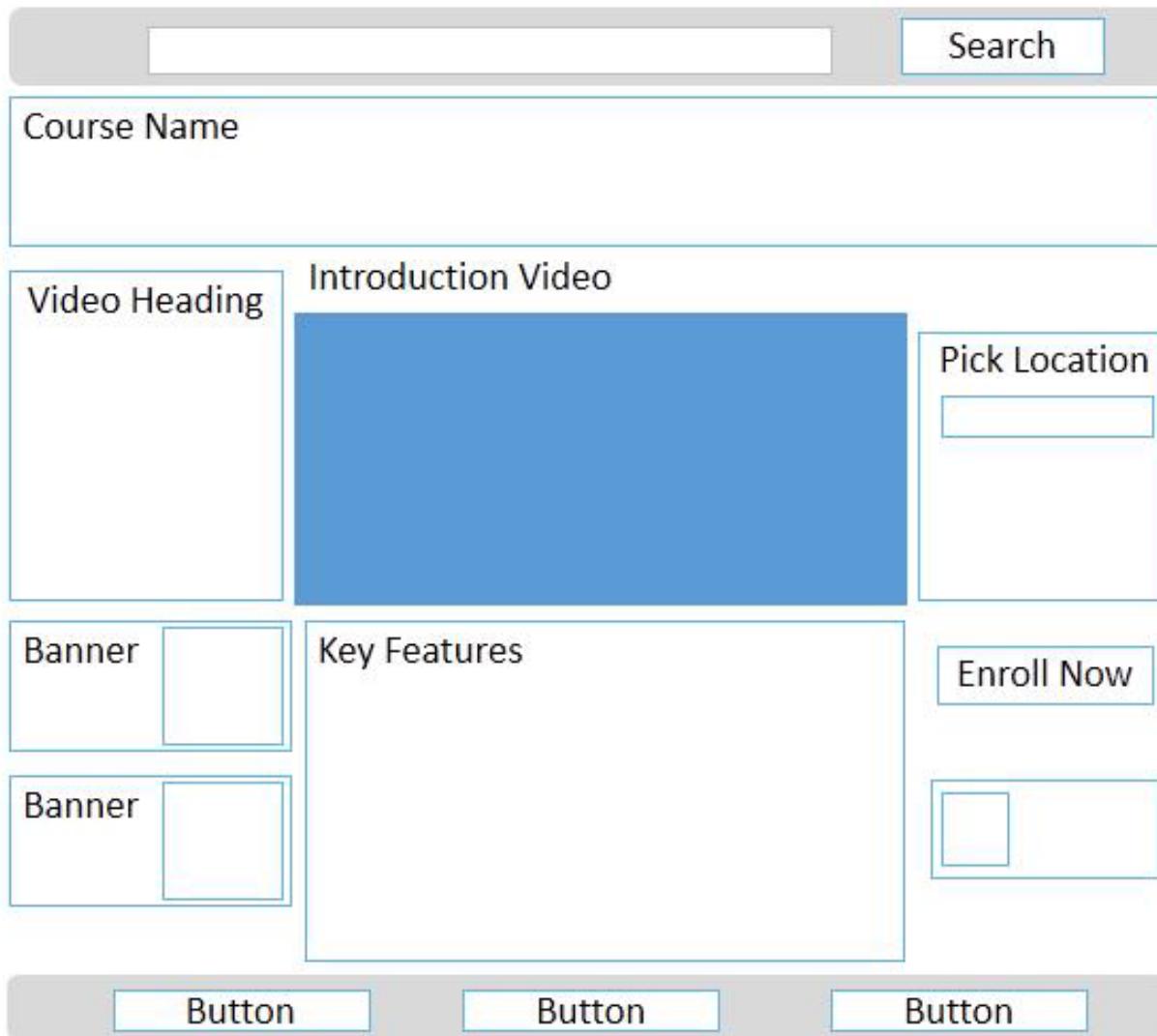
A wireframe:

- is a low fidelity, non-graphical prototype/artifact.
- should be simple with no colors, graphics, or other layout features.
- shows the skeleton of a screen, representing its structure, and basic layout.
- contains and localizes contents, features, navigation tools, and interactions available to the user.

Agile wireframes are:

- black and white, and are accompanied by some annotations to describe the behavior of the elements, their relationships, and their importance.
- often put in context within a storyboard and are refined frequently.
- used as a communication tool that serves as an element of conversation and confirmation of 'Agile' user stories.

Website with Wireframe Design



Agile Personas

The Agile Persona is a central element of Alan Cooper's interaction design. A persona is an imaginary representation of a typical user, which can be extended to user roles. Avoid picking personas who are real users. Add details to each persona:

- Likes and dislikes
- When, where, why (using the product)
- Model and make of car
- Job
- Goals



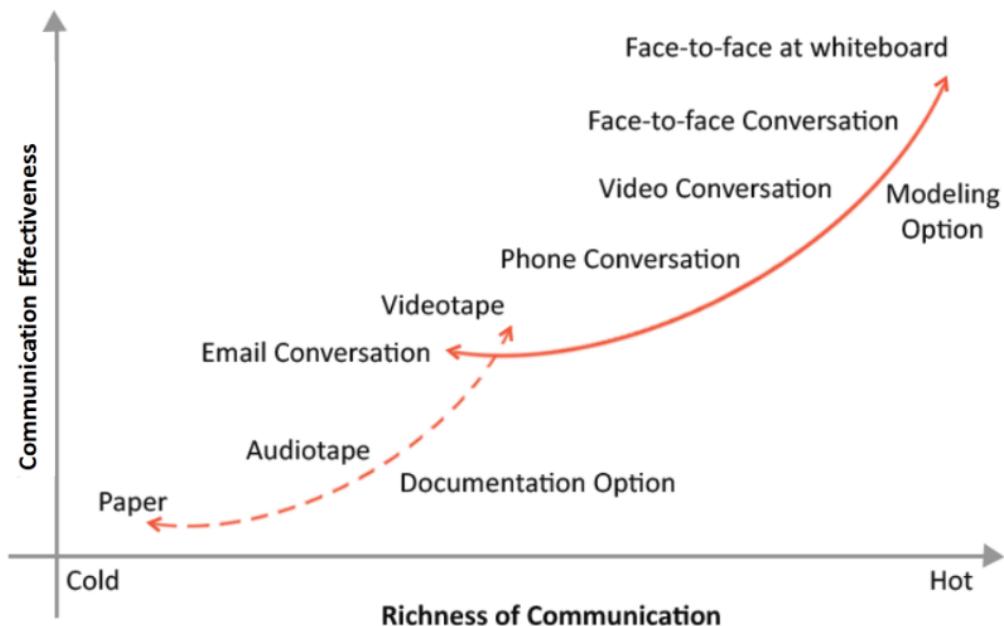
Example of an Agile Persona:

Jason is a software developer who lives in Manhattan. He has an apartment in Soho and enjoys walking his Chihuahua around the neighborhood chatting with people he runs into. He uses an Apple Air for his programming and often spends his day working in the Urban Grind coffee shop enjoying their fresh ground cappuccinos. Jason is passionate about Agile techniques and is an active member of the local Agile Leadership Network.

Communication Management

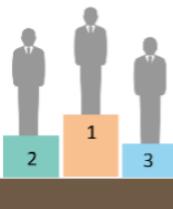
Every project manager learns that the most valuable factors for project success is communication.

Agile emphasizes that the most effective communication occurs face-to-face, which promotes trust and a two-way communication.



Agile Negotiation and Conflict Management

Kenneth Thomas and Ralph Kilmann identified five conflict modes:



Competing

High assertiveness and low cooperativeness. The goal is to “win.”



Avoiding

Low assertiveness and low cooperativeness. The goal is to “delay.”



Compromising

Moderate assertiveness and cooperativeness. The goal is to “find a middle ground.”



Collaborating

High assertiveness and high cooperativeness. The goal is to “find a win-win solution.”

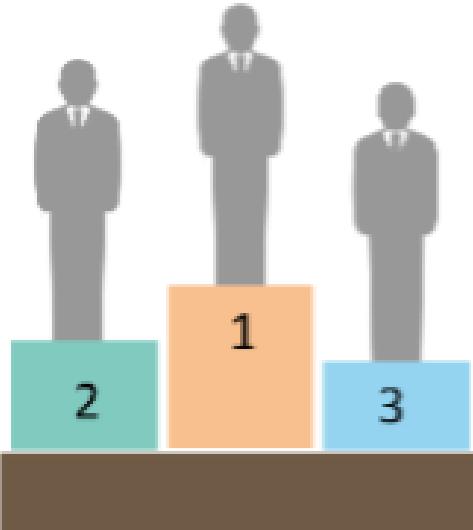


Accommodating

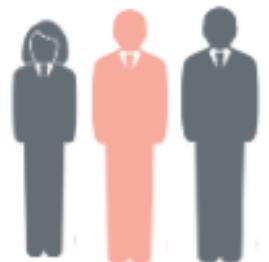
Low assertiveness and high cooperativeness. The goal is to “yield.”



Agile focuses on the ‘Collaborating Mode’ as the preferred conflict mode instrument. Any negotiation in Agile projects should always result in a Win-Win Scenario.



Competing
High assertiveness and
low cooperativeness.
The goal is to “win.”

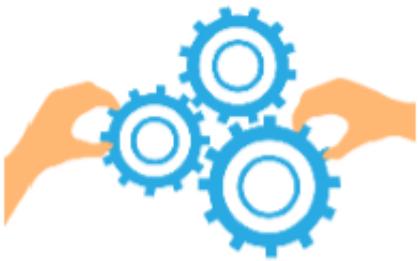


Avoiding
Low assertiveness and
low cooperativeness.
The goal is to “delay.”



Compromising

Moderate assertiveness and cooperativeness. The goal is to “find a middle ground.”



Accommodating

Low assertiveness and high cooperativeness. The goal is to “yield.”

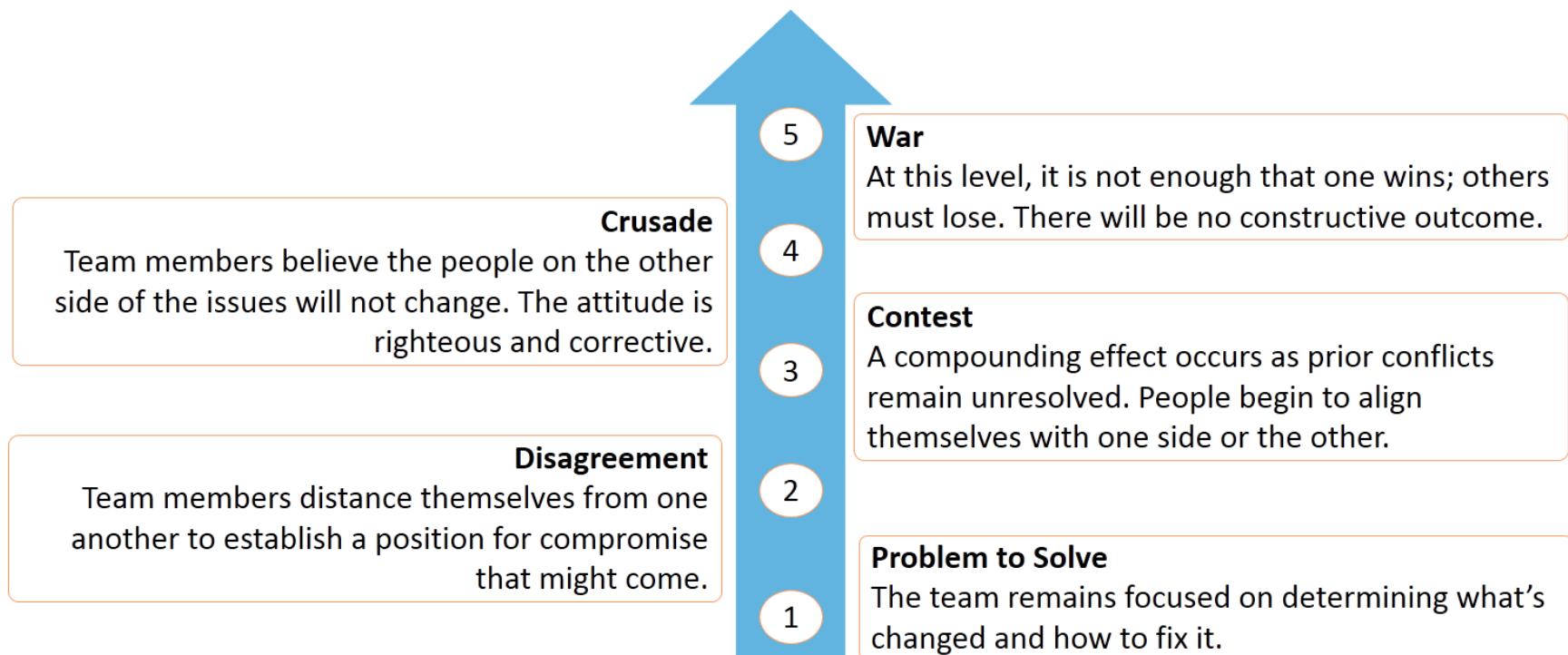


Collaborating

High assertiveness and high cooperativeness. The goal is to “find a win-win solution.”

Five Levels of Conflict

Conflicts are inevitable and even desirable in any team. Five levels of conflict are identified by Speed Leas depending on the intensity of the conflict:



1

Problem to Solve

The team remains focused on determining what's changed and how to fix it.

2

Disagreement

Team members distance themselves from one another to establish a position for compromise that might come.

3

Contest

A compounding effect occurs as prior conflicts remain unresolved. People begin to align themselves with one side or the other.

Crusade

Team members believe the people on the other side of the issues will not change. The attitude is righteous and corrective.

4

5

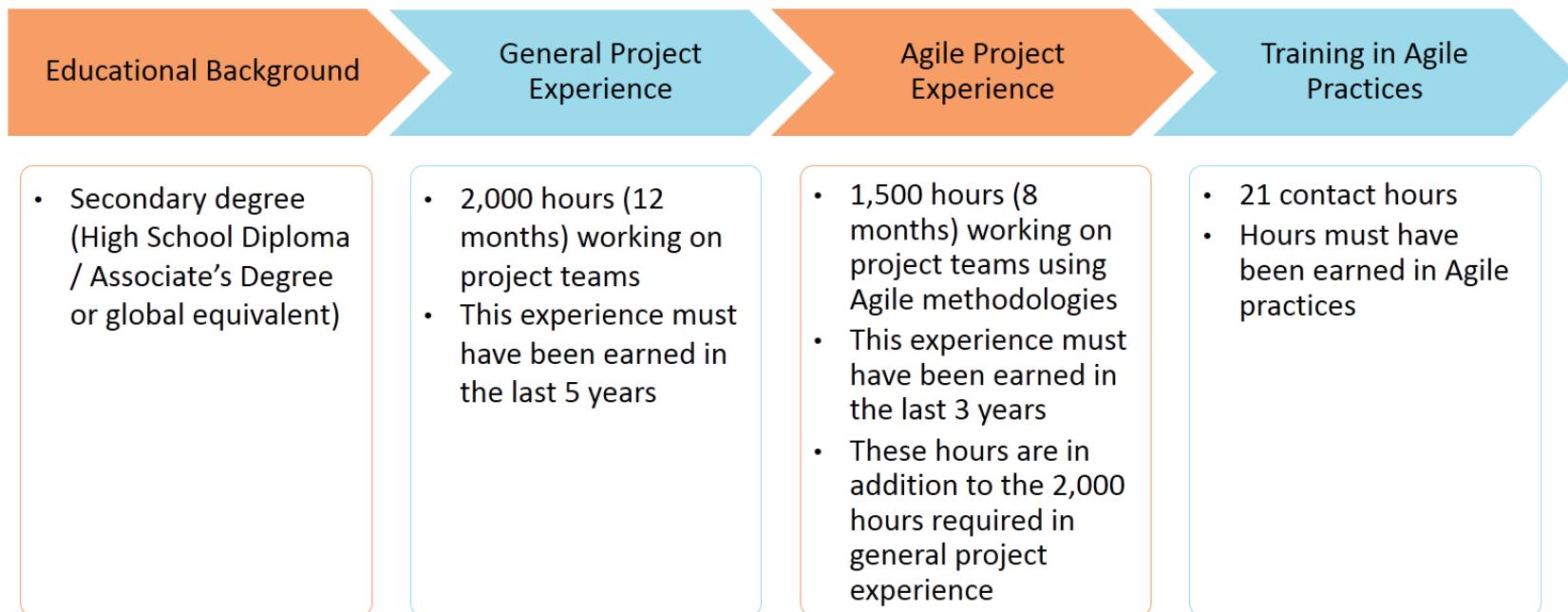
War

At this level, it is not enough that one wins; others must lose. There will be no constructive outcome.

4

PMI-ACP Eligibility Criteria

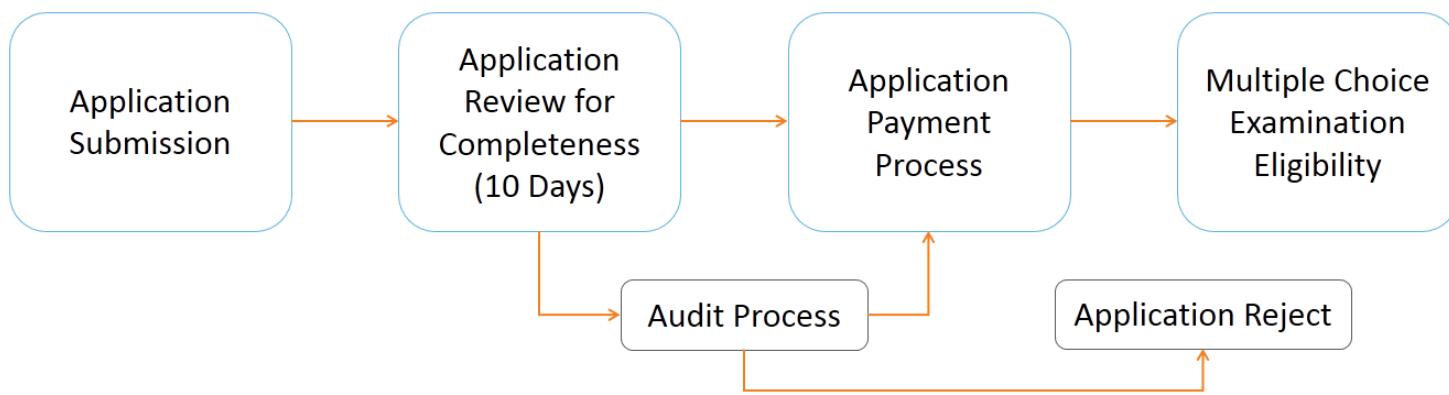
The eligibility criteria for the ACP certification can be summarized as follows.



PMI-ACP Application Timeline

Application for the PMI-ACP exam can be submitted online. There will be 90 days to complete the application.

The process is as follows:



Fee Structure

The fee structure is as follows:

Exam Type	PMI Member Status	Fees (US Dollars)
Computer-Based Testing (CBT)	Yes	435
Computer-Based Testing (CBT)	No	495
Paper-Based Testing (PBT)	Yes	385
Paper-Based Testing (PBT)	No	445
Reexamination CBT	Yes	335
Reexamination CBT	No	395
Reexamination PBT	Yes	285
Reexamination PBT	No	345

Renewal

The renewal process comprises the following:

- Certification needs a renewal after 3 years.
- Within the 3 years validity, 30 Professional Development Units (PDUs) must be earned.
- If the renewal is skipped, the certification is suspended for one year.
- If it is not renewed in this time period, the certification will expire and certified status cannot be claimed.

Exam Type	PMI Member Status	Fees (US Dollars)
Continuing Certification Requirements (CCR) Certification Renewal	Yes	60
Continuing Certification Requirements (CCR) Certification Renewal	No	150

Exam Questions

The following table provides important information about PMI-ACP exam:

Exam Component	Specifications
Allotted exam time	3 hours
Total number of questions	120
Total number of scored questions	100
Total number of pre-test questions	20

!

It is not possible to identify the experimental questions out of these 120 questions. There is no penalty for wrong answers.

Exam Questions Distribution

The allocation of questions in the PMI-ACP examination is as follows:

Domain	Percentage in the Examination
Domain I: Agile Principles and Mindset	16%
Domain II: Value-Driven Delivery	20%
Domain III: Stakeholder Engagement	17%
Domain IV: Team Performance	16%
Domain V: Adaptive Planning	12%
Domain VI: Problem Detection and Resolution	10%
Domain VII: Continuous Improvement (Product, Process, People)	9%

Practice Exam Questions

Some key points about questions in this course:

- Quiz questions are provided to check the understanding of the concept.
- Practice test papers are provided:
 - These indicate how the questions could be presented in the exam.
 - They should not be used as a source of knowledge.
 - These questions will help the candidate revisit the topics covered in the lesson.

Reference Books

PMI-ACP recommended references:

- PMI-ACP Certification Handbook:
http://www.pmi.org/Certification/~/media/PDF/Certifications/PMI-ACP_Handbook.ashx
- Agile Retrospectives: Making Good Teams Great: *Esther Derby, Diana Larsen, Ken Schwaber*: ISBN #0977616649
- Agile Software Development: The Cooperative Game – 2nd Edition: *Alistair Cockburn*: ISBN #0321482751
- The Software Project Manager's Bridge to Agility: *Michele Sliger, Stacia Broderick*: ISBN #0321502752
- Coaching Agile Teams: *Lyssa Adkins*: ISBN #0321637704
- Agile Project Management: Creating Innovative Products – 2nd Edition: *Jim Highsmith*: ISBN #0321658396
- Becoming Agile: ...in an imperfect world: *Greg Smith, Ahmed Sidky*: ISBN #1933988258
- Agile Estimating and Planning: *Mike Cohn*: ISBN #0131479415
- The Art of Agile Development: *James Shore*: ISBN #0596527675
- User Stories Applied: For Agile Software Development: *Mike Cohn*: ISBN #0321205685
- Agile Project Management with Scrum: *Ken Schwaber*: ISBN #073561993X
- Lean-Agile Software Development: Achieving Enterprise Agility: *Alan Shalloway, Guy Beaver, James R. Trott*: ISBN #0321532899