Agile Foundations

THE FOLLOWING PMI-ACP® EXAM TOPICS ARE COVERED IN THIS CHAPTER:

Agile Principles and Mindset

- Task 1: Advocate for agile principles by modeling those principles and discussing agile values in order to develop a shared mindset across the team as well as between the customer and the team.
- Task 2: Help ensure that everyone has a common understanding of the values and principles of agile and a common knowledge around the agile practices and terminology being used in order to work effectively.
- Task 3: Support change at the system or organization level by educating the organization and influencing processes, behaviors, and people in order to make the organization more effective and efficient.
- Task 4: Practice visualization by maintaining highly visible information radiators showing real progress and real team performance in order to enhance transparency and trust.
- Task 5: Contribute to a safe and trustful team environment by allowing everyone to experiment and make mistakes so that each can learn and continuously improve the way he or she works.
- Task 6: Enhance creativity by experimenting with new techniques and process ideas in order to discover more efficient and effective ways of working.
- Task 7: Encourage team members to share knowledge by collaborating and working together in order to lower risks around knowledge silos and reduce bottlenecks.
- Task 8: Encourage emergent leadership within the team by establishing a safe and respectful environment in which new approaches can be tried in order to make improvements and foster self-organization and empowerment.
- Task 9: Practice servant leadership by supporting and encouraging others in their endeavors so that they can perform at their highest level and continue to improve.
In this chapter, we will go through some of the information found in *Agile Principles and Mindset* of the PMI-ACP exam as an overview of how Agile began. We will continue in Chapter 2 and Chapter 3 as we get into greater depth on frameworks and the tasks of Agile.

**History and the Agile Manifesto**

When most people think of Agile, they see it as a recent development of a new kind of methodology. In fact, in the age of technology and software development, there are a lot of buzzwords out there revolving around continuous delivery and software production. The history of Agile spans a decade or more. As early as the 1990s, it became necessary for organizations to keep up with the rapid pace of enterprise software development. Technology went from answering machines to dial-up modems and “You’ve got mail” to the types of technology that we’re using today. Due to the proliferation of software programs, apps, and other cutting-edge technologies, it became necessary for organizations to find a better way to manage and adapt.

Most organizations had trouble keeping up with changes in requirements, computer systems, and software, and many were using outdated modalities and best practices. In some industries, it took much longer than expected to create the technology necessary to run the organization or to get certain projects off the ground. The industries that were affected were not necessarily companies that produce software or computer technologies; rather, it was the companies that were using those technologies to get their projects off the ground that felt the greatest impact. Industries like government, telecommunications, automotive, and others that were dependent on software and processing technologies being totally up-to-date were finding that the technologies that they were using were outdated and not effective for the projects they were working on currently.

Organizations were also figuring out that heavier project management methodologies, which focused more on long-term planning, were not as effective for the types of projects they were working on now. Due to a highly changing environment and constant demands to stay current on innovative technologies, it was imperative to find newer and better ways of doing things.

**The Agile Alliance**

The frustration of heavy methodologies that didn’t work for the industry and attempting to find a more “lightweight” model of project management led 17 software developers to meet and discuss new methods or ways to embrace changes and to provide enhanced on-time feedback.
When most people hear about these software developers getting together, they think about the well-known Agile Alliance, which created the Agile Manifesto. While it wasn’t the first time this group got together to discuss a variety of methodologies and best practices, the most famous meeting was in Snowbird, Utah, in February 2001.

The goal of this meeting was to discuss ways to simplify or create a lightweight type of practice or practices that could fluctuate depending upon the project’s needs. The ability to build working software quickly by understanding what the customer needs with very little front-end planning and documentation formed a large part of the discussions. Some of the more recognizable names that make up the Agile Alliance are Kent Beck and Ward Cunningham, who created the eXtreme Programming methodology, or XP, as well as Jeff Sutherland and Ken Schwaber, who created the Scrum process in the early 1990s.

The Agile Alliance determined that new methods should be based on iterative and incremental development rather than a preplanned and well-defined scope of work right at the very beginning of the project. This would allow the result to surface organically as new features or requirements were discovered.

The second driving factor was creating higher quality software in shorter time frames, running short sprints or iterations and work to produce something usable at the end of each. This would allow the user to test the increment and make determinations for new requirements for the next iteration. To run quick iterations and create usable increments, it was difficult to preplan everything. Thus, the Agile Alliance determined that it was necessary to have requirements and solutions evolve through collaboration and self-organizing, cross-functional teams. This would allow for business value to evolve and develop a cross-pollinated understanding of what the results should be at the team level. Everyone knows the vision, even when the vision changes.

### The Agile Manifesto

For those variables to work effectively, it was necessary to adapt current methods, including a Waterfall type framework to suit the needs of more technical types of projects. Based on the discussions and the need to adapt the common Waterfall types of methodologies to suit a rapid, frantic pace in technology, the Agile Manifesto was born.

The Agile Manifesto was designed to be a set of lightweight and guiding principles rather than set rules and formal processes. The goal of a written manifesto is for a person or persons to publicly announce something they feel strongly about or to make their views known. That doesn’t mean a long statement spanning pages and pages. In fact, much like Agile methodologies in general, the manifesto is short and easy to understand and gets straight to the point without any additional noise.

The Agile Manifesto forms the basis for most methods currently in use today, including Scrum, eXtreme Programming, Lean, Crystal Methods, and others.

The Agile Manifesto has very specific values that are stated very simply without any additional explanation. The reason for this is so that each project could develop their best practices around these very simple considerations.

**Individuals and Interactions over Processes and Tools** The first value puts people first over the staunch best practices found in heavier methodologies. Without interactions and
collaboration, the processes and tools don’t work. Instead, they hinder a project’s ability to be successful in the tech space specifically. This isn’t a suggestion that process and tools are unnecessary; rather, it’s more that there is better value found in individuals and interactions.

**Working Software over Comprehensive Documentation** Excessive documentation is seen as wasteful if the software doesn’t work. Too many methodologies are focused on up-front planning, setting hard due dates and baselines, and continual updates as things change. Many Agile practices focus on documentation at the “last responsible moment.” It’s more important to have software that works and that meets business value and tech specs than it is to spend time putting together massive plans that will change.

**Customer Collaboration over Contract Negotiation** As we go forward, you’ll come to see that procurement is more flexible on Agile types of projects because it has to be. Having requirements that are flexible doesn’t mean that external sellers or support staff via contractual relations aren’t necessary—they are. However, collaboration with the customer and working toward the right solution is more important than locking down a contract that doesn’t meet the needs of the customer in the end. Breach of contract is no joke, and lack of flexibility in procurement counteracts flexibility of requirements and customer needs.

**Responding to Change over Following a Plan** This goes back to comprehensive documentation being a waste of time if you know that it is going to change anyway. Anyone in the project management space knows that putting together a well-thought-out plan and then finding out that things have changed is frustrating. It’s a lot like planning and saving to buy what you thought was the latest, greatest piece of technology only to find out that your neighbor just bought the next version of this technology and it’s way cooler than yours. Not fun! All kidding aside, preplanning something that you suspect may look and act totally different in the end won’t work. The ability to pivot and be agile is the crux of all methods and frameworks. Change happens—it’s expected and embraced.

A lot of people look at the Agile Manifesto and think that individuals and interactions are more important than processes and tools, or that customer collaboration takes the place of contract negotiation. That is not the case at all. The consensus was that heavier processes and an overuse of tools and documentation were not working in cutting-edge industries, and that current methods and frameworks were too heavily skewed toward items that dragged projects down rather than on areas that could enhance the overall project and customer needs.

Too many projects were focused on the project manager and team spending an exorbitant amount of time doing comprehensive documentation, and this was taking away from interacting with individuals on the team and creating working software. Add to that limited collaboration with the customer and not being able to quickly respond to changes in the scope of work.

The Agile Manifesto does not suggest that we do one over the other; rather, it advocates that while all of the mentioned items are important, some are valued more than the others. If there is a need for comprehensive documentation due to a customer or organizational request, that’s fine if there is also working software. Working software, though, is valued more than excessive or comprehensive documentation that may change soon after you have
completed it. Any methods that fall under the Agile umbrella are based on iterative and incremental development by producing higher quality software in shorter time frames. As you will see, requirements and solutions can evolve with a team that is able to collaborate freely, self-organize, and be cross functional.

For the PMI-ACP exam, be very aware of the Agile Manifesto and its values because that will enable you to answer more questions correctly. If you are unsure of the correct answer, or you are deciding between two answers, always revert to the Agile Manifesto and ask yourself if the answer you are leaning toward is describing processes and tools, comprehensive documentation, contract negotiation, and following a plan; if so, then the best answer is the one that best reflects the Agile Manifesto rather than answers that are more focused on processes, tools, and documentation. It’s tempting because of your own individual experiences on the job to zero in on an answer that seems to match your experience. Make sure that you read the other answers, and use the Agile Manifesto as your guide to the correct choice.

The 12 Principles of the Agile Manifesto

The 12 major principles that were developed and placed in the Agile Manifesto are also items to bear in mind as you are reading questions on the exam. This goes further than an exam situation though, as it also addresses organizations who are trying to implement an Agile methodology and may be used to a heavier set of processes. Organizations may have to adapt their current organizational culture to embrace the 12 principles of the Agile Manifesto. Although the methodologies are lightweight and the manifesto is brief, trying to implement new methodologies is easier said than done. The Agile Manifesto and its values and principles are easy to talk about but very hard to implement.

The 12 principles are the guiding force in all methodologies that we will cover, and they are very heavily represented on the exam.

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 must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and 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.

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 and then tunes and adjusts its behavior accordingly.

The 12 Principles Simplified

Even though the 12 principles make sense as their own entities, it’s always a good idea to break them down to how they apply to keeping that Agile mindset on the exam and in your work environments. Remember that all of these principles are easier said than done in the real world, and mastering all of them takes time and dedication to the craft of Agile.

1. Customer Satisfaction: All Agile methodologies are looking for ways to bring value to the customer on a regular basis by communicating and adapting to changing customer needs.

2. Welcome Changes: This keeps the team on top of new requirements and allows for some flexibility in the design rather than preplan and go through a formal change control system every time there is an update to the scope of work.

3. Frequent Delivery: Like many methodologies and frameworks that deal with short iterations, the goal is to produce something usable early and often. Frequent delivery is about producing a usable increment in a short time span that the customer finds valuable.

4. Collocated Teams: Many best practices in Agile project management have collocated teams. If team members are located remotely or are virtual, the best practice is to collocate them for at least one iteration if possible.

5. Motivated Individuals: Teams are self-managed and self-organizing, and they are there because they want to be there. Demotivation doesn’t produce good working increments or keep the team focused on providing value.

6. Face to Face Contact: Communication is a large part of Agile project management, and face-to-face contact is the best way to communicate. It ties into collocation as well as open and honest communication across the team dynamic.

7. Working Software: Focus is on a usable increment that works, and not spending a lot of time on creating software that doesn’t work. This is done by getting frequent feedback, utilizing frequent testing and reviews, and looking back to see what changes for the better are necessary.

8. Constant/Sustainable Pace: Try to achieve a 40-hour work week and no overtime if possible.

9. Continuous Attention: The entire team looks for ways to improve quality, the design, and the overall process on a regular basis.
10. **Simplicity**: Keep it simple. Don’t add extra features that are unnecessary.

11. **Self-Organization**: The team decides for itself what it can and can’t do, and it works together on solutions.

12. **Regular Reflection**: The team keeps a constant focus on looking back to move forward more successfully.

### The Declaration of Interdependence

The Agile Manifesto gained traction in the knowledge work arena, but many believed additional input was necessary to provide principles that could be utilized in other situations outside of software development. Several of the original authors of the Agile Manifesto, as well as other experts in the field, put together the *Declaration of Interdependence* in 2005. The concept behind the Declaration of Interdependence was to develop modern management principles essential to project management and management in general. Since the Agile Manifesto was specific to software development, more clarification was needed to apply the mindset to other projects.

The six principles are based on what might be required to achieve the mindset of an Agile-type project, regardless of the industry. The Declaration of Interdependence begins with a statement that provides the philosophy of the creators and is followed by six principles.

> Agile and adaptive approaches for linking people, projects, and value. We are a community of project leaders that are highly successful at delivering results.

To achieve these results:

1. **We increase return on investment by making continuous flow of value our focus.**
2. **We deliver reliable results by engaging customers in frequent interactions and shared ownership.**
3. **We expect uncertainty and manage for it through iterations, anticipation, and adaptation.**
4. **We unleash creativity and innovation by recognizing that individuals are the ultimate source of value and creating an environment where they can make a difference.**
5. **We boost performance through group accountability for results and shared responsibility for team effectiveness.**
6. **We improve effectiveness and reliability through situationally specific strategies, processes, and practices.**

Citation: [©2005 David Anderson, Sanjiv Augustine, Christopher Avery, Alistair Cockburn, Mike Cohn, Doug DeCarlo, Donna Fitzgerald, Jim Highsmith, Ole Jepsen, Lowell Lindstrom, Todd Little, Kent McDonald, Pollyanna Pixton, Preston Smith, and Robert Wysocki.](http://pmdoi.org/)
The Declaration of Interdependence Simplified

The Declaration of Interdependence is a guide to the importance of an Agile mindset, and it is an excellent way to absorb that mindset for the exam. As we go further you will see a lot of frameworks and methodologies that build on both the Agile Manifesto and the Declaration of Interdependence and have led to the creation of many of today’s Agile best practices.

We increase return on investment by making continuous flow of value our focus. Many projects don’t reach expected return on investment (ROI) due to scope creep and lack of a clear requirements definition, which in turn bogs down the schedule and the budget and results in defect repair or added scope costs. The goal is to produce something of value by collecting requirements as current information is gathered. Requirements are collected on a continuous basis, with quick turn-around of results so that value is returned and ROI is achieved.

We deliver reliable results by engaging customers in frequent interactions and shared ownership. Many times, the customer isn’t involved in the planning or review of items produced on a project until the very end, and only then do you find out it wasn’t what they thought they wanted. The results are dependent on interacting with the customers on a regular basis to find out what is new, what has changed, and what they find valuable today. Then and only then can we produce what they want or need. The customer is part of owning the result and therefore is more invested in how they communicate and interact.

We expect uncertainty and manage for it through iterations, anticipation, and adaptation. All projects carry risk, changes, and uncertainty. We can’t control everything, but we can accept that we must be ready at any time for whatever happens. By carrying the expectation that we don’t have an if situation but instead a when situation, we can adapt and pivot as needed.

We unleash creativity and innovation by recognizing that individuals are the ultimate source of value and creating an environment where they can make a difference. Think of the many organizations that are on the cutting edge, such as Amazon, Facebook, Apple, Microsoft, Cisco, Tesla, or the many more out there who are leading the charge for creativity and innovation. Those organizations recognize the value of innovative ideas, trying out new things, and creating an environment where their people can flex their creative muscles and produce the newest ways to influence the world. It is because these organizations set it up that way that it works.

We boost performance through group accountability for results and shared responsibility for team effectiveness. This is a far cry from the hierarchical, functional organizations that were prevalent during the industrial revolution. While those organizations still exist and are necessary for some industries, the Agile world is more equal in the distribution of the work, the accountability, and the glory. It is a shift in focus to a team dynamic rather than a top-down hierarchy.

We improve effectiveness and reliability through situationally specific strategies, processes, and practices. The crux of Agile frameworks and best practices is being just that: agile. We must shift, pivot, change, and adapt as required by the situation. We can’t cram one
set of rules onto every single project and expect it to work every time. What works for one project will not work for another. Knowing this leaves a more fluid focus on solutions rather than on rules, and with that comes ROI, reliable results, managed risk, creativity, and boosted performance. Everything is interdependent on the rest.

It is typical to adopt the defined (theoretical) modeling approach when the underlying mechanisms by which a process operates are reasonably well understood. When the process is too complicated for the defined approach, the empirical approach is the appropriate choice.

Process Dynamics, Modeling and Control by Bابتunde Ogunnaike and W. Harmon Ray

**Empirical Process Control**

Ken Schwaber and Jeff Sutherland are the creators of the Scrum methodology. They adapted and used empirical process control to develop the best practices of Scrum along with the Agile Manifesto as the guiding principle. The three key aspects of *empirical process control* are also the three pillars of Scrum.

Empirical process controls focus on the following:

**Transparency:** This encompasses not just the process itself but all communications.

**Inspection:** Frequent inspection and the utilization of frequent reviews of the product service or results is essential.

**Adaptation:** None of the above will work fully without adaptation, which is the ability to embrace uncertainty and changes and manage risks accordingly.

Many of the questions on the PMI-ACP exam are based on your ability to place yourself into the mindset of the Agile team and how to function in an Agile environment. Empirical process control embraces the Agile Manifesto, but when you think of the definition of *empirical* (using your own experiences and knowledge rather than buying into a set of processes based on theory or even known logic at the time), it’s easy to see that when we find better ways of doing things, we embrace them rather than being stuck in a process or processes that simply do not work. Empiricism involves learning and adapting as needed.

---

### Real World Scenario

**There Is a Better Way...**

Most of my experience has been working with some kind of technology or another either as a consultant or running my own projects. I’ve worked with red tape and slow-moving wheels of change and I know how frustrating it is to be accountable for over-budget, behind-schedule projects due to heavy expectations of process, scope creep, or excessive documentation. Many times in my career, I would be asked to do something a “certain way,” and my response was, “Yeah but...there is a better way.” In my experience,
this wasn’t taken very well because the process was set in stone. Change is painful for organizations, and the thought that they have always done it “this way” is hard to circumvent. That is the beauty of Agile; it is based on observation and conversation. Don’t confuse that with a free-for-all though. Many of the frameworks and methods that we’ll discuss have specific timeboxing rules and best practices. However, if someone says, “Yeah but there is a better way,” then everyone listens and they decide to move forward with it as a team. If they fail, they dust themselves off and try again.

The following are great guiding principles to know that get you into the mindset of empirical process control:

- Iterative
- Incremental
- Frequent reviews
- Adaptation
- Uncertainty and risks during execution
- It isn’t a “defined” process; it’s a way of being and doing

Agile vs. Waterfall Project Management

Regardless of the methodologies you choose, in both Agile and Waterfall types of project management, the goal is to produce a result that meets spec and customer requirements. But what exactly is the difference between Agile and Waterfall project management? We know based on the Agile Manifesto that software development got a bit of a facelift once the decision was made to embrace frequent changes, frequent reviews, and adaptation, and also by putting certain aspects first, like people over formal process. That wasn’t always the case with software development projects (or other projects). The Waterfall method was created in the 1970s, and it was utilized in many large projects, including those within the United States Department of Defense. Many standards and books of knowledge were created from the Waterfall philosophy, including A Guide to the Project Management Body of Knowledge (PMBOK® Guide), Fifth Edition (Project Management Institute, 2013).

Waterfall and its model are attributed to an article written by Winston W. Royce, and even though he didn’t specifically use the term “Waterfall,” and he looked at the model as flawed in general, he is often the one credited with creating the Waterfall model.

In Royce’s original model, work flows from top to bottom, as does a Waterfall. In software development, the project begins with a collection of requirements and the documentation of those requirements. Once those requirements are collected, you can then move into the design phase of software architecture. Only after the architecture has been designed
can you move forward to project implementation. Once the product is complete, verification can take place. Approval of the final deliverable is needed before moving into the maintenance phase and/or operations and support.

While there are many opinions on the Waterfall model being ineffective for software development, the fact of the matter is that all projects can benefit from Waterfall best practices. Many of the best practices for Waterfall project management are found in the PMBOK® Guide. The Project Management Institute recognizes the need for tailoring projects and potentially having a combined approach between Waterfall and Agile project management (when needed) for the effective delivery of a result that meets requirements. It is important to note that if an organization is building a skyscraper, a bridge, or a highway system, Agile methods may not be the best way to manage those projects. It’s inherently better to have preapproved requirements, a plan in place, formal change control, and validation that the project requirements for scope, time, and cost are met and are monitored and controlled. In some cases, there is room for both.

The PMBOK® Guide embraces best practices of a formal life cycle much like the Waterfall approach where many things are preplanned before execution, and it follows a very formal change control system not found in many Agile approaches. Because technology and software development is a very large part of the project management industry globally, it is relevant to compare different best practices and have a flexible approach to producing the result for the best interests of the customer and the organization. It may also be relevant to mention that projects are unique and therefore one project might need a formal Waterfall type of approach and yet another might benefit strictly from Agile types of approaches like Scrum or XP. Still another project may need a more tailored project management approach with a combination of some Waterfall best practices and some Agile best practices.

The Benefits of Using Agile in Any Industry

The benefits of using Agile in other industries beyond software development are many. In order to tailor approaches that better meet the rapid pace of projects today, it is important for organizations to be able to respond to the swift changes in technology, competition for products, services, results, and changing requirements. That, however, is easier said than done.

For an organization to practice agility at the true organizational level, it must start with an understanding and internalization of Agile principles. Not until recently has there been much of a collective understanding of Agile methodologies across the board. It’s been something of a well-guarded secret for only software development projects or Scrum aficionados. Some organizations attempt to implement Agile best practices but are not using them effectively. It is important for effective implementation of any Agile methodology to practice the steps until proficient and to encourage others not only to internalize Agile best practices but to practice them regularly. The hard part is that a lot of organizations are used to their current process flow, and even though it may not be working sufficiently, change is always a bit painful.
Agile, as a term, is the umbrella over all of the different methodologies that you can absorb, use, and understand. To pass the PMI-ACP exam, you have got to get into that frame of mind. The Agile principles mindset involves the following:

- Exploring, embracing, and applying Agile principles while incorporating that mindset across the team and the organization
- Focusing value driven delivery on high-value increments and making sure that they are produced early and often
- Meeting and reviewing stakeholder priorities
- Gaining feedback on the increments you produce and then prioritizing and improving upon them

You’ll see some aspects of these principles across every single one of the different domains.

Regardless of the methodology to which you subscribe, you must be able to participate and collaborate. Those are the skills that are crucial to a successful project outcome.

You must make effective decisions and meet expectations and, at the same time, practice agility. There is a continually changing landscape requiring updates or changes to the value of what you’re trying to accomplish.

You can’t do that without team performance, and that is dependent on trust, learning, collaboration, and conflict resolution.

You’ll see quite a few questions on the exam about self-organizing teams utilizing emotional intelligence to enhance relationships, and how that helps to help create a culture of high performance. To do that is to be adaptive in your planning and have an evolving, changeable, and malleable plan from initiation all the way through closure.

From iteration to iteration and sprint to sprint, incorporate a lot of stakeholder feedback. During reviews, you’ll find out if it is working and if the result is what you thought it would be. How is risk impacting your project? Is it possible to reach fast failure where a risk event overshadows everything, resulting in a cancelled project?

All of this is based on stakeholder feedback and the ability to plan (but adapt) team performance, not to mention continuously identifying problems and impediments and continuously improving quality effectiveness and value of the product and/or service.
You also may have to go through an analysis and design process to get the organization on board. This can be a more difficult aspect than learning the ins and outs of Agile yourself and guiding your team to the understanding of what that methodology looks like. Some good questions to ask yourself and your teams are as follows:

- What are the guidelines?
- What are the processes that you can utilize to estimate how long things will take or how much work you can accomplish in a sprint or iteration?
- What interpersonal skills can you utilize to manage your team of individuals who are also self-managed? What different metrics should you follow closely and how do you plan, monitor, and adapt?

The big dance is “Agile,” and part of that dance is incorporating continuous improvement in the process:

- Making sure that you’re producing something to quality spec
- Managing risk
- Having the backlog or the work that you’re performing be of value to the customer
- Prioritizing the work in such a way that you are providing value consistently through each sprint/iteration of the project

Whatever works for you is cool, and that’s the beauty of agility.

Scope change is just our day-to-day routine. We try to get over that or get through it, but we know that change is going to occur, so we prepare ourselves for it. This means that continuous improvement is necessary. With Agile (think agility), you must be malleable to improve your best practices and products or services. You are self-driven, self-motivated, and self-managed, yes, but you are also coached in the best practices of the methodologies that you choose or a hybrid approach of several methodologies.

Work is planned at the last responsible moment with the expectation that things will change, which is completely the reverse of The *PMBOK® Guide* mentality of project management. What is interesting is that more Waterfall types of projects are now incorporating some of that Agile methodology, and even the PMI-ACP exam is adapted to accommodate more Agile types of approaches. The *PMBOK® Guide* is inclusive of Agile and tailoring approaches. This will make the PMI-ACP certification more relevant than ever. Soon more people will get certified, and more people will be implementing the many best practices available. Agile will improve knowledge work globally.
**Real World Scenario**

**Satir Model**

We will call our fictitious organization the ABC Company. ABC determined that an Agile approach to its projects was necessary. So it hired an Agile coach and sent its staff to a training program to learn the latest and greatest best practices in Agile project management. The team came back from training and was looking forward to implementing the best practices that they had learned.

The Agile coach worked on a regular basis to help them become more proficient in the practices they were using. Every day the team had a 10-minute stand-up meeting, and the team members would update their progress in a highly visible, low-tech, high-touch way of information radiating. At the same time, executive management was constantly asking for formal documentation, constant updates in the form of long meetings, and long-term projections of the schedule and the costs as well as demanding all of the updates that they were used to getting on other projects prior to the implementation of Agile.

Over time, the project team went back to the habits they had created prior to the training and implementation of Agile practices; the Agile coach became increasingly frustrated with the lack of support from senior management, and senior management deemed that Agile project management simply doesn’t work.

This is not a unique situation. Anytime an organization updates its processes, it is important for all levels to understand the best practices and to embrace the changes that will inevitably improve business exponentially. The hard part is the actual change.

The Satir model, created by Virginia Satir during her work as a family counselor, correctly depicts how organizations go from the status quo to the introduction of a foreign element and then resist the changes. Once resistance begins, chaos ensues. Most organizations throw away the best practices of Agile because of the chaos instead of realizing that chaos is necessary for the transformation of ideas and the integration of those innovative ideas in order to move forward to a new status quo.

ABC Company determined that chaos was the new norm and never got to the point where the entire organization engaged in the transforming ideas and integration of new best practices, and therefore it went back to the old status quo. Millions of dollars were spent and lost because chaos won.

**Satir Model**

The Satir model was created by Virginia Satir for changes in family dynamics during counseling. It applies to implementing change organizationally. In Figure 1.1 you will see the Satir model visually represented.
Organizational Agility Tips

To implement Agile methodologies successfully, your organization will need to do the following:

- If your team or organization is new to Agile, it is best to use a proven method of best practices and then to try to create a hybrid approach.
- Implement Agile on the organizational level starting with an understanding and internalization of Agile principles.
- Practice the steps of a chosen Agile methodology until proficient.
- Encourage others to internalize and practice Agile.
- Learn the techniques before throwing a methodology out as not working.
- Embrace the higher criticality and weight of method based on team size and project length.

Summary

This chapter started with a discussion of the roots of Agile’s history, which served as the springboard for the Agile Manifesto and multiple types of frameworks and methodologies. Next, we reviewed the 12 principles of Agile that influence every aspect of Agile project management regardless of type. The Agile Manifesto and its principles are excellent to keep in mind whether you are implementing some type of process on your projects or answering questions on your exams.

While Agile differs from a Waterfall type of project management, it is relevant to mention again that it is possible to have best practices come from each as needed on your unique projects. All of what we will discuss in this book are best practices, and they can be adapted as needed for your organization.
Even though Agile has been around in some way since the early 1990s, it is just now becoming more relevant across multiple modalities, project types, and organizations due to the technology age and cutting-edge knowledge work taking place in all industries globally. All can benefit from either an Agile approach to projects or a tailored approach between aspects of Waterfall and Agile project management.

Exam Essentials

Be able to understand and embrace the Agile Manifesto. Know how Individuals and Interactions over Processes and Tools, Working Software over Comprehensive Documentation, Customer Collaboration over Contract Negotiation, and Responding to Change over Following a Plan are important aspects of all Agile methodologies. Know that the Agile Manifesto will help you answer many questions on the exam.

Be able to embrace the principles of the Agile Manifesto. The principles include customer satisfaction, welcome changes, frequent delivery, collocated teams, motivated individuals, face-face contact, working software, constant/sustainable pace, continuous attention, simplicity, self-organization, and regular reflection. These principles will be excellent reminders of Agile, and you will see them reflected many times throughout all domains.

Be able to understand the Declaration of Interdependence and empirical process control. Know how both aspects led to better definition of the principles and the development of Agile frameworks/methodologies like Scrum and XP.

Understand the differences between Agile and Waterfall. Agile is focused on rapid production of a working increment in an iterative fashion. Iterations are run for abbreviated time spans, which allow continuous feedback and reflection with a focus on continuous improvements.

Waterfall project management is more involved with long-term planning and making sure that the plan is in place and approved before execution of work begins. Changes are embraced, but they follow a strict change control system throughout the life cycle. Heavier documentation, larger teams, and more tangible products, services, or results are created. Hybrid methods of Agile/Waterfall are becoming more popular today.

Be able to place yourself in an Agile type of environment and get into the Agile mindset. If you are new to Agile or specialize on one type of framework, then it is important to keep an open mind about diverse ways of running an Agile project. You must always keep in mind the influencers and their desire for a better way of producing valuable results.

Be able to utilize the Agile Manifesto in order to answer questions appropriately. This is probably the most important aspect outside of memorizing the manifesto for the exam. Every question will relate to your ability to be in the mindset of best practices and determine the best answer based on the principles of Agile.
Review Questions

You can find the answers to the review questions in Appendix B.

1. What is the first value in the Agile Manifesto?
   A. Contract Negotiations over Working Plans
   B. Working Software over Comprehensive Documentation
   C. Individuals and Interactions over Processes and Tools
   D. Comprehensive Documentation over Customer Interaction

2. Who were the creators of the Agile Manifesto?
   A. PMI
   B. The Agile Alliance
   C. Ken Beck
   D. The Scrum Alliance

3. In the Agile Manifesto, it is more important to respond to changes than to do which of the following?
   A. Comprehensively document
   B. Use processes and tools
   C. Negotiate contracts
   D. Follow a plan

4. Which of the following is the highest priority of the 12 major principles?
   A. Face-to-face conversation
   B. Motivated individuals
   C. Working software
   D. Customer satisfaction

5. You are working to implement Agile in your organization, and a key stakeholder asks you to explain Agile to them. What would be the best answer to explain Agile to someone who doesn’t know anything about it?
   A. Agile is a way to produce software without change control.
   B. Agile is a framework for better project management.
   C. Agile is a mindset that allows an organization to focus on better ways to produce value for the customer and practice continuous improvement.
   D. Agile is the opposite of Waterfall.
6. Bill is a key stakeholder in your organization and has recently learned about your usage of an Agile framework on your current project. You’ve invited Bill to a planning meeting, and he says, “I thought you didn’t do formal planning in Agile.” How should you respond?
   A. “We do plan our projects, but we keep our focus on the immediate items and we value working software over comprehensive documentation during planning.”
   B. “We plan as much as anyone working on projects and try to incorporate the opinions of many stakeholders to help us do it.”
   C. “We don’t plan normally, but since you are new to Agile we thought we would plan as much as possible until you get comfortable with Agile.”
   D. “We build plans that we know won’t work to meet stakeholder requirements.”

7. Empirical process control has three main aspects to it. What would those three be?
   A. Transparency, inspection, and adaptation
   B. Iterative, incremental, and uncertainty
   C. Transparency, iterative, reliability
   D. Inspection, incremental, adaptable

8. In the past, your team has run projects using a more formal Waterfall method and now is incorporating Agile methodologies. What will be the biggest difference in how you manage your projects?
   A. Waterfall preplans the entire project before execution, and Agile only plans out just far enough to create a workable increment that is valuable to the customer.
   B. Agile is much more flexible in how the best practices work. As long as your team creates something usable, it doesn’t matter how you accomplish it.
   C. The Waterfall method has many books of knowledge and best practices, but Agile only has the Manifesto.
   D. Agile plans less, and Waterfall plans more.

9. The Satir model correctly predicts why some organizations have a challenging time changing methodologies. What is the reason for that based on the model?
   A. Organizations don’t spend enough time training their people and that causes chaos.
   B. Organizations often determine that the new method isn’t working because they have not reached the integration stage and have only seen the chaos.
   C. Organizations are used to doing things a certain way and nobody likes to change.
   D. Organizations don’t allow enough time for the chaos, which in turn doesn’t allow things to level out in integration.

10. What is the main reason organizations should not jump into a hybrid method if they are new to Agile approaches?
    A. It takes practice to implement Agile effectively on an organizational level. It’s better to practice one method and become proficient before trying to combine two.
    B. Organizations that have used both should determine which one works the best individually before creating a hybrid option.
C. Hybrids are not recommended for large software organizations.
D. It can get very confusing as to which frameworks to follow if it is a combination of two types.

11. Seventeen developers met in Snowbird, Utah, in February 2001 to discuss better ways of managing software projects. What was the result of that meeting?
   A. The Declaration of Interdependence
   B. The Agile Manifesto
   C. The Scrum Theory
   D. The PMBOK® Guide

12. What does the Agile Manifesto mean by “individuals and interactions over processes and tools”?
   A. It means that processes and tools aren’t needed on Agile projects.
   B. It means that without processes and tools, the individuals and interactions will not be effective.
   C. It means that individuals and interactions are valued more than processes and tools.
   D. It means that both are necessary and work together to produce working software.

13. Which of the following best describes why Waterfall project management isn’t the most effective way to manage software development projects?
   A. The rapid pace and constant changes make it difficult to preplan and create baselines.
   B. The Agile life cycles are different and therefore incompatible with Waterfall.
   C. Waterfall is for construction, and Agile is for software.
   D. Waterfall is an effective way to manage software projects as long as you have the scope of work up front.

14. Complete the rest of this statement from the Agile Manifesto: Individuals and interactions over ____________.
   A. Following a plan
   B. Comprehensive documentation
   C. Contract negotiation
   D. Processes and tools

15. Complete the rest of this statement from the Agile Manifesto: Responding to change over ____________.
   A. Following a plan
   B. Comprehensive documentation
   C. Contract negotiation
   D. Processes and tools
16. Complete the rest of this statement from the Agile Manifesto: Working software over
______________.
A. Following a plan
B. Comprehensive documentation
C. Contract negotiation
D. Processes and tools

17. Complete the rest of this statement from the Agile Manifesto: Customer collaboration over
______________.
A. Following a plan
B. Comprehensive documentation
C. Contract negotiation
D. Processes and tools

18. In the Declaration of Interdependence, which of the following increases by making continuous flow of value the focus?
A. Reliable results
B. Creativity and innovation
C. ROI
D. A boost in performance

19. What is the best way to explain Agile project teams based on the values of the Agile Manifesto?
A. Managed by a project manager
B. Managed by an Agile project manager
C. Not managed by anyone
D. Self-directed and self-managed

20. Agile project teams work best in what dynamic?
A. Collocated
B. Virtual
C. A combination of collocated and virtual
D. A cross-functional dynamic