Chapter 17 PMP® Bootcamp: Agile Fundamentals Worksheet

Student Name: _____ Date: _____

For Students New to Project Management

Objective: To understand the core concepts, values, and common methodologies of the Agile approach to project management. This will provide a foundational understanding for both the PMP exam and modern project work.

Instructions: As we discuss each topic in class, use this worksheet to follow along, fill in the blanks, and jot down key ideas. Think about how these concepts differ from a traditional, predictive (Waterfall) approach.

Part 1: The Foundation – The Agile Mindset

Agile isn't just a process; it's a **way of thinking**. It's about embracing change, collaborating closely with the customer, and delivering value in small, frequent increments.

The Agile Manifesto: 4 Key Values

The Manifesto prioritizes certain values. This doesn't mean the items on the right have *no* value, just that we value the items on the left *more*.

We Value...	More Than...	In Simple Terms, This Means...
Individuals and interactions	Processes and tools	People talking to each other is more effective than relying on complex tools or rigid processes.
Working software	Comprehensive documentation	Our primary measure of progress is a product that works, not how much we've written about it.
Customer collaboration	Contract negotiation	We work *with* our customer throughout the project, rather than defining everything upfront in a strict contract.
Responding to change	Following a plan	We expect and welcome change as an opportunity to make the product better, instead of resisting it.

The 12 Agile Principles (Summarized into Key Themes)

The 12 principles are the "how-to" guide for living the Agile values. Let's group them:

1. **Customer Focus & Value Delivery:** Our highest priority is to satisfy the customer through early and continuous delivery of valuable software. We deliver working features frequently.
2. **Embrace Change:** We welcome changing requirements, even late in development.
3. **Team Empowerment & Collaboration:**
    - Business people and developers must work together daily.
    - Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
    - The best way to convey information is **face-to-face conversation**.
4. **Focus on Simplicity & Technical Excellence:**
    - Simplicity—the art of maximizing the amount of work *not* done—is essential.
    - Continuous attention to technical excellence and good design enhances agility.
5. **Iterative Process & Reflection:** At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

The Agile Triangle: A New Perspective on Constraints

In traditional projects, we fix the **Scope** and estimate **Time** and **Cost**. In Agile, we flip this.

- **Traditional Triangle:**
    - **Fixed:** Scope (What we will build)
    - **Variable:** Time & Cost (Our estimates of how long/much it will take)
- **Agile Triangle (The "Iron Triangle" Inverted):**
    - **Fixed:** Time (Sprints/Iterations) & Cost (Team size)
    - **Variable: Scope** (The features we deliver are adjusted to fit the fixed time/cost)

Key Takeaway: Agile prioritizes delivering the **highest value features** within a fixed budget and schedule.

Part 2: The "How" – Common Agile Methodologies & Frameworks

1. Lean

- **Core Idea:** Eliminate waste and focus only on what creates value for the customer. "Waste" is anything that doesn't add value.

- **Think of it as:** Tidying up a messy garage. You only keep the useful tools (value) and get rid of everything else (waste).
- **Key Concept: Value Stream Mapping** - Visualizing all the steps from a customer request to delivery to identify and remove delays and waste.

2. Kanban

- **Core Idea:** A visual method for managing workflow. It helps you see bottlenecks and improve the flow of work.
- **Think of it as:** A digital or physical whiteboard with columns: **To Do | In Progress | Done**.
- **Key Concepts:**
    - **Visualize the Workflow:** See all the work in one place.
    - **Work-In-Progress (WIP) Limits:** Limit how many items can be in the "In Progress" column at one time. This prevents overload and forces the team to finish work before starting new work.

3. Scrum (The Most Popular Agile Framework)

- **Core Idea:** A framework for developing and delivering complex products in short cycles called **Sprints**.
- **Think of it as:** A series of short races (Sprints) to build a product piece by piece. After each race, the team shows what they built and plans the next race.

Scrum Core Concepts (The 3 Pillars):

1. **Transparency:** Everyone can see what's happening. The work is visible on a board, and progress is known to all.
2. **Inspection:** The team regularly checks its progress toward the Sprint Goal and the quality of the product.
3. **Adaptation:** If inspection shows that things are off track, the team adjusts its process or the product itself.

Scrum & Agile Generic Terms (Vocabulary Builder):

Term	Simple Definition

Product Owner	The "voice of the customer." Owns the Product Backlog and prioritizes features to maximize value.
Scrum Master	A servant-leader who helps the team use Scrum effectively and removes impediments (blockers).
Developers	The people who do the work of creating the product increment (e.g., designers, programmers, testers).
Sprint	A time-boxed event of 1-4 weeks where the team works to create a usable product **Increment**.
Product Backlog	A single, prioritized list of everything needed for the product (features, fixes, etc.). Managed by the Product Owner.
Sprint Backlog	A list of items from the Product Backlog that the team commits to completing in the current Sprint.
Increment	The sum of all the Product Backlog items completed during a Sprint. It must be usable and meet the **Definition of Done**.
Definition of Done (DoD)	A shared checklist of what it means for work to be "complete" (e.g., coded, tested, documented).
User Story	A simple way to describe a feature from a user's perspective: "As a [type of user], I want [some goal] so that [some reason]."
Velocity	The average amount of work a team completes during a Sprint. Used for forecasting, not for comparing teams.

The Daily Scrum (or "Daily Stand-up"):

- **Purpose:** A 15-minute planning meeting for the Developers, *not* a status report for managers.
- **Goal:** To coordinate activities and create a plan for the next 24 hours.
- **The 3 Questions (Classic format):**
    1. What did I do yesterday to help the team meet the Sprint Goal?

2. What will I do today to help the team meet the Sprint Goal?
3. Do I see any impediments (blockers) that prevent me or the team from meeting the Sprint Goal?

4. XP (Extreme Programming)

- **Core Idea:** A software development framework focused on **technical excellence** and high-quality code.
- **Think of it as:** Scrum with a heavy emphasis on engineering best practices.

XP Values (Mnemonic: CRRCS):

Value	Meaning
Communication	Everyone is part of the team and we communicate face-to-face.
Respect	Team members respect each other's expertise and contributions.
Simplicity	What is the simplest thing that could possibly work?
Courage	To tell the truth about progress, refactor code, and adapt.
Feedback	From the customer, the system, and each other, as rapidly as possible.

Key XP Practices:

- **Pair Programming:** Two developers work at one computer to write code. This improves quality and spreads knowledge.
- **Test-Driven Development (TDD):** Write a failing automated test *before* writing the code to make it pass.
- **Continuous Integration (CI):** Developers merge their code into a central repository frequently, where it is automatically built and tested.

5. Other Methodologies to Know

- **Crystal Family:** A collection of methodologies that are people-centric and adaptive. The idea is that the process should be tailored to the size and criticality of the team/project.
    - **Key Idea:** Processes are "tuned" for the team, not the other way around.

- **DSDM (Dynamic Systems Development Method):** An older Agile method that is very formal and focuses on "fitness for business purpose."
    - **Key Idea:** Uses timeboxing and prioritizes requirements using MoSCoW (Must have, Should have, Could have, Won't have).

Part 3: Scaling Up – Agile for the Enterprise

Scaled Agile Framework (SAFe®)

- **What is it?** A framework for applying Agile, Lean, and DevOps principles at a very large, enterprise scale.
- **Core Idea:** It synchronizes many Agile teams into an "Agile Release Train" (ART) that work together to deliver large-scale solutions.
- **Key Takeaway:** When a single Scrum team isn't enough to build a product (e.g., building a new car or a banking system), frameworks like SAFe® provide the structure to coordinate everyone's work.

Part 4: Review & Reflection

Answer the following questions to solidify your understanding.

1. In your own words, what is the biggest difference between the traditional "iron triangle" and the Agile "inverted triangle"? What does this mean for the project's scope?

2. A stakeholder asks your team to skip the Daily Scrum for a week to "save time and focus on coding." As a team member, how would you explain the value of this meeting using the Scrum pillars (Transparency, Inspection, Adaptation)?

3. Imagine you are on a team building a mobile app. The Product Owner adds a new User Story to the Product Backlog: *"As a user, I want to log in with my fingerprint so that I don't have to type my password."* What methodology (XP, Scrum, Kanban) would be most helpful for ensuring the login feature is secure and high-quality? Why?