

# Introduction to Scrum Theory

Welcome to the world of Scrum theory! This foundational knowledge is crucial for understanding Agile project management and passing the Scrum Master certification exam. We'll explore the core concepts that make Scrum an effective framework for complex projects.

by Mayko Silva



# Core Concepts of Scrum

## Empiricism

The first pillar of Scrum theory. It emphasizes knowledge gained through experience and observation. This approach allows teams to adapt based on real-world feedback.

## Lean Thinking

The second pillar focuses on maximizing efficiency. It aims to reduce waste and concentrate on essential tasks. This principle helps teams deliver more value with less effort.



# Empiricism in Scrum

## 1 Doing

Teams actively engage in project work, gaining hands-on experience with tasks and challenges.

## 2 Inspecting Results

Regular review of outcomes allows teams to gather data on their performance and progress.

## 3 Adapting Approach

Based on inspection results, teams modify their strategies to improve efficiency and effectiveness.



# Lean Thinking in Practice

## Waste Reduction

Identify and eliminate unnecessary steps, meetings, or documentation that don't add value to the project.

## Focus on Essentials

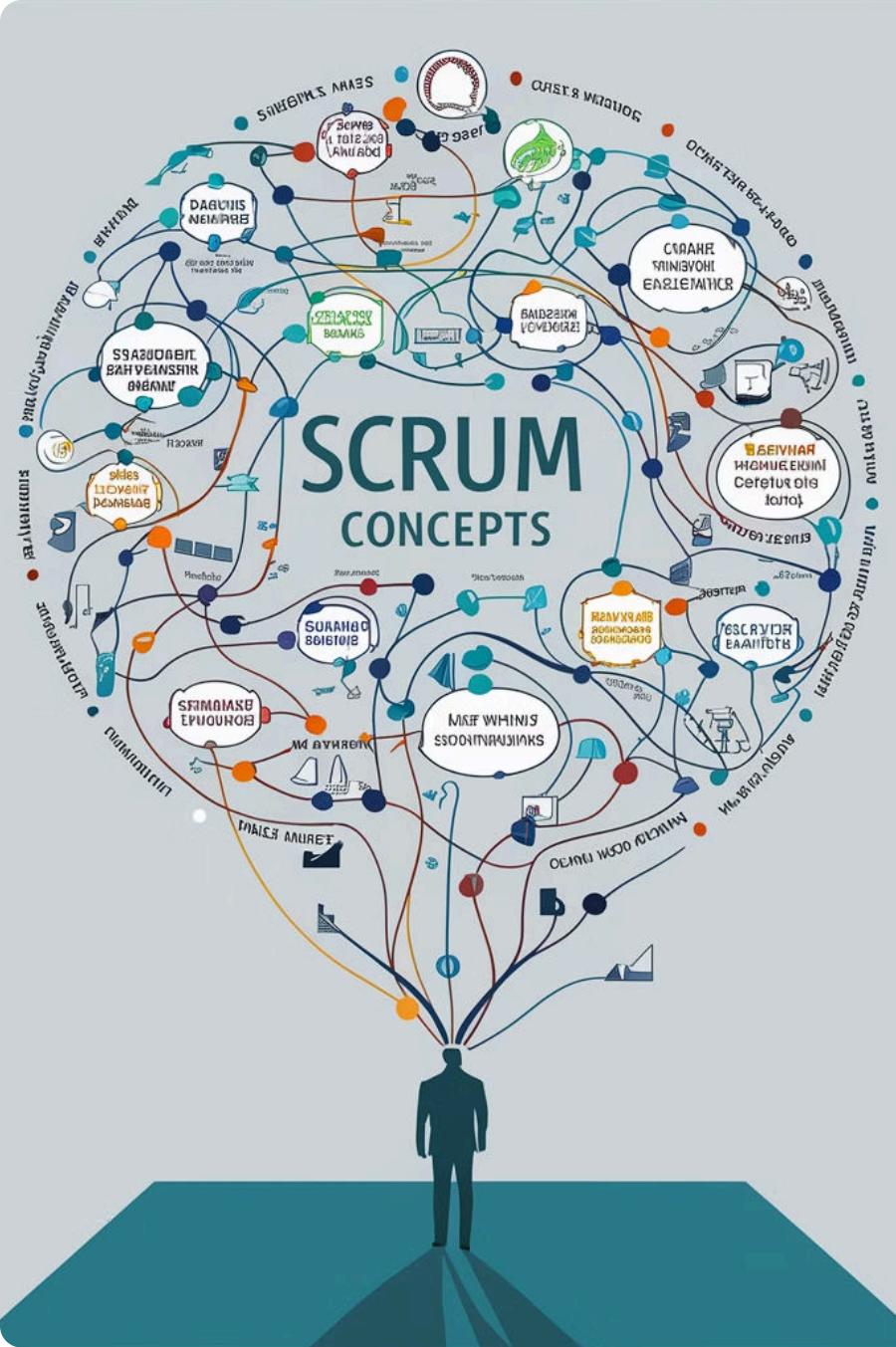
Prioritize tasks that directly contribute to project goals and customer satisfaction.

## Process Streamlining

Continuously refine workflows to remove bottlenecks and improve efficiency.

## Value Maximization

Ensure that every sprint delivers tangible value to stakeholders and end-users.



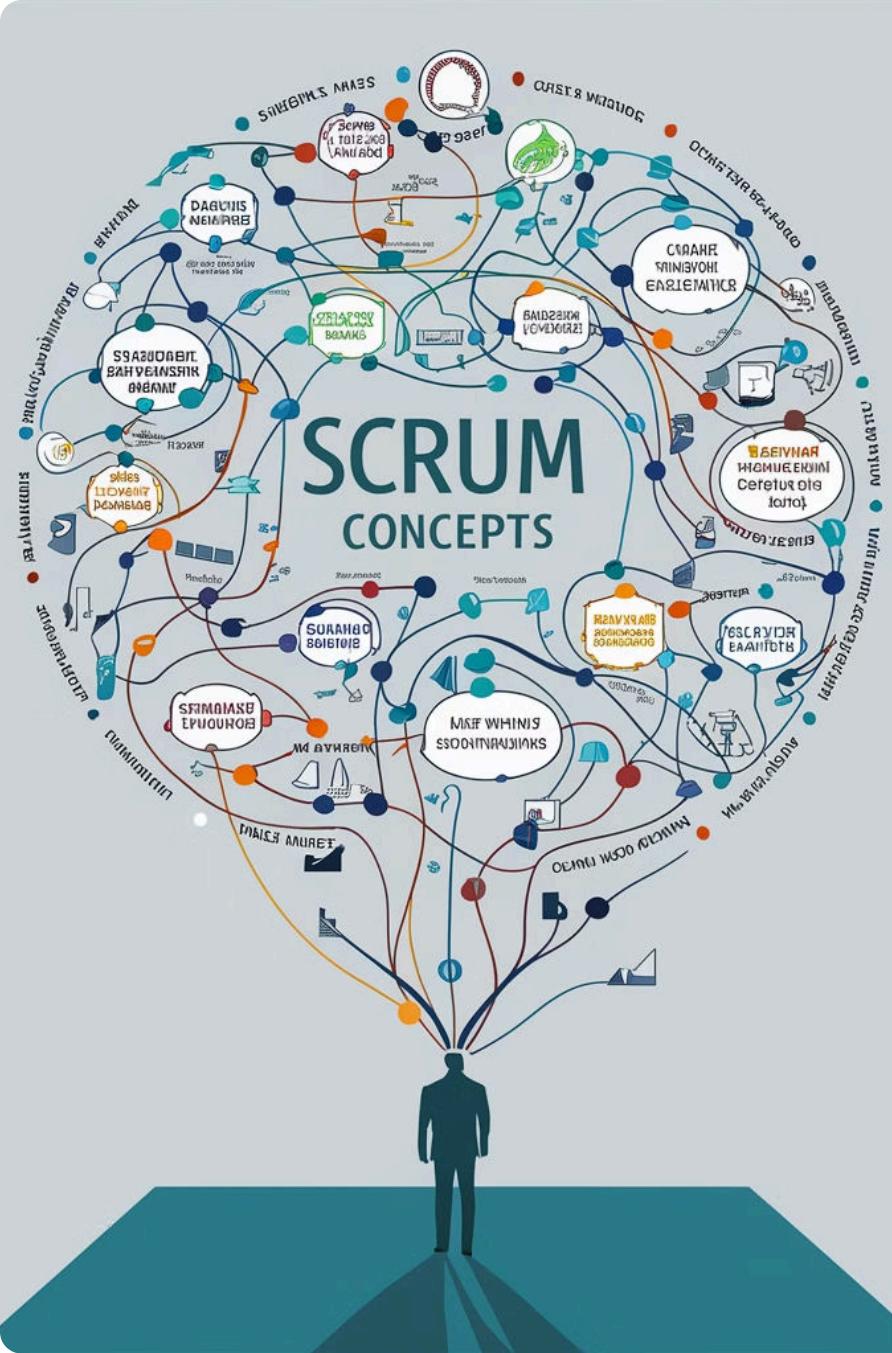
# Exploring Scrum Theory

## Rationalist vs Empiricist

# Practical Examples

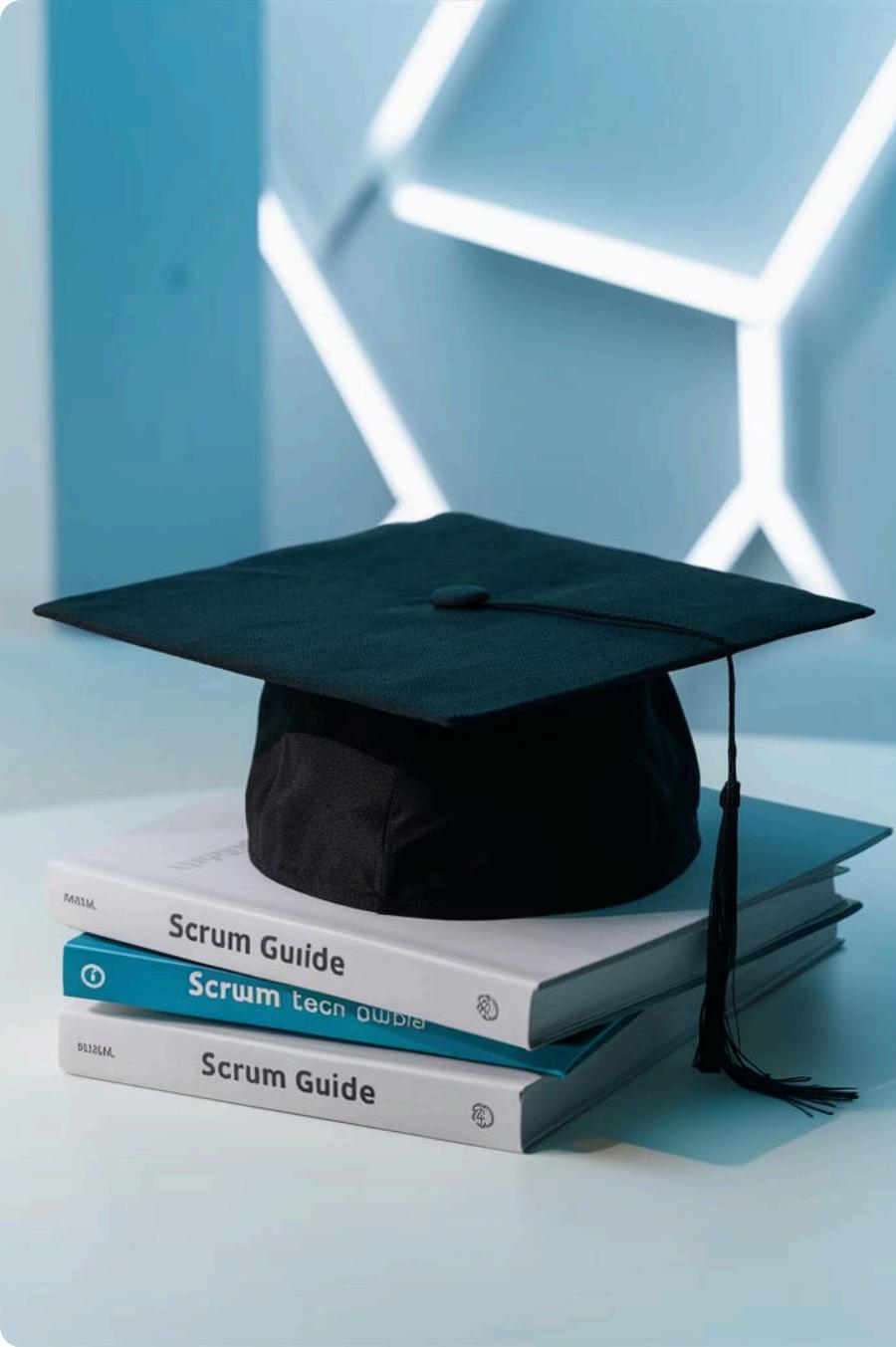
# Lean thinking principles in Scrum

# Integration of empiricism and lean thinking



# Exploring Scrum Theory

- Managing predictability and risk
  - Incremental nature of Scrum
  - Importance of cross-functional teams
  - Sprint transitions
  - Maintaining focus on the goal



# Learning Outcomes

## 1 Theoretical Foundations

Gain a deep understanding of the principles that underpin Scrum methodology.

## 2 Scrum Mechanics

Comprehend why Scrum processes are structured the way they are.

## 3 Certification Preparation

Master key concepts essential for success in the Scrum Master certification exam.



# Ready to Dive In?



## Study

Dive deep into Scrum theory and principles.



## Apply

Implement Scrum concepts in real-world scenarios.



## Certify

Prepare for and ace your Scrum Master certification exam.

# Rationalists vs Empiricists

Rationalists believe that knowledge comes primarily from reason, while empiricists argue that it comes from sensory experience. This philosophical debate has shaped Western thought for centuries.

— by Mayko Silva



# Two Approaches to Problem-Solving



## Rationalist Approach

Relies on logic, reason, and deductive reasoning to solve problems. Focuses on developing theories and models to understand the world.



## Empiricist Approach

Emphasizes observation, experimentation, and inductive reasoning to gain knowledge. Focuses on gathering data and testing hypotheses.

# Rationalist Approach

## Knowledge Derived

Rationalists believe knowledge is derived through reason, logic, and deduction rather than empirical observation.

## Theoretical Models

Rationalists rely on theoretical models and abstract thinking to understand the world.

## Example: Mathematics

Mathematics is a prime example of a rationalist field, where knowledge is built through logical proofs and deductive reasoning.



# **Empiricist Approach**

## **Knowledge from Experience**

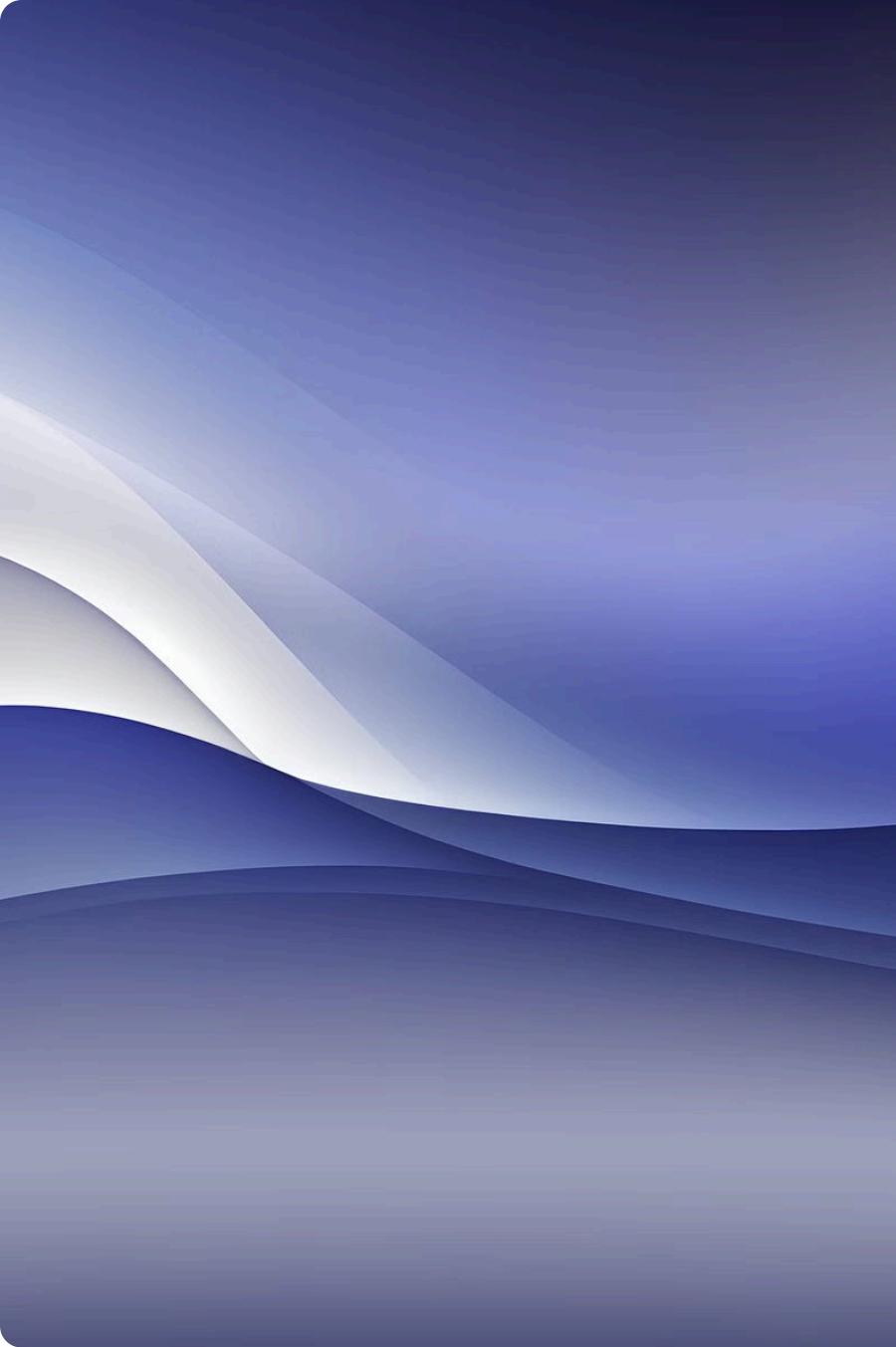
Empiricists believe that knowledge comes from experience and observation, not just pure reason.

## **Learning through Experimentation**

Empiricists advocate learning through experimentation and gathering real-world evidence, rather than relying solely on deductive reasoning.

## **Example: Scientific Method**

The scientific method, with its emphasis on hypothesis testing and data collection, is a prime example of the empiricist approach.



# Where Does Scrum Fall?

Scrum is firmly rooted in empiricism, which means it relies on experimentation and observation rather than pure logic or theory. This approach is well-suited for complex, uncertain environments where traditional planning methods may not be effective.

# Why Scrum is Empirical



## Complex Environments

Scrum is designed for complex, unpredictable environments where traditional planning methods may not work.



## Inspection & Adaptation

Scrum emphasizes continuous inspection and adaptation to respond to changes and challenges.



## Transparency

Scrum promotes transparency, ensuring everyone has a clear understanding of the project's progress.



# Practical Example: Mobile App Development

1

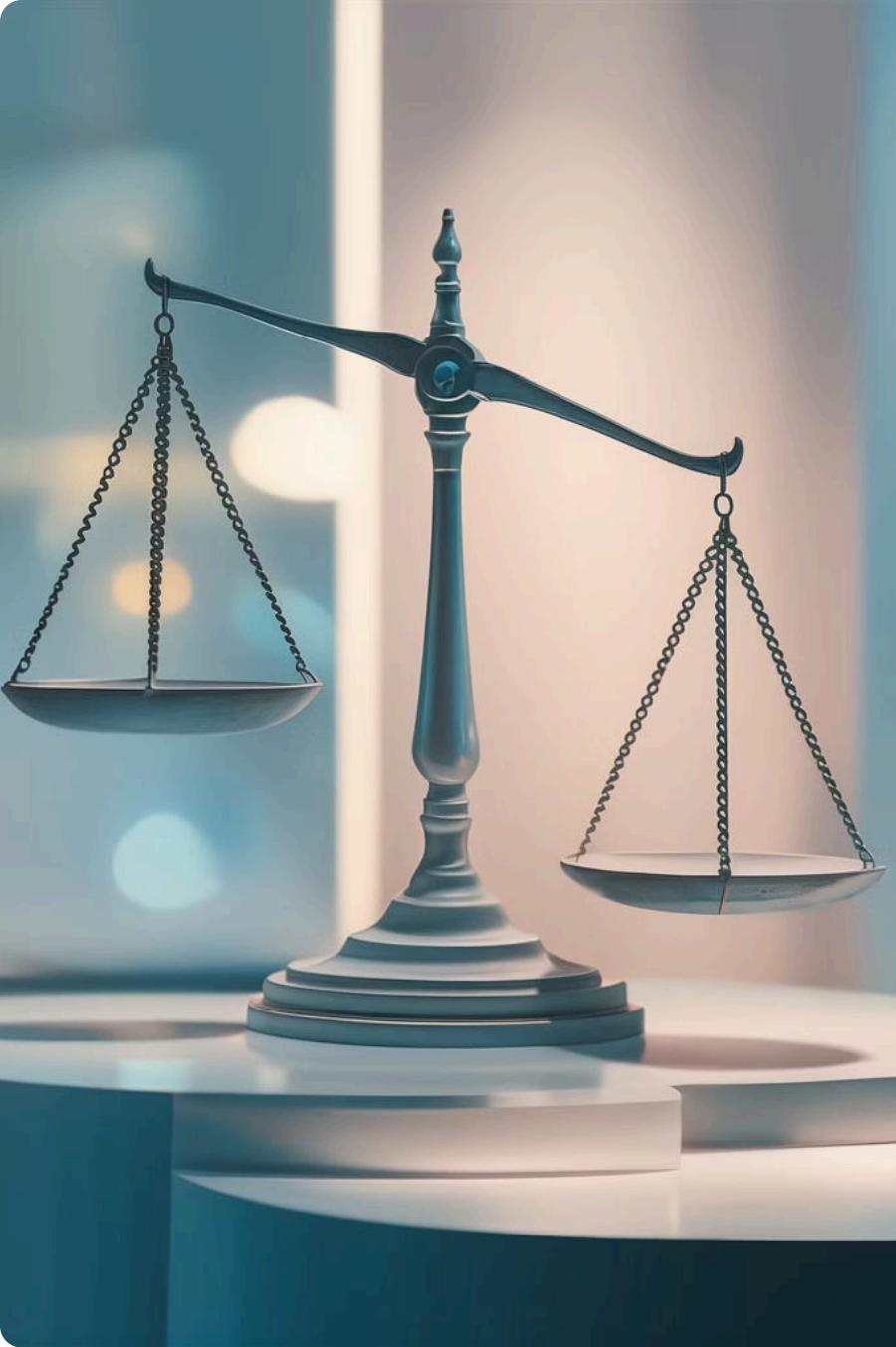
## Rationalist Approach

Detailed six-month plan based on market research and theoretical user behavior.

2

## Empiricist (Scrum) Approach

1. Create minimal viable product (MVP)
2. Release to small user group
3. Gather feedback
4. Make improvements based on feedback
5. Repeat cycle



# Scrum's Balance

## Primarily Empirical

Scrum focuses on empirical processes, relying on feedback and experimentation to guide decision-making.

## Doesn't Discard Rational Thinking

Scrum recognizes the value of rational planning and analysis, but balances it with an empirical approach.

## Provides Structure for Empirical Processes

Scrum's framework and ceremonies create a structured environment for empirical processes to thrive.

# **Key Takeaway**

## **Unknown Outcomes**

In complex environments, what will happen is unknown.

## **Guided by the Past**

Only past events can guide future decisions.

## **Essence of Empiricism**

This is the essence of empiricism in Scrum.

# **Reflection**

Reflecting on problem-solving approaches, I recall using an empirical approach that relied more on observation and experimentation. This differed from a rationalist approach focused on logic and deduction. The empirical method allowed for more flexibility and adaptation to changing circumstances.



# Next Steps



## Explore Empiricism

We'll dive deeper into how empiricism manifests in Scrum practices.



## Apply Empirical Thinking

Consider how to apply empirical thinking to your current projects.



# Thank You



## For your attention

Thank you for your time and engagement throughout this presentation.



## Looking forward to continuing our exploration of Scrum theory!

We've covered a lot of ground, and I'm excited to continue our discussion on the principles and practices of Scrum.

# Empirical Example: The Hockey Puck Experiment

Explore the fascinating physics behind the motion of a hockey puck through an engaging real-world experiment. Observe the principles of momentum, friction, and energy transfer in action.

by Mayko Silva



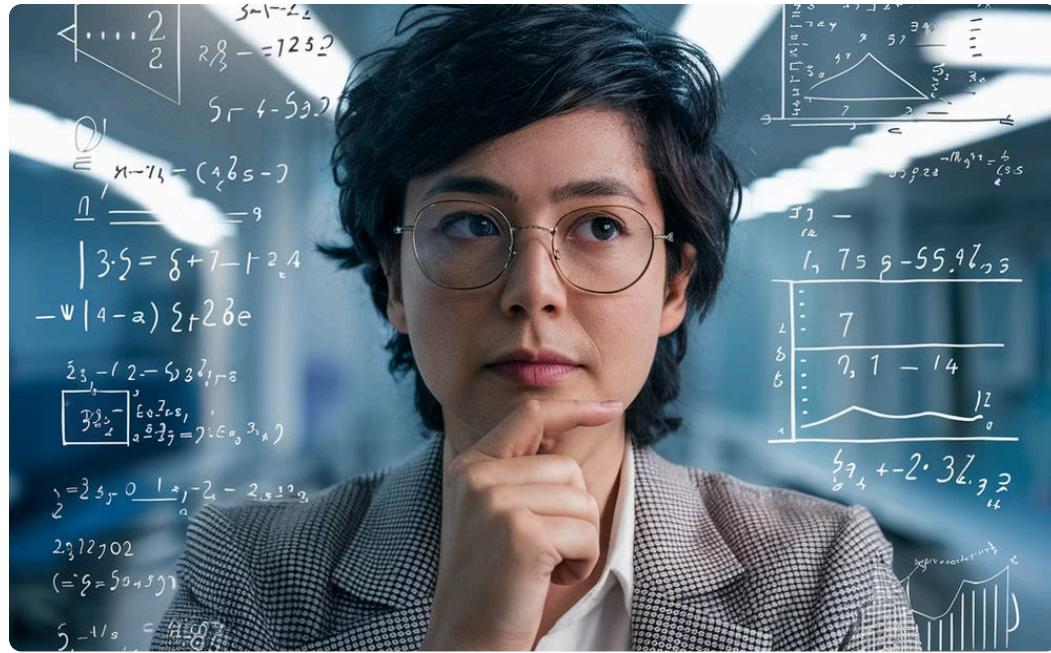


# The Question

## **Intriguing Inquiry**

How long would it take for a hockey puck to fall from the top of the CN Tower in Toronto to the ground?

# Two Approaches



## Rationalist Approach

Relying on logic, reason, and mathematical models to understand and predict phenomena.



## Empirical Approach

Basing understanding on systematic observation and experimentation to gather evidence.

# Rationalist Approach

## Mathematical Formulas

The rationalist approach relies on using mathematical formulas to calculate factors like the height of the CN Tower, gravity, and wind resistance.

## Theoretical Models

This approach considers theoretical models and calculations to predict the behavior of the hockey puck, rather than empirical observation.



# Empirical Approach



## Hands-On Testing

The Scrum team prefers to actually drop hockey pucks from the CN Tower and time how long they take to hit the ground.



## Collect Data

By gathering real-world data through experimentation, the team can make observations and draw conclusions about the behavior of the hockey pucks.



## Empirical Observation

This empirical approach allows the team to directly observe and measure the phenomenon, rather than relying solely on theoretical models.

# Scrum Team's Empirical Approach

1

## Plan the Experiment

The Scrum team carefully plans the experiment, defining the goals, variables, and success criteria.

2

## Conduct the Experiment

The team executes the experiment, closely observing and recording the results.

3

## Collect Data

The team gathers all relevant data from the experiment, ensuring it is accurate and comprehensive.

4

## Analyze Results

The team thoroughly analyzes the data, looking for patterns, insights, and learnings.

5

## Draw Conclusions

Based on the analysis, the team draws conclusions about the effectiveness of the approach.

6

## Iterate if Necessary

If the results are not satisfactory, the team refines the experiment and repeats the process.

# Essence of Empiricism in Scrum

## Based on Direct Observation

Scrum relies on empirical data gathered through direct observation and experience, rather than theoretical assumptions.

## Allows for Inspection

The Scrum framework provides opportunities to inspect real results and adapt accordingly.

## Enables Adaptation

Empiricism in Scrum empowers teams to make data-driven decisions and adjust their approach as needed.



# Why Empirical Over Mathematical?

## Complex Environments

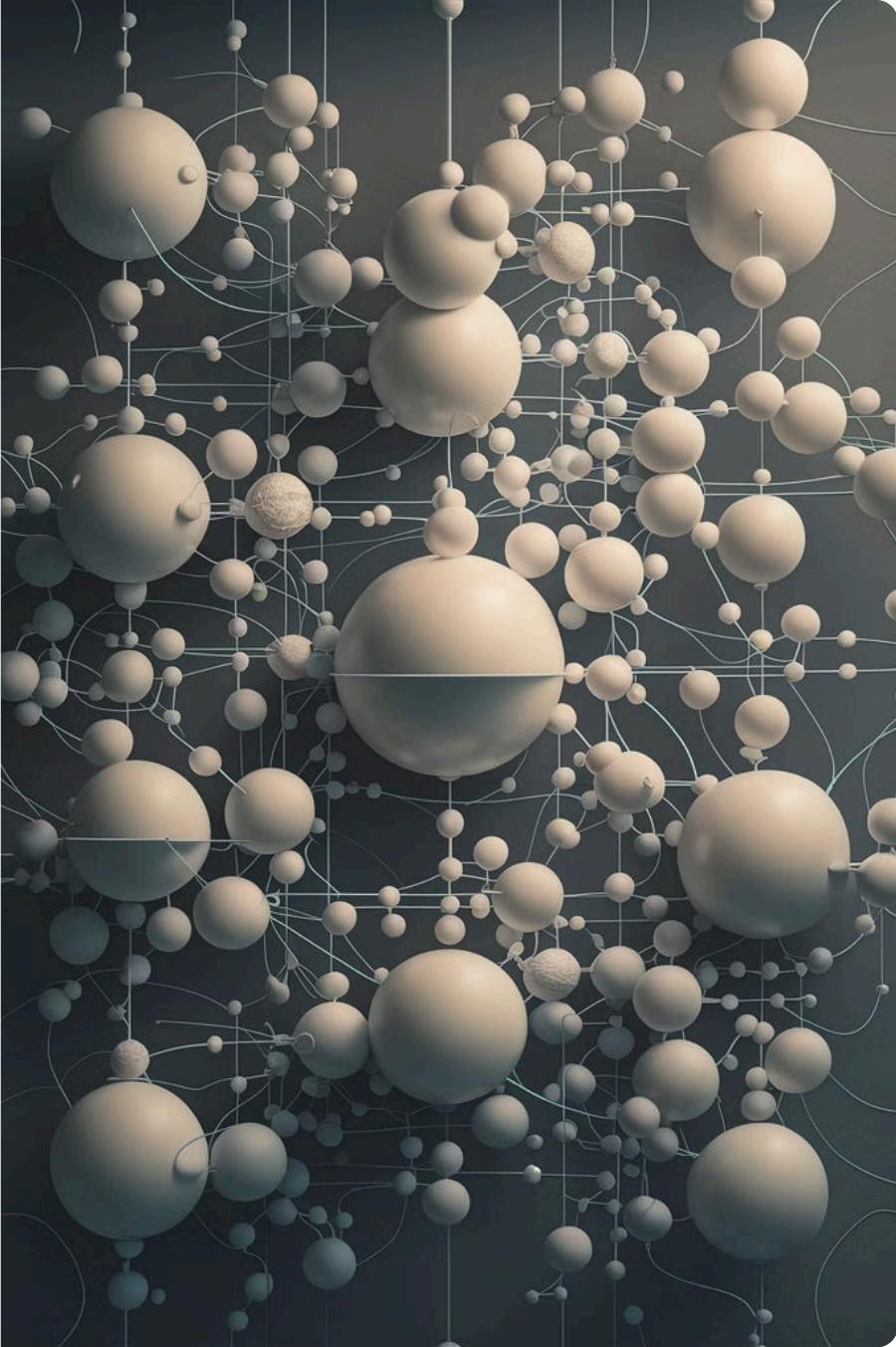
Complex real-world environments often defy theoretical models, making an empirical approach necessary.

## Accounting for Factors

Empirical methods capture all relevant factors, even those that may not be accounted for in mathematical calculations.

## Unanticipated Outcomes

An empirical approach allows for the discovery of unanticipated outcomes that may not be predicted by theoretical models.





# Why Empirical Over Mathematical?

- 1 Embracing Uncertainty**  
Scrum acknowledges that complex problems cannot be solved through rigid mathematical models alone. It embraces empiricism to navigate the unpredictable.
- 2 Continuous Learning**  
Scrum teams continuously inspect and adapt based on real-world feedback, allowing them to respond to changing conditions.
- 3 Adaptability**  
The empirical approach enables Scrum teams to be more agile and responsive, adjusting their plans as they learn from experience.

# Reflection

Can you recall a time when an empirical approach led to insights or solutions that a purely theoretical approach might have missed? This is the essence of empiricism - relying on real-world observation and experimentation to uncover truths, rather than just abstract reasoning.

# Next Steps

## Explore Empiricism

We'll dive deeper into how the empirical mindset applies to software development and other complex projects.

## Real-World Examples

We'll look at case studies and real-world examples of teams successfully using an empirical approach.



# Remember



## Evidence Over Theory

Focus on empirical evidence rather than relying solely on theoretical assumptions.



## Observation Over Assumption

Make decisions based on direct observation and data, not just preconceived notions.



## Adaptation Over Rigid Planning

Be flexible and responsive to change, rather than sticking to a rigid plan.

# Thank You

For your attention. Looking forward to continuing our exploration of Scrum!



# Lean Thinking in Scrum

Explore the principles of Lean Thinking and how they are applied within the Scrum framework to drive efficient and effective software development.

— by Mayko Silva



# Origins of Lean Thinking



## Toyota's Manufacturing

Lean thinking originated from Toyota's manufacturing practices in the 1950s and 1960s.



## Car Manufacturing

Initially, lean principles were applied to car manufacturing, but they are now universal.



## Product Development

The principles of lean thinking can be applied to any product development process.





# What is Lean Thinking?

## Maximizing Value

Lean thinking focuses on delivering the maximum value to customers by identifying and eliminating waste in processes.

## Minimizing Waste

The core concept of lean is to minimize waste in all forms - time, resources, effort, and anything that does not directly contribute to customer value.

## Continuous Improvement

Lean encourages a mindset of continuous improvement, constantly evaluating and refining processes to become more efficient and effective.

# Key Concepts of Lean Thinking in Scrum

## Eliminate Unnecessary Steps

Focus on only the essential tasks required to deliver value. Identify and remove any redundant or non-essential activities.

## Tie Work to Specific Outcomes

Ensure each piece of work is directly linked to a measurable outcome or business objective.

## Avoid Over-engineering

Build the simplest solution that meets the current requirements, avoiding the temptation to over-design or over-build.

## Direct Connection with Stakeholders

Maintain a close, collaborative relationship with stakeholders to ensure the work aligns with their needs.

# More Key Concepts



## Strong, Servant Leadership

Effective Scrum leaders empower their teams, remove obstacles, and foster a collaborative environment focused on continuous improvement.

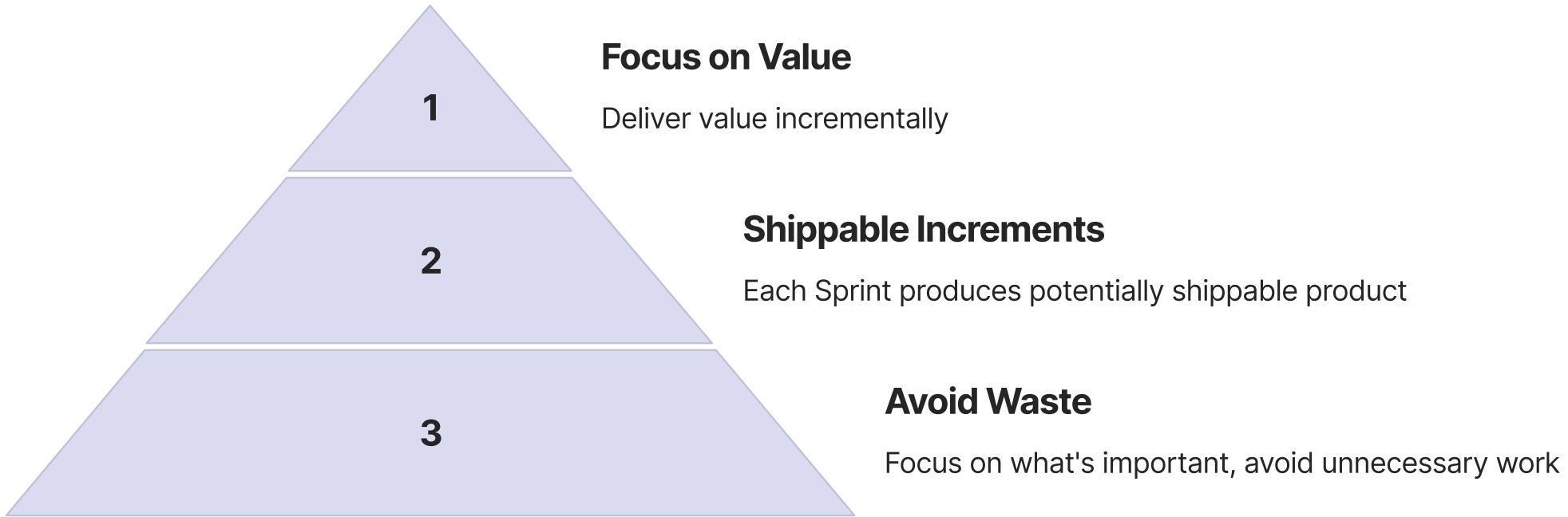
## Continuous Flow of Work

Lean Thinking in Scrum emphasizes a steady, uninterrupted flow of work to maximize efficiency and deliver value quickly.

## Empirical Process Control

Scrum relies on transparency, inspection, and adaptation to make informed decisions and continuously improve the process.

# Eliminate Unnecessary Steps



The key is to eliminate unnecessary steps and focus on delivering value to customers. Each Sprint should produce a potentially shippable product increment, avoiding any work that doesn't directly contribute to that goal.



# Tie Work to Specific Outcomes

- 1**
- 2**
- 3**

## **Embodied in the Sprint Goal**

Every Sprint has a clear, specific objective that the team works towards.

## **Focused Effort**

By tying work to a specific outcome, the team can stay focused and aligned on the most important priorities.

## **Measurable Progress**

Linking work to outcomes makes it easier to track and measure the team's progress throughout the Sprint.

# Avoid Over-engineering

1

**Minimum Viable Product**

2

**Iterate on Feedback**

3

**Prevent Waste**

Develop a minimum viable product (MVP) to get the core functionality out quickly. Iterate on the product based on customer feedback, rather than wasting time on potentially unneeded features. This helps avoid over-engineering and ensures the focus remains on delivering value.



# Direct Connection with Stakeholders

1

## Sprint Review

The Sprint Review provides a regular opportunity to get feedback from stakeholders on the product.

2

## Stakeholder Needs

Maintaining a direct connection with stakeholders ensures the product meets their needs and expectations.

# Strong, Servant Leadership

1

## Scrum Master Role

The Scrum Master embodies the principle of strong, servant leadership.

2

## Serving the Team

The Scrum Master focuses on serving the team and removing impediments to their progress.

3

## Enabling Success

By leading with a servant mindset, the Scrum Master empowers the team to achieve their goals.





# Continuous Flow of Work

## Iterative Nature

Scrum's iterative approach ensures the team focuses on the most valuable items, minimizing idle time.

## Responsive to Change

The Scrum framework's flexibility allows teams to quickly adapt to changing priorities and requirements.

1

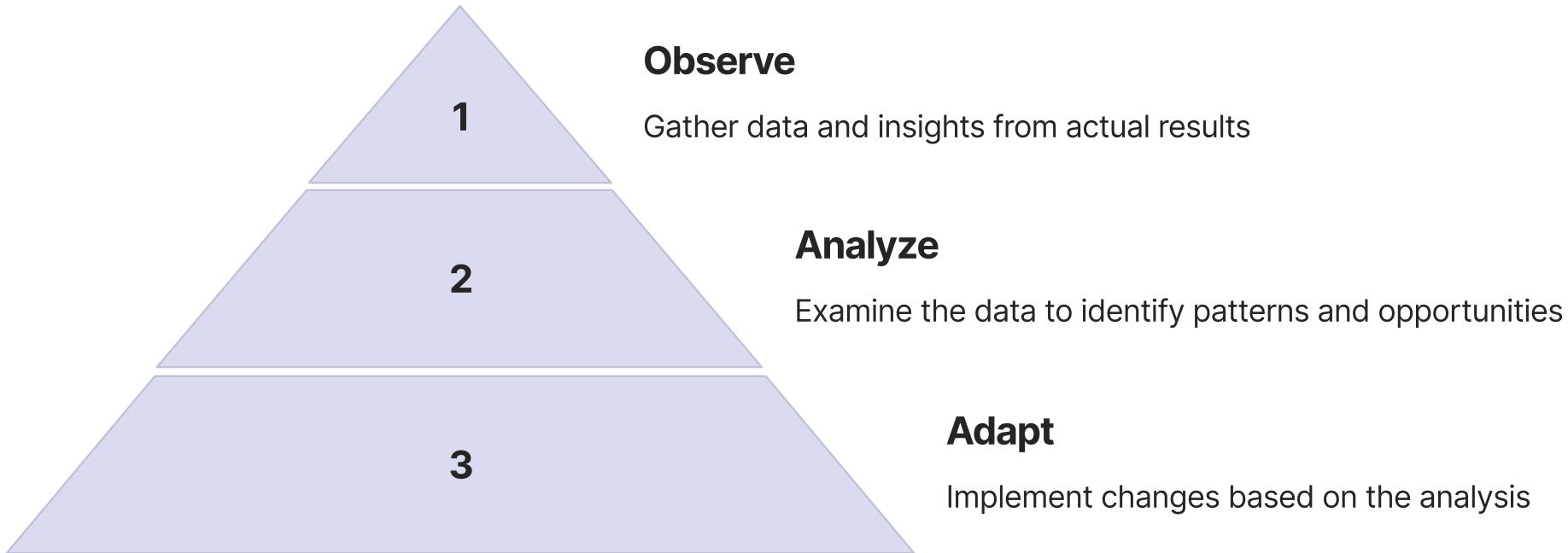
2

3

## Continuous Delivery

By delivering work in small, frequent increments, Scrum enables a continuous flow of value to stakeholders.

# Empirical Process Control



Empirical process control in Scrum relies on making improvements based on actual observations and results, not just theories or assumptions. This iterative cycle of observing, analyzing, and adapting helps the team continuously enhance their processes and deliver better outcomes.

# Practical Example: E-commerce Platform

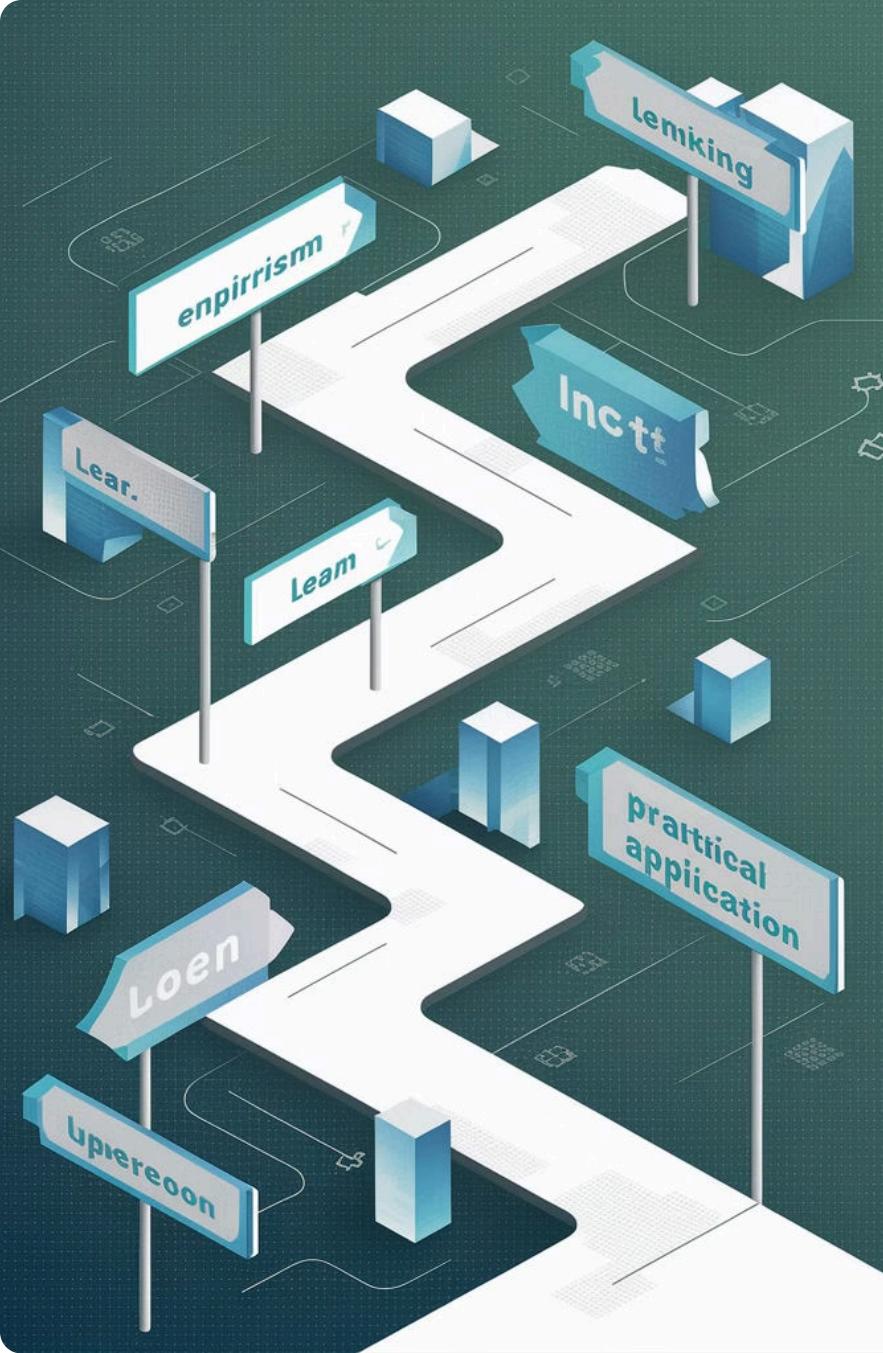
- 1 Identify core functionality
- 2 Build MVP with these features
- 3 Release to small user group
- 4 Gather feedback
- 5 Use feedback to guide next Sprint's priorities

By focusing on the core functionality, building a minimum viable product, and gathering feedback from a small user group, the team can iteratively improve the e-commerce platform based on real user needs. This lean approach helps avoid over-engineering and ensures the product remains aligned with customer requirements.



# Reflection

Reflecting on past projects, I recall instances where eliminating unnecessary steps could have improved efficiency. Applying lean thinking principles to identify and remove waste would have streamlined workflows and enhanced overall project delivery.



# Next Steps

## Explore Empiricism

We'll dive deeper into how empiricism, the foundation of Scrum, works together with lean thinking principles.

## Understand Lean Concepts

Key lean concepts like continuous flow, eliminating waste, and direct stakeholder feedback will be examined in the Scrum context.

## Apply in Practice

We'll discuss practical ways to apply these ideas to improve your Scrum implementation and deliver value more efficiently.



# Remember

## Maximize Value

In Scrum, we're always looking to maximize the value we deliver to our customers and stakeholders.

## Minimize Waste

At the same time, we strive to eliminate any unnecessary steps or activities that don't contribute to that value.

## Continuous Improvement

It's a never-ending process of identifying and removing waste, while continuously improving our processes and practices.

# Thank You



## For your attention

Thank you all for your time and engagement during this presentation on Lean Thinking in Scrum.



## Looking forward to continuing our exploration

We look forward to continuing our exploration of how Lean Thinking can enhance your Scrum practices.

# Empiricism and Lean Thinking in Scrum

Explore the principles of empiricism and lean thinking that underpin the Scrum framework. Understand how these approaches drive continuous improvement and value delivery in agile software development.

by Mayko Silva



# Quick Recap



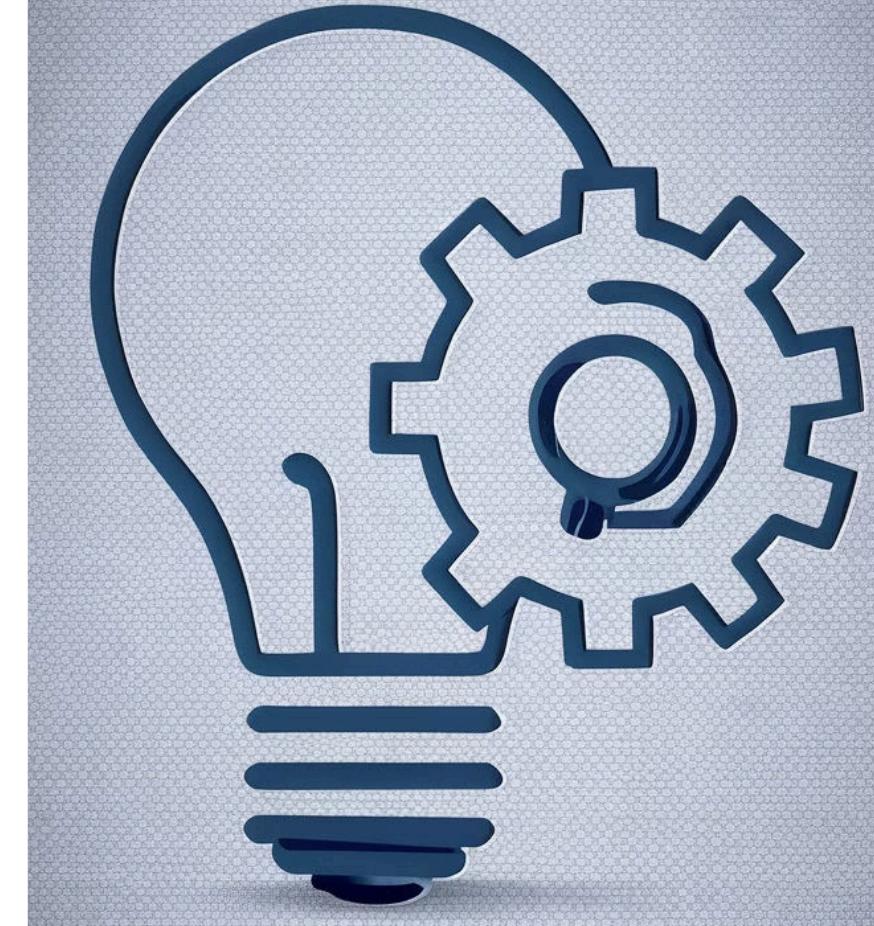
## Empiricism

Making decisions based on observation and experience.



## Lean Thinking

Maximizing value while minimizing waste.



# Scrum Events: Sprint Review



## Inspect Product Increment

The Sprint Review embodies empiricism by inspecting the actual product increment developed during the Sprint.

## Gather Stakeholder Feedback

The Sprint Review leverages lean thinking by using stakeholder feedback to guide the next steps for the product.

## Inform Next Steps

Insights from the Sprint Review help the Scrum Team determine how to best proceed with the product development.

# Scrum Concepts: Sprint Goal

## Ties Work to Outcome

The Sprint Goal in Scrum ties the work being done to a specific, measurable outcome. This aligns with the principles of lean thinking.

## Sets Clear Target

The Sprint Goal provides a clear target for the Scrum team to focus on and observe empirically during the Sprint.



# Scrum Roles: Scrum Master

## 1 Removes Impediments

The Scrum Master works to identify and remove any obstacles or impediments that are hindering the team's progress.

## 2 Improves Efficiency

By applying lean thinking principles, the Scrum Master helps the team optimize their processes and work more efficiently.

## 3 Encourages Inspection and Adaptation

The Scrum Master fosters a culture of empiricism, where the team regularly inspects their ways of working and adapts them as needed.

# Development Approach: Incremental Development

## Frequent Delivery

Incremental development delivers working increments frequently, avoiding the waste of unused features.

## Lean Thinking

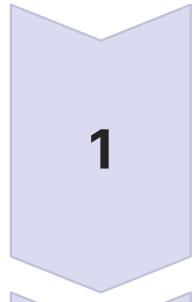
By focusing on delivering only what is needed, incremental development aligns with lean thinking principles.

## Empirical Feedback

Incremental development gets real-world feedback quickly, enabling teams to adapt and improve based on empirical data.

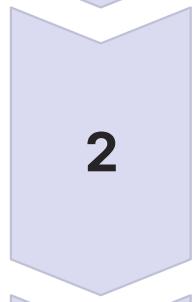


# Practical Example: Mobile App Development



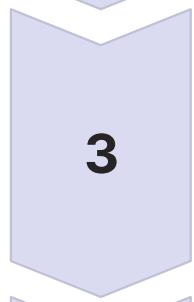
## 1 Plan Features

Plan a small set of features for a two-week Sprint, using lean thinking to focus on the essentials.



## 2 Build and Release

Build the features and release the app to a test group, applying empiricism to gather user feedback.



## 3 Gather Feedback

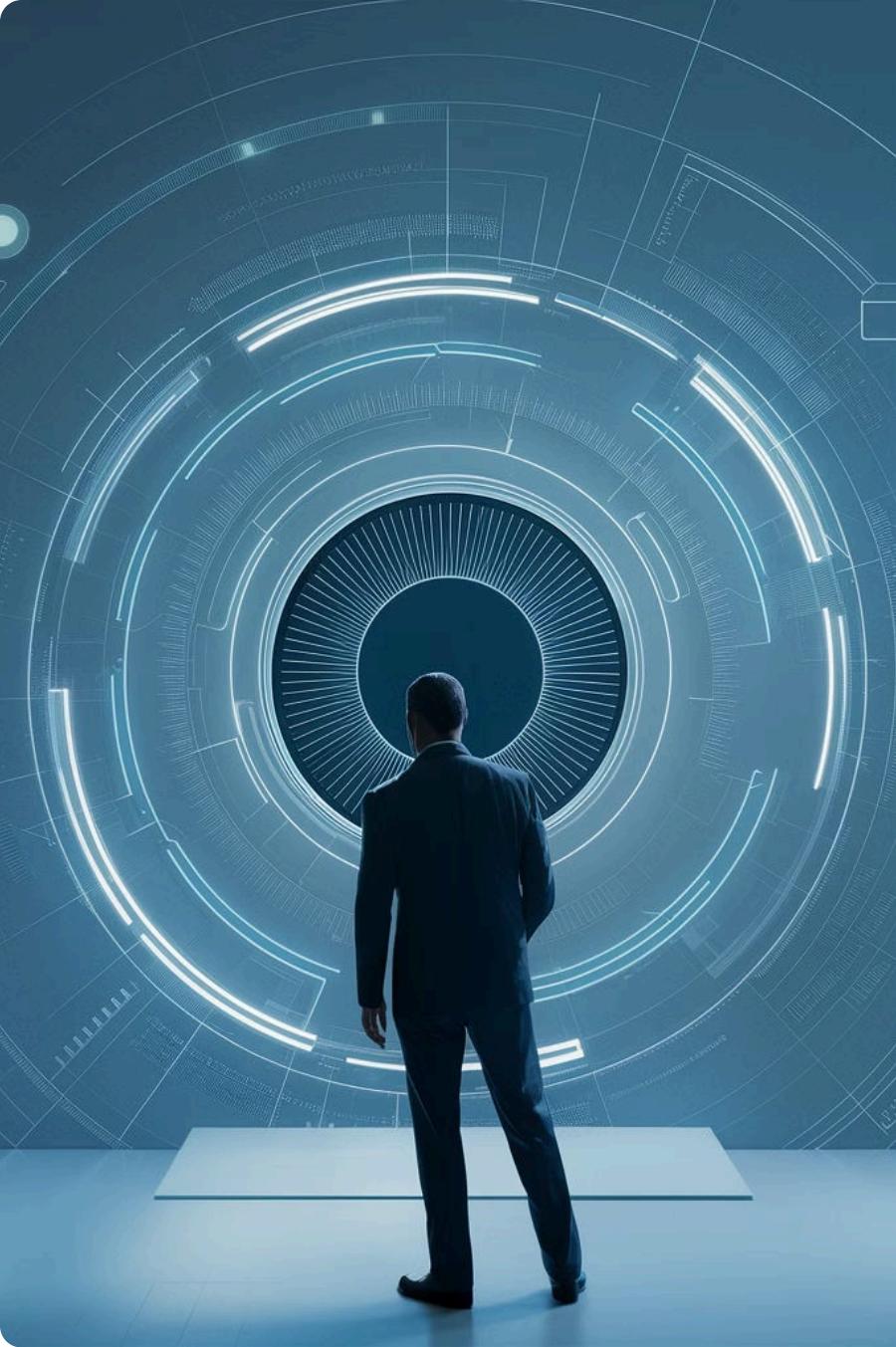
In the Sprint Review, collect user feedback to understand how the app is performing.



## 4 Adjust Plans

Based on the feedback, adjust the plans for the next Sprint, using lean thinking to iterate and improve.





## Understanding Check

**Which best represents empiricism and lean thinking in Scrum?**

- A) Detailed 6-month development plan
- B) Daily stand-up meetings
- C) Waiting for all features before release
- D) Using customer feedback for Product Backlog prioritization



# Answer and Explanation



## Correct Answers

B and D are the correct answers.



## Explanation

B and D align with the principles of empiricism and lean thinking in Scrum, while A and C go against these principles.

# Reflection

## Apply in Current Projects

Look for ways to make your work more empirical and lean, such as implementing rapid prototyping, gathering user feedback, and continuously improving your processes.

## Embrace Lean Thinking

Challenge assumptions, focus on value delivery, and eliminate waste. Experiment and learn quickly to optimize your approach.

## Continuous Improvement

Regularly reflect on your work and identify opportunities to apply the principles of empiricism and lean thinking to drive ongoing progress.



# Next Steps



## Predictability and Risk

We'll explore how Scrum helps manage predictability and risk in software development projects.

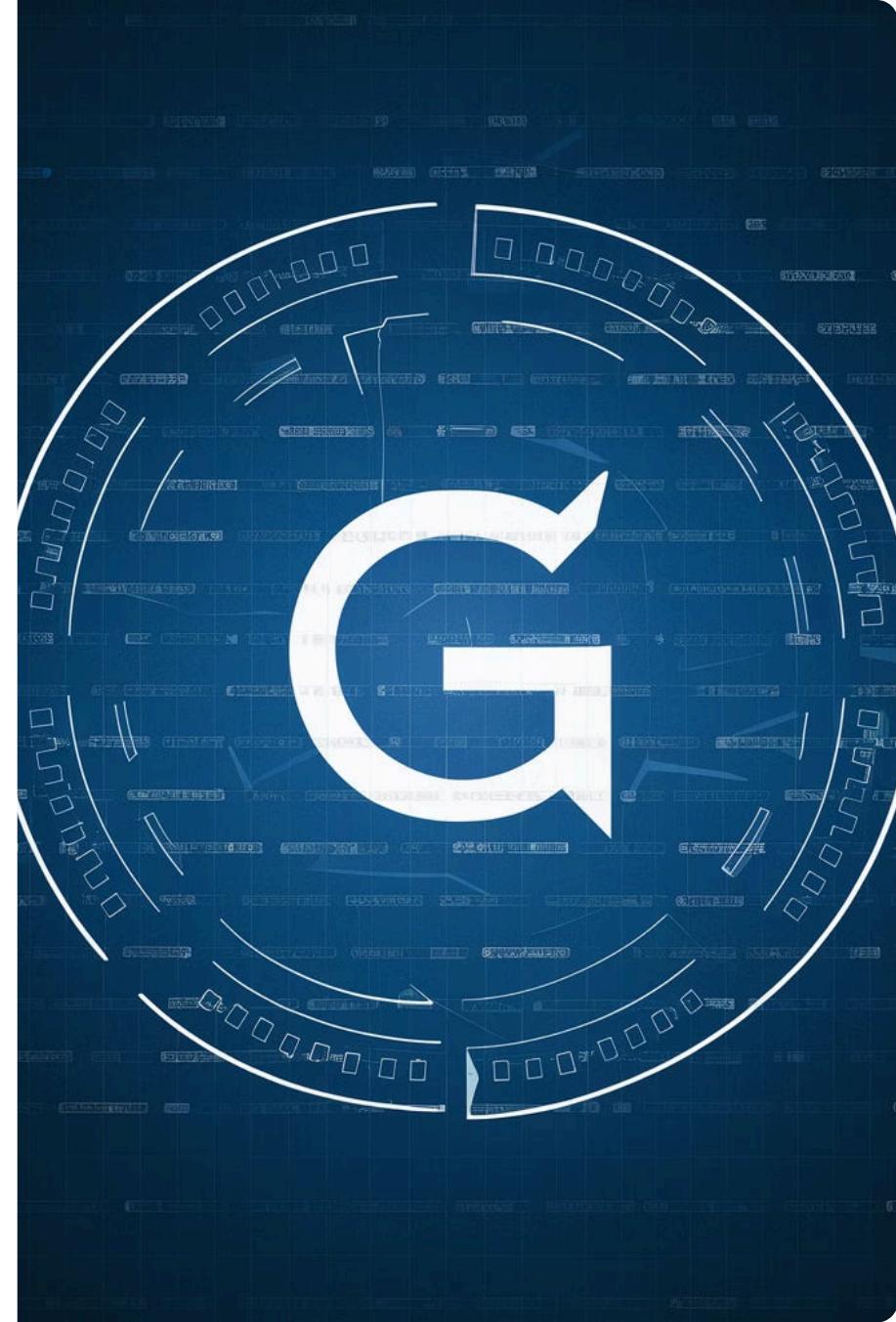
## Scrum Roadmap

Understanding the Scrum framework's approach to planning

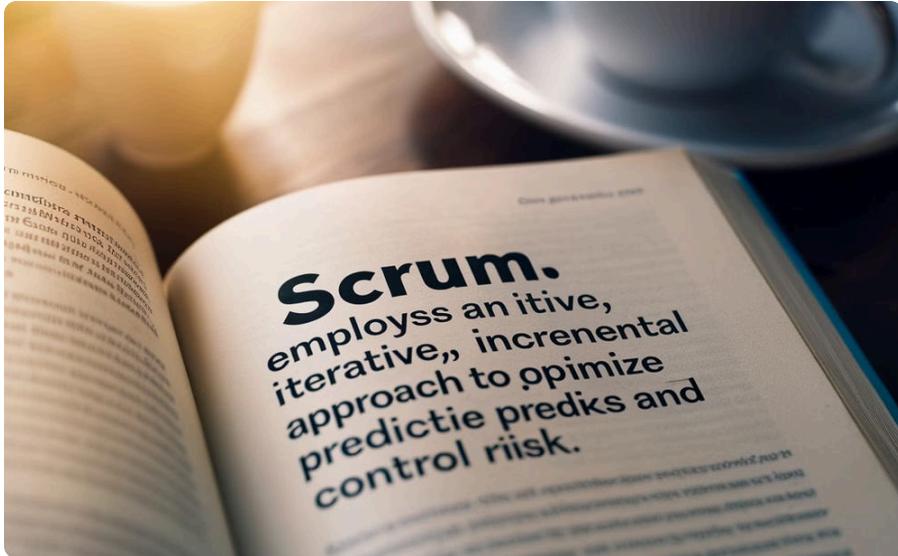
# Predictability and Risk in Scrum

Explore the balance between predictability and risk in the Scrum framework.  
Understand how to manage uncertainty and adapt to change while  
maintaining project visibility.

by Mayko Silva



# Key Quote from Scrum Guide



## Iterative and Incremental

The Scrum Guide emphasizes that Scrum uses an iterative and incremental approach to improve predictability and manage risk.



# Iterative Approach in Scrum

1

## Product Backlog Prioritization

The Product Owner prioritizes the Product Backlog, determining the order in which features will be developed.

2

## Sprint Planning

The Scrum Team selects items from the prioritized Product Backlog to include in the current Sprint.

3

## Incremental Development

The Scrum Team works to develop a valuable Increment of the product during the Sprint.

4

## Inspection and Adaptation

The Scrum Team and stakeholders inspect the results and make adjustments as needed.



## **Benefits of Iterative Approach**



## **Regular Inspection**

The iterative approach allows for regular inspection and adaptation, ensuring the project stays on track.



## Risk Management

Iterative development is key to managing risk, as issues can be identified and addressed early on.



# Improved Predictability

The iterative approach improves predictability, as progress can be measured and adjustments made accordingly.

# **Incremental Approach in Scrum**

Scrum focuses on delivering small, valuable increments of work. This incremental approach allows teams to gather feedback, learn, and adapt quickly, rather than waiting for a large, monolithic project to be completed.





# Benefits of Incremental Approach

## Early Stakeholder Feedback

The incremental approach allows for early and frequent feedback from stakeholders, enabling course corrections and ensuring the product meets their needs.

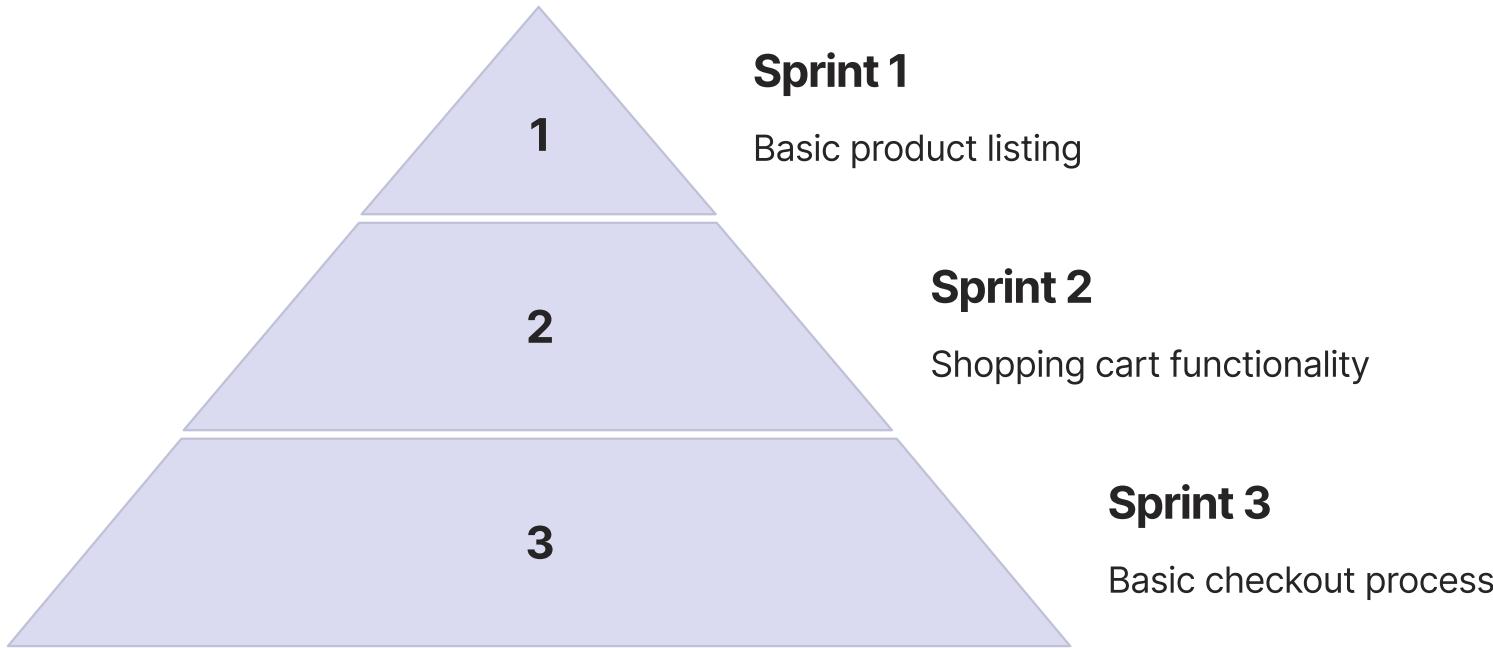
## Reduced Risk

By breaking the work into manageable chunks, the incremental approach reduces the risk of building the wrong thing.

## Improved Predictability

The incremental approach improves predictability by dividing the work into smaller, more manageable pieces.

# Practical Example: E-commerce Platform



Instead of a 6-month build, the e-commerce platform is developed incrementally through 3 sprints. Feedback is gathered after each sprint to adjust the plans, allowing for a more responsive and user-centric approach.



# How This Approach Helps

## Reduces Risk

Quick detection if heading wrong direction. Allows for timely adjustments.

## Improves Predictability

Regular delivery of working increments. Better estimation of future feature timelines.

# **Understanding Check**

**Which best represents how Scrum manages predictability and risk?**

- A) Detailed upfront project plan
- B) Regular delivery of working increments
- C) Fixed deadlines for all features at project start
- D) Stakeholder feedback only after all features complete

# **Answer and Explanation**

## **Correct Answer: B**

Regular increments allow frequent inspection and adaptation, which is key to managing risk and improving predictability in Scrum.

## **Explanation**

Options A, C, and D represent less agile, traditional approaches that do not provide the same benefits as the iterative and incremental approach of Scrum.

# Reflection

## Apply Iterative Approach

Consider how you can break down your current projects into smaller, valuable increments that can be delivered iteratively.

## Identify Increments

Analyze your work and see if you can identify opportunities to deliver value in a more incremental way.

## Embrace Change

Remember that the iterative and incremental approach allows you to adapt to changing requirements and priorities.



# Next Steps

We'll explore the incremental nature of Scrum in more depth, examining how it helps teams deliver value iteratively and adapt to changing requirements.





# Remember

## 1 Manage Risk and Improve Predictability

Through short work cycles, frequent value delivery, and adaptation based on feedback.

# Thank You

For your attention. Looking forward to continuing our exploration of Scrum!



# The Incremental Nature of Scrum

Scrum is an agile framework that emphasizes the incremental and iterative nature of software development. It focuses on delivering value in small, manageable chunks rather than a single, monolithic release.

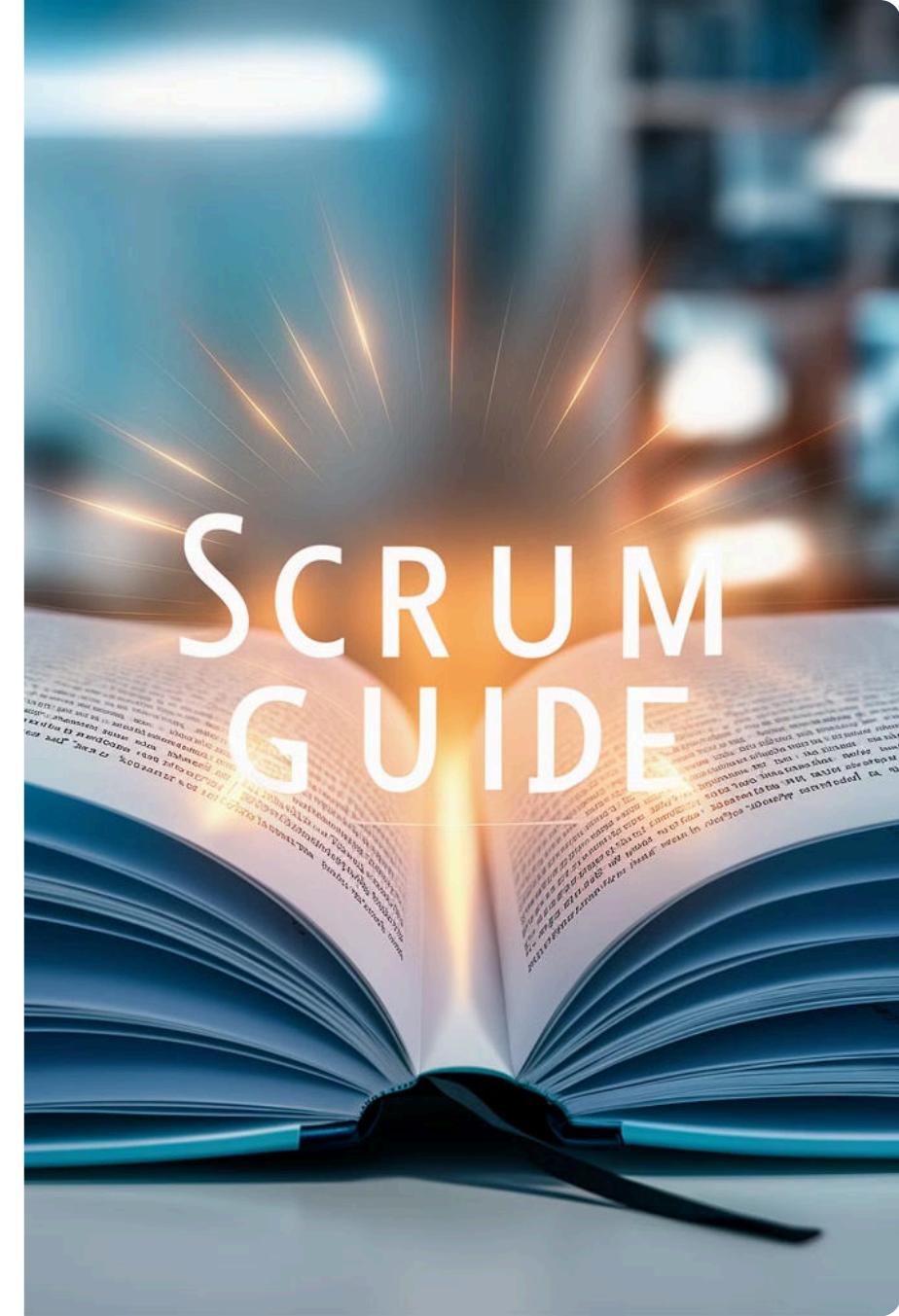
— by Mayko Silva



# Key Quote from Scrum Guide

 **"The Scrum Team turns a selection of the work into an Increment of value during a Sprint."**

This quote highlights the incremental nature of Scrum, where the team delivers value in small, manageable chunks rather than a single, large deliverable.



# What Does This Mean?



## Incremental Development

Build small, valuable pieces over time, with each piece an Increment.

## Building Value

Each Increment adds value to the product, focusing on creating something useful and potentially releasable.

## Continuous Progress

Increments build upon each other Sprint by Sprint, gradually forming the complete product.



# Benefits of Incremental Approach

## Early Feedback

Frequent delivery allows regular stakeholder input.

## Flexibility

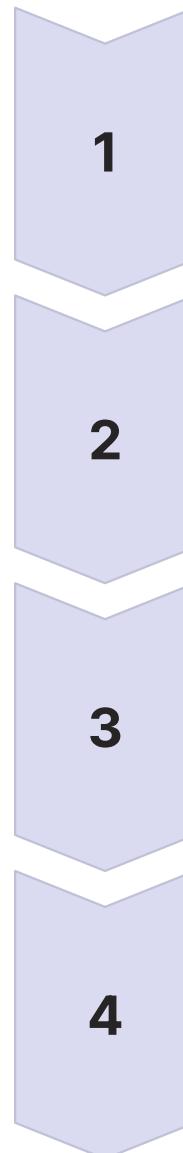
Quick adaptation to changing market conditions or customer needs.

## Risk Reduction

Constant delivery and validation helps identify issues early.



# Practical Example: Social Media App



## Sprint 1

Basic user registration and login functionality.

## Sprint 2

Create and view user profiles.

## Sprint 3

Post status updates feature.

## Sprint 4

Friend/follow other users.



# **Understanding Check**

**Which best represents the incremental nature of Scrum?**

- A) Planning entire project upfront, executing in phases
- B) Delivering complete, fully-featured product at project end
- C) Building and delivering small, valuable pieces each Sprint
- D) Focusing on completing as many tasks as possible per Sprint

# **Answer and Explanation**

## **Correct answer: C**

Building and delivering small, valuable pieces each Sprint is the essence of Scrum's incremental nature.

## **Incorrect answers:**

A and B represent waterfall-style approaches, while D focuses on tasks rather than delivering value.

# Reflection

Consider how you can apply this incremental approach to your current projects. Can you break down your work into smaller, valuable increments that can be delivered and tested regularly? This can help you stay agile and responsive to changing needs.

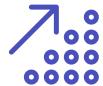


# Next Steps



## Cross-Functional Teams

We'll explore how Scrum leverages cross-functional teams to drive incremental progress.



## Continuous Improvement

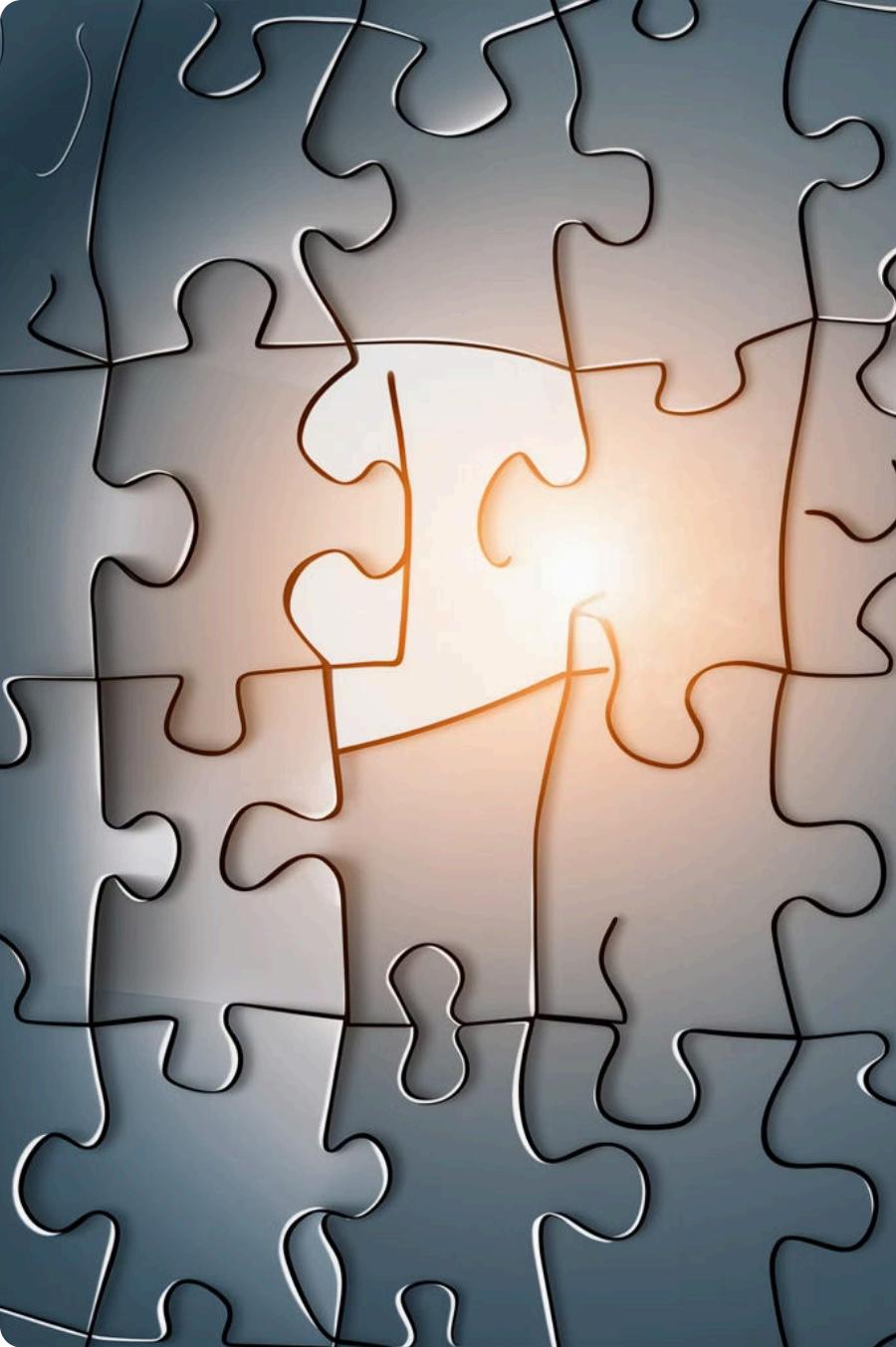
The Scrum framework emphasizes continuous learning and adaptation to drive ongoing improvement.



## Iterative Sprints

We'll dive deeper into the Scrum sprint cycle and how it enables rapid, incremental delivery.





# Remember

## **Build Incrementally**

Construct products in small, manageable pieces rather than all at once.

## **Deliver Frequently**

Provide customers with new value on a regular basis, not just at the end.

## **Adapt to Feedback**

Continuously improve based on user input and changing requirements.

# Thank You

Thank you for your attention. We look forward to continuing our exploration of Scrum!



# Cross-Functional Teams in Scrum

Scrum emphasizes the use of cross-functional teams to deliver software incrementally. These teams bring together diverse skills and perspectives to tackle complex challenges efficiently.

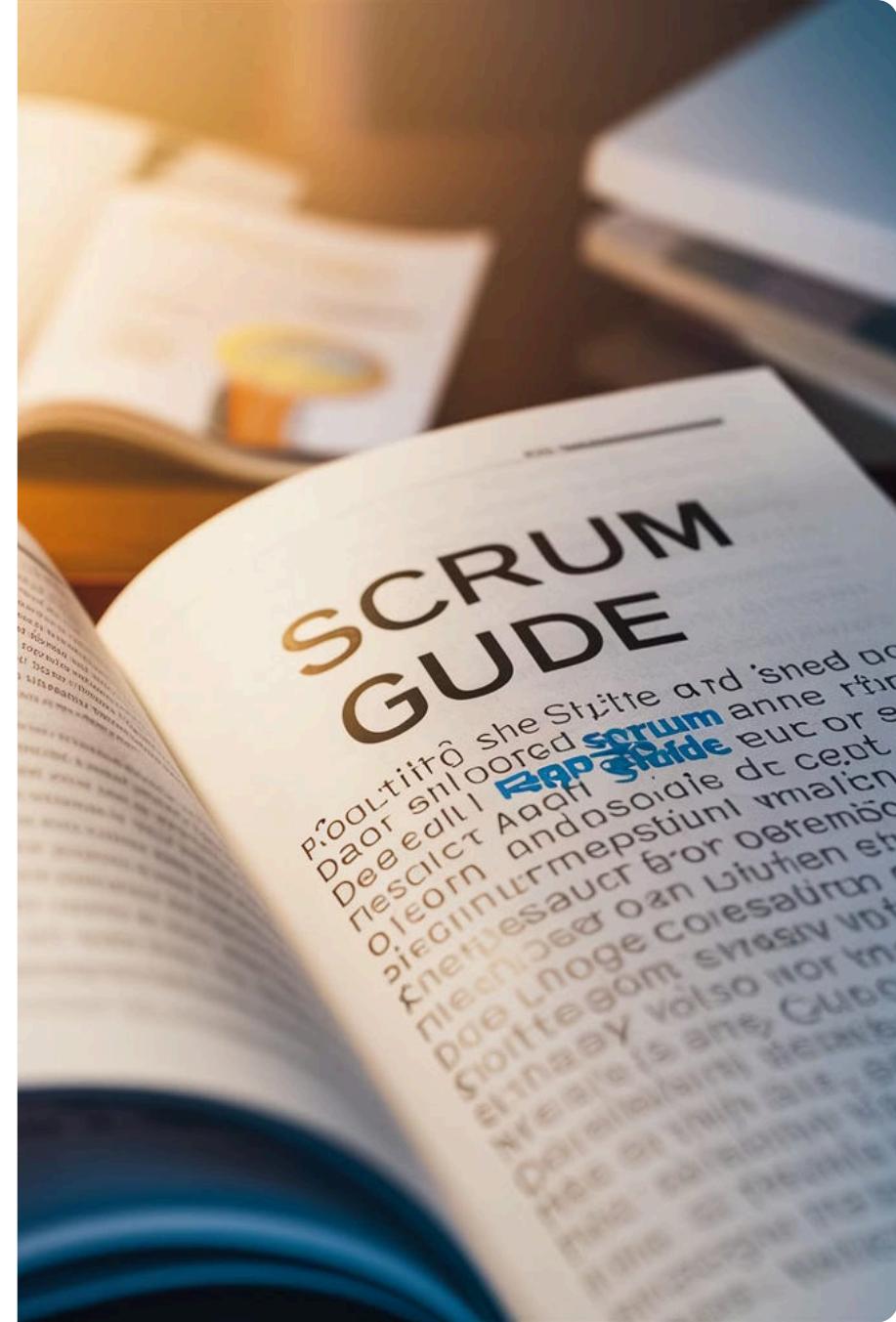
by Mayko Silva



# Key Quote from Scrum Guide

"Scrum engages groups of people who collectively have all the skills and expertise to do the work and share or acquire such skills as needed."

This quote highlights the importance of cross-functional teams in Scrum, where team members possess diverse skills to complete the work.





# What Does This Mean in Practice?

- 1
- 2
- 3

## All-encompassing Skills

The team has all the necessary skills for the project, including developers, testers, designers, and other specialists.

## Skill Sharing

Team members actively share their skills and knowledge with one another.

## Skill Acquisition

The team acquires new skills as needed, rather than outsourcing work.

# Benefits of Cross-Functional Teams

## Efficiency

Cross-functional teams can handle all aspects of a project without external dependencies, improving efficiency and reducing delays.

## Collaboration

Cross-functional teams encourage teamwork and mutual learning, fostering a collaborative environment.

## Flexibility

Cross-functional teams can more easily adapt to changing project needs, making them more flexible and responsive.

# Practical Example: Mobile App Development



## iOS and Android Developers

The Scrum team includes developers for both iOS and Android platforms to ensure the app works seamlessly across devices.



## UI/UX Designers

Designers collaborate with the team to create an intuitive and visually appealing user experience for the mobile app.



## Quality Assurance

QA specialists ensure the mobile app meets high standards of functionality, usability, and performance.



## Technical Writer

A technical writer provides clear and comprehensive documentation to support the mobile app users.

# **Understanding Check**

**Your team needs AI knowledge but lacks experience. What's the best action?**

- A) Outsource AI development
- B) Remove AI features from project
- C) Explain team can't implement AI
- D) Have team members learn about AI implementation





# Answer and Explanation

## In Scrum, teams acquire needed skills rather than outsourcing

This aligns with the cross-functional team principle, where team members develop a wide range of skills to work on all aspects of the project.

## Continuous Improvement

The ability to acquire new skills supports the Scrum value of continuous improvement, as teams can adapt and enhance their capabilities over time.



# Reflection Questions



## Apply Cross-Functionality

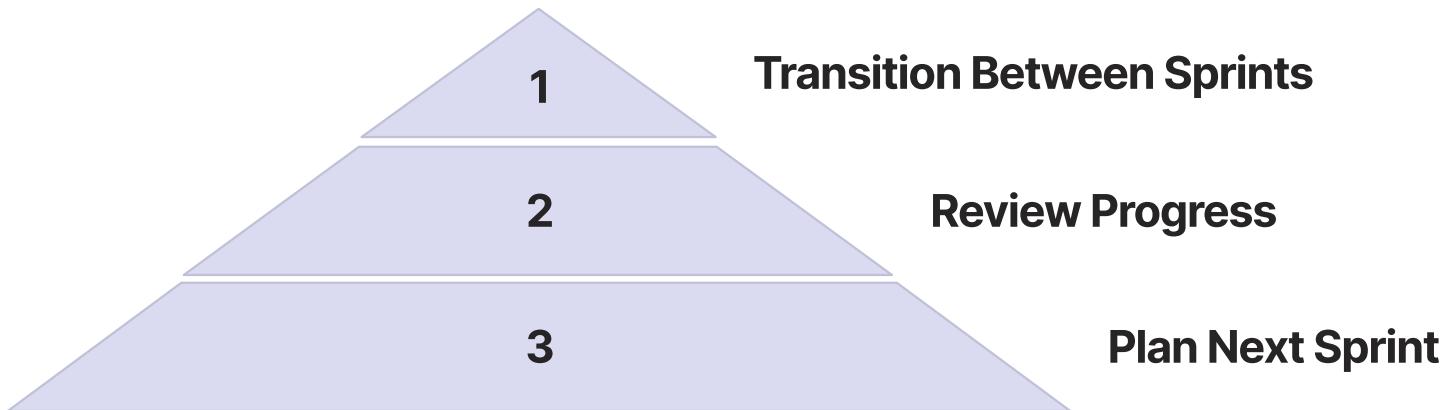
How can you apply cross-functionality in your current projects?



## Skill Acquisition

What skills could your team benefit from acquiring?

# Next Steps



We'll explore how Scrum Teams transition between Sprints, reviewing their progress and planning for the next Sprint to continuously improve their process and deliver value.



# Key Takeaways



## Build Cross-Functional Teams

Assemble teams with all the necessary skills to complete the project.



## Continuous Learning

Be ready to learn and adapt as needed throughout the project.



## Agility

Embrace an agile mindset to respond quickly to changing requirements.

# Thank You

Thank you for your attention. We look forward to continuing our exploration of Scrum!



# Moving from Sprint to Sprint

Navigating the agile development process with efficiency and adaptability.

— by Mayko Silva





## Key Principle



### Sprint Cycle

In Scrum, each project is divided into time-boxed Sprints. As soon as one Sprint ends, the next Sprint begins.



### Continuous Delivery

The goal is to deliver working software incrementally, with each Sprint building upon the previous one.



### Time-Boxed

Sprints have a fixed duration, typically 2-4 weeks, to maintain a consistent pace and rhythm.

# What Does This Mean in Practice?



## Continuous Flow

No buffer time between Sprints - new Sprint starts immediately after previous one ends.



## All-inclusive Sprints

All work, including testing, QA, and integration, happens within each Sprint.



## Seamless Transition

The team moves directly from one Sprint to the next without interruption.



# Benefits of This Approach

## 1 Momentum

Maintains the team's rhythm and productivity.

## 2 Focus

Keeps the team concentrated on delivering value continuously.

## 3 Adaptability

Allows for quick adjustments based on the previous Sprint's outcomes.



# Practical Example: Mobile App Development

1

**Friday, 4 PM**

Conclude Sprint 5 (Review and Retrospective).

2

**Monday, 9 AM**

Begin Sprint 6 (Sprint Planning).

3

**No Gap Between Sprints**

Seamless transition between Sprints 5 and 6.



# **Understanding Check**

## **When does a new Sprint begin??**

- A) After Sprint Planning is completed
- ]B) When the Product Owner declares the start in the project management tool
- C) When the Scrum Master announces the start of the Sprint
- D) Immediately after the previous Sprint ends



# Answer and Explanation

## Correct Answer: D

In Scrum, a new Sprint begins immediately after the previous Sprint ends, with no gap or buffer time between Sprints.

# Reflection

## Smooth Transitions

Ensure clear communication and alignment between Sprints to maintain momentum and avoid disruptions.

## Identifying Gaps

Review current practices to identify any unnecessary delays or disconnects between Sprints.

# Next Steps

## Explore Inspection

We'll dive deeper into the concept of inspection in Scrum, understanding how teams regularly review their progress and make adjustments.

## Understand Adaptation

The team will learn how to adapt their processes and plans based on the insights gained through inspection, ensuring continuous improvement.

# Remember



## Maintain Continuous Flow

Keep work moving seamlessly from one Sprint to the next, maintaining a steady pace of progress.



## Always Adapt and Improve

Be ready to adjust and enhance processes based on learnings, ensuring continuous optimization.

# Thank You

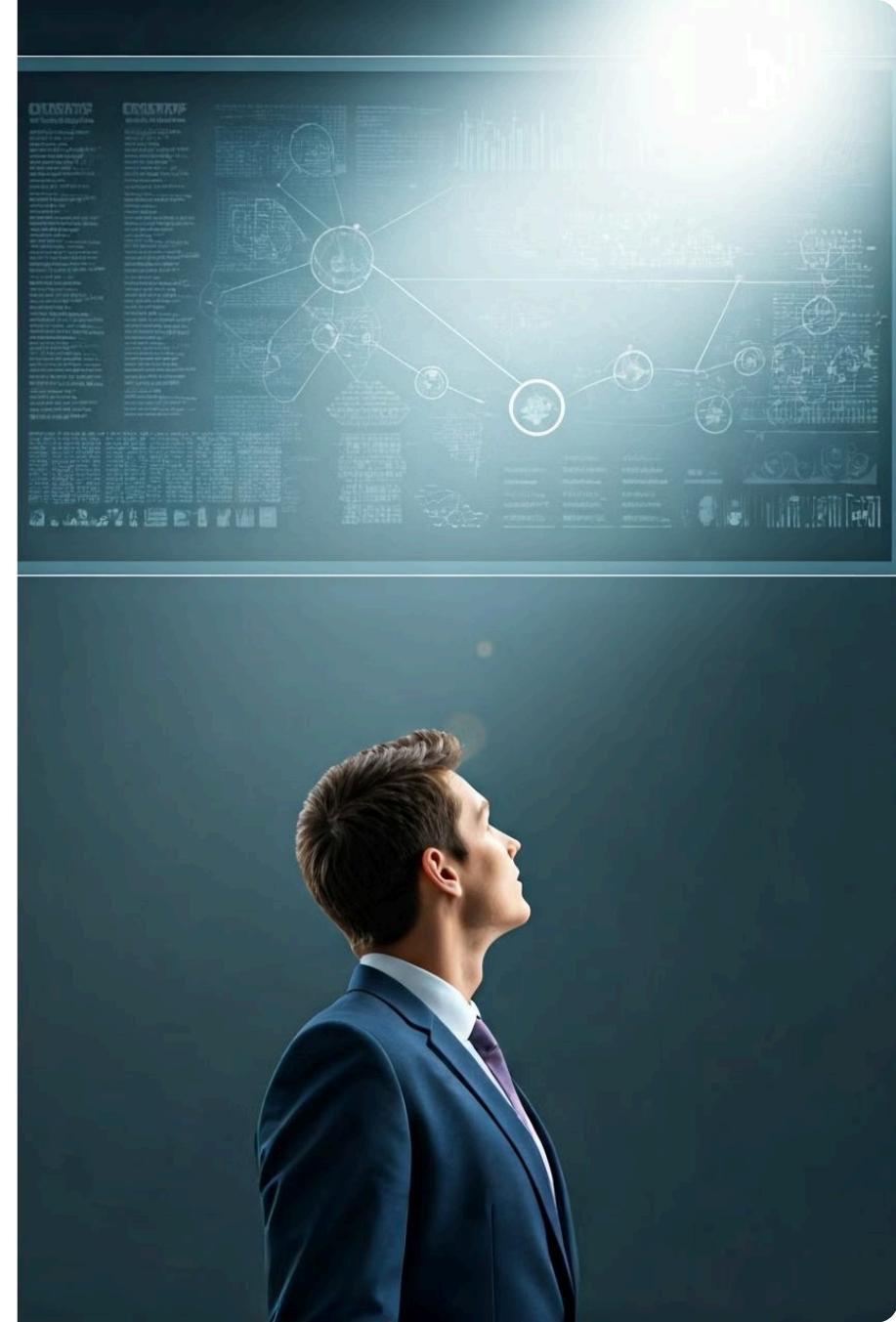
For your attention. Looking forward to continuing our exploration of Scrum!



# Forget Everything You Think You Know

Prepare to challenge your preconceptions and open your mind to new possibilities. This presentation will guide you through a transformative journey of discovery.

— by Mayko Silva



# Key Principle

## Forget Everything

To pass the Scrum Master certification exam, you must first forget everything you think you know about Scrum.

## Focus on Core Concepts

Instead, focus on the core concepts contained within the Scrum Guide.

## Understand Fundamentals

Mastering the fundamentals of Scrum is key to passing the certification exam.





## What Does This Mean in Practice?



### Focus on the Scrum Guide

The exam is based solely on the official Scrum Guide content, not on common practices or interpretations.



### Avoid Distractions

Concepts like user stories, story points, and the Agile Manifesto are not part of the official Scrum framework.



### Stick to the Basics

The exam tests your understanding of Scrum as defined in the Guide, not additional methodologies or practices.



# Benefits of This Approach



## Clarity

Helps focus on what's truly important for the exam.



## Accuracy

Ensures answers align with the official Scrum framework.



## Efficiency

Saves time by avoiding unnecessary topics.



# Practical Example: Work Estimation

1

# Temptation: Discuss story points or planning poker

It can be tempting to dive into discussions about story points or planning poker, as these are common practices in Agile development.

2

## **Reality: These aren't in Scrum Guide**

However, the Scrum Guide does not actually mention story points or planning poker as required elements of Scrum.

3

# Focus instead on Guide's content

Instead, the focus should be on the content of the Scrum Guide, which states that developers are responsible for all estimates.



# Understanding Check

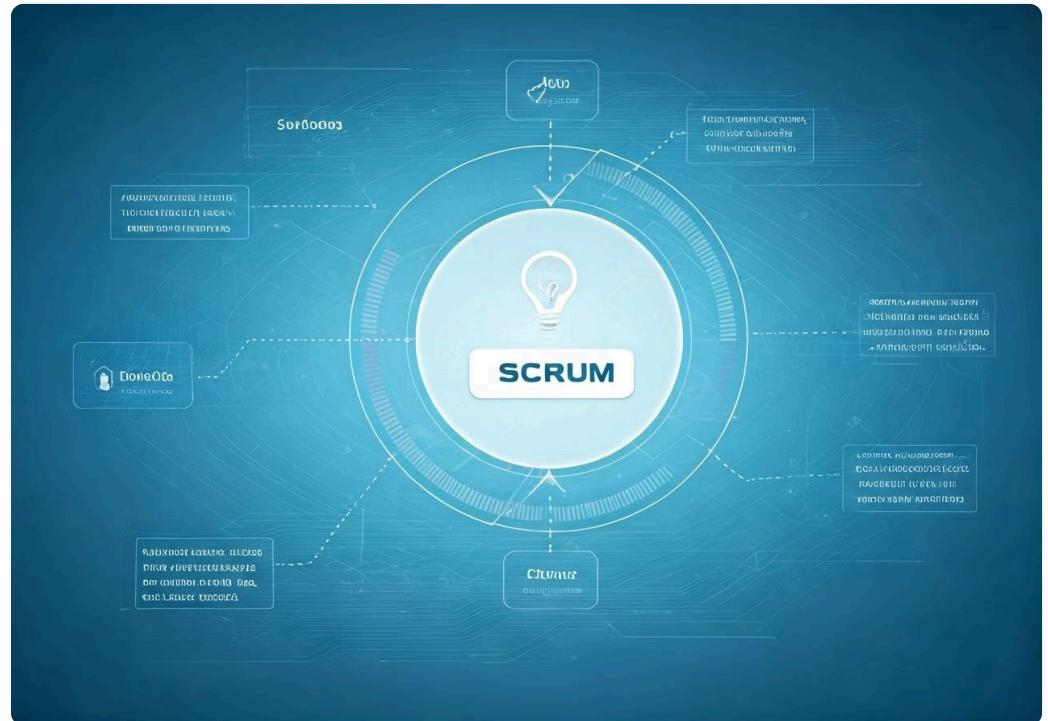
**Which of the following is NOT a concept directly addressed in the Scrum Guide??**

- A) The Sprint Goal
- B) The Product Backlog
- C) User Stories
- D) The Definition of Done

# Answer and Explanation

## Correct answer: C

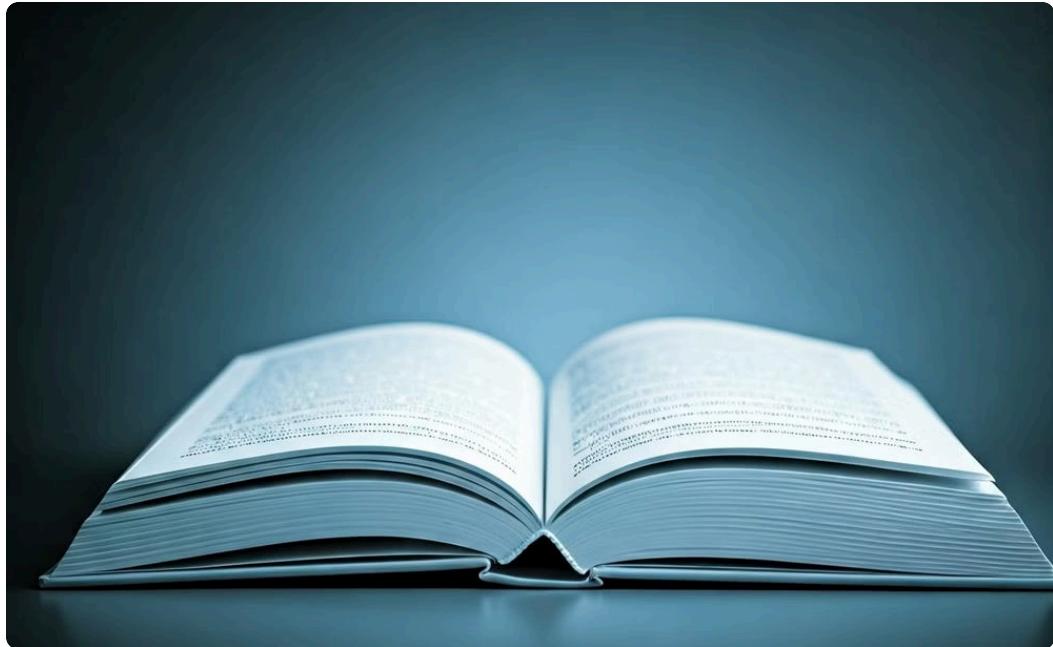
User Stories not mentioned in Scrum Guide, but a common practice in many Scrum teams. Not part of the official Scrum framework.



# Reflection

Consider any Scrum-related concepts you've assumed are part of the official framework. Examine how these assumptions might impact your exam preparation. Identifying and addressing any misconceptions will help you better understand the true Scrum principles.

# Remember



## Focus on Scrum Guide

Concentrate on the official Scrum Guide as the primary source of information for the certification exam.



## Set Aside Preconceptions

Avoid relying on ideas not explicitly stated in the Scrum Guide when preparing for the exam.

# Thank You

For your attention, and looking forward to continuing our exploration of Scrum!

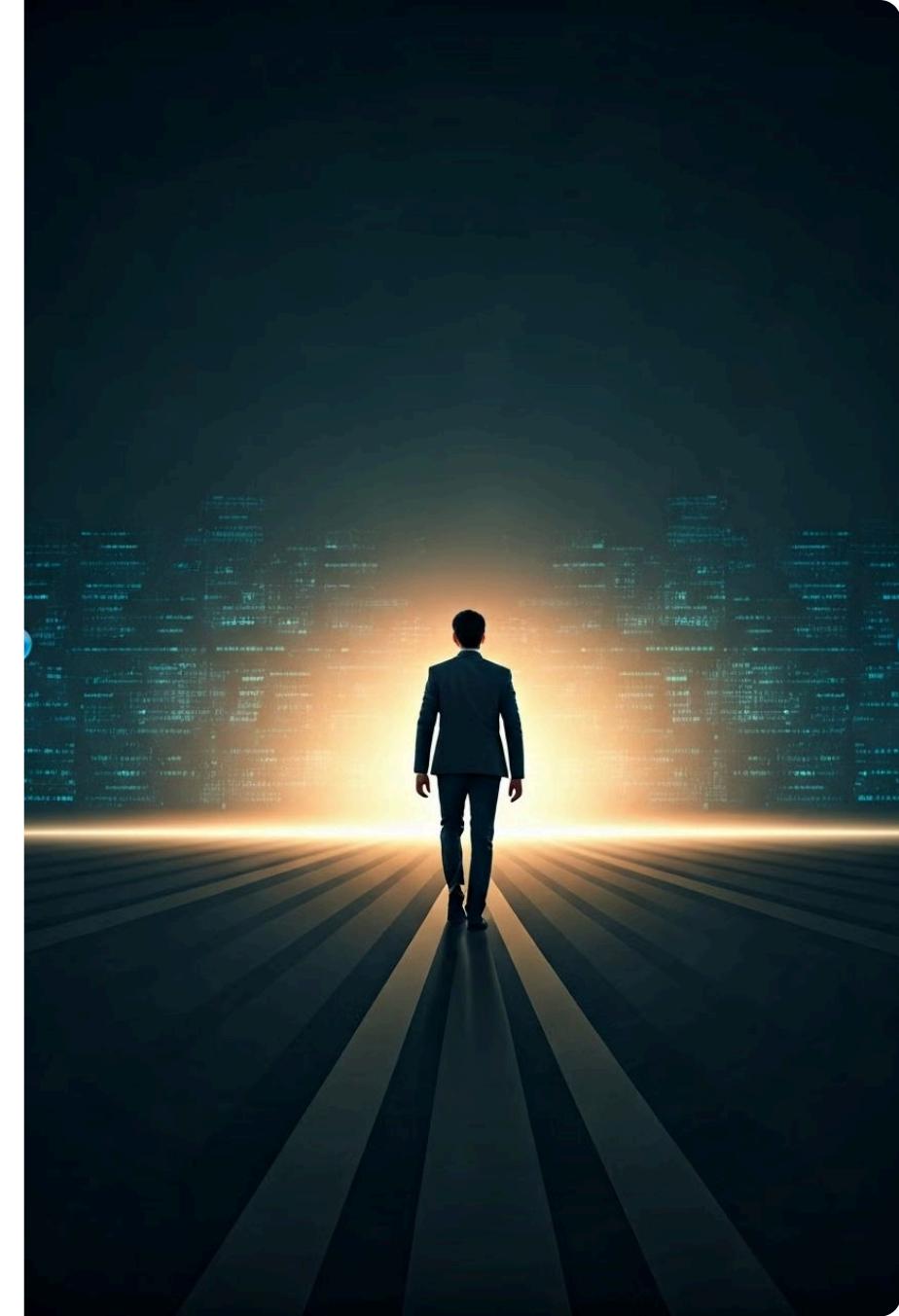
A faint, light-blue-toned photograph of a landscape featuring a suspension bridge, possibly the Golden Gate Bridge, spanning across a body of water under a cloudy sky.

*Thank  
you!*

# Keep the Focus on the Goal

Stay determined and keep your eye on the prize. Maintain a clear vision of what you want to achieve and let that guide your actions.

— by Mayko Silva



# Key Principle

## Focus

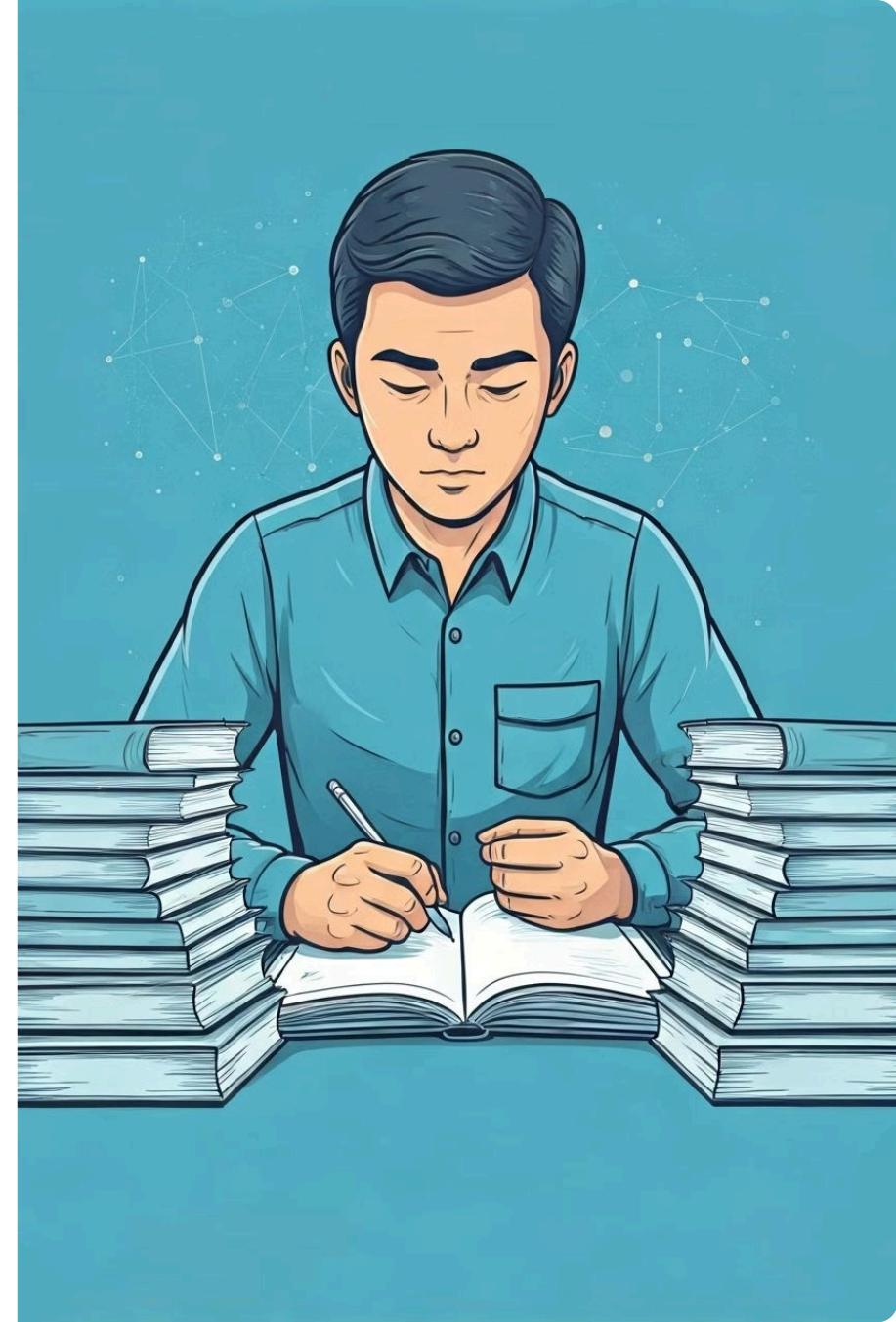
Maintain a laser-like focus on the topics that will be heavily tested on the exam. Avoid getting sidetracked by less relevant information.

## Efficiency

Maximize your study time by constantly reinforcing the most important concepts, rather than wasting time on peripheral topics.

## Relevance

Ensure that your preparation is tightly aligned with the exam requirements, so you can perform at your best.



# What Does This Mean in Practice?

1

## Prioritize Official Scrum

Focus on Scrum Guide and official concepts

2

## Avoid Distractions

Don't get sidetracked by non-official topics

3

## Repetition of Key Concepts

Consistently revisit and reinforce important Scrum

In practice, this means staying laser-focused on the core Scrum framework as defined in the official Scrum Guide. Avoid getting distracted by peripheral topics and instead repeatedly emphasize the most critical Scrum concepts to ensure deep understanding.

# Benefits of This Approach



## Efficiency

Saves time by focusing on exam-relevant content.



## Clarity

Avoids confusion between official concepts and common practices.



## Confidence

Repeated exposure builds understanding of key topics.



# **Practical Example: Study Focus**

## **Avoid Distractions**

Steer clear of spending time on "story points" or "velocity" - these aren't even mentioned in the Scrum Guide.

## **Focus on Essentials**

Instead, concentrate on the official Scrum concepts like the Sprint Goal and Definition of Done.

# **Understanding Check**

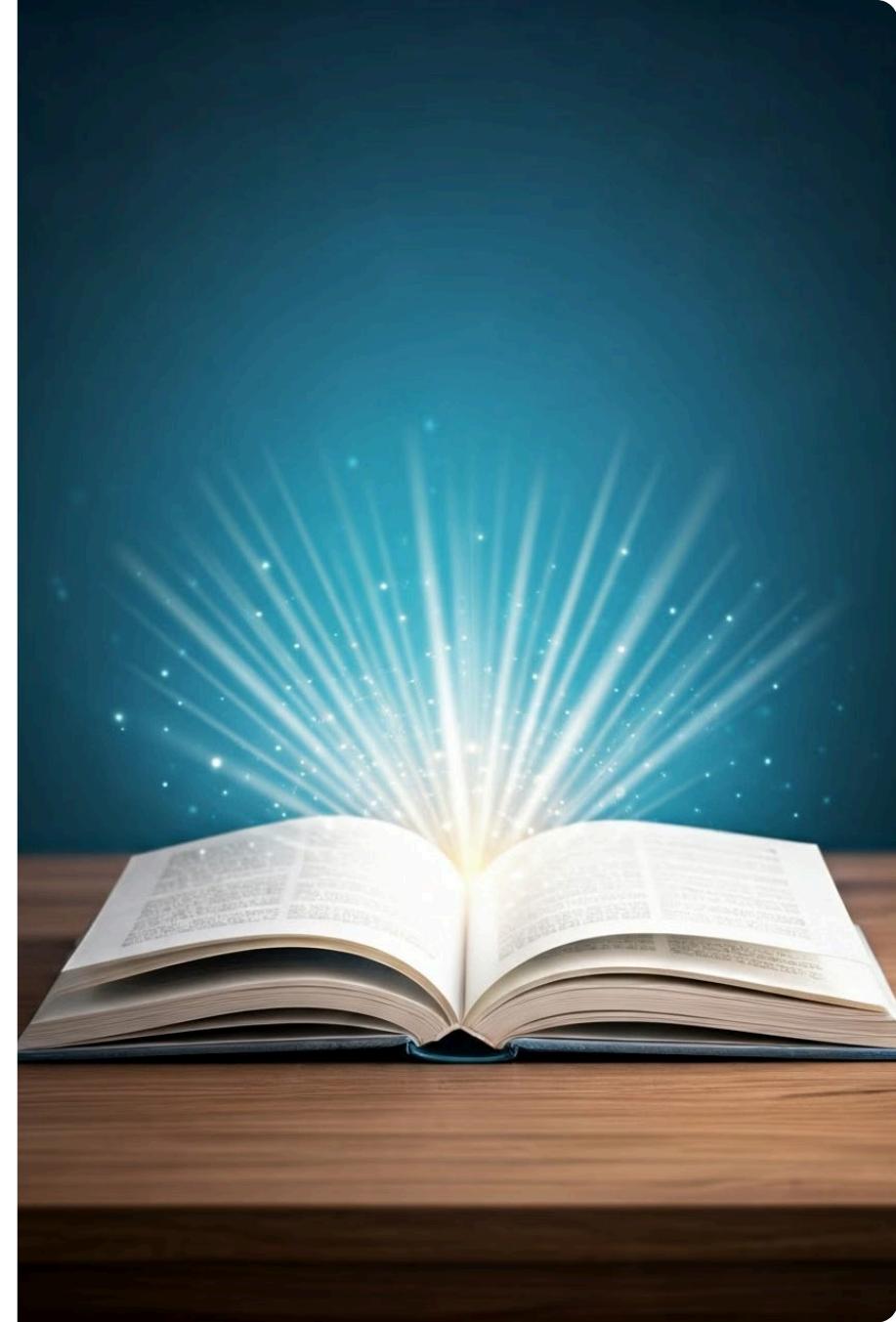
**Which of the following should you focus on when preparing for the Scrum Master certification exam??**

- A) The Agile Manifesto.
- B) Story point estimation techniques.
- C) The Scrum Guide's definition of Scrum events.
- D) Kanban board management.

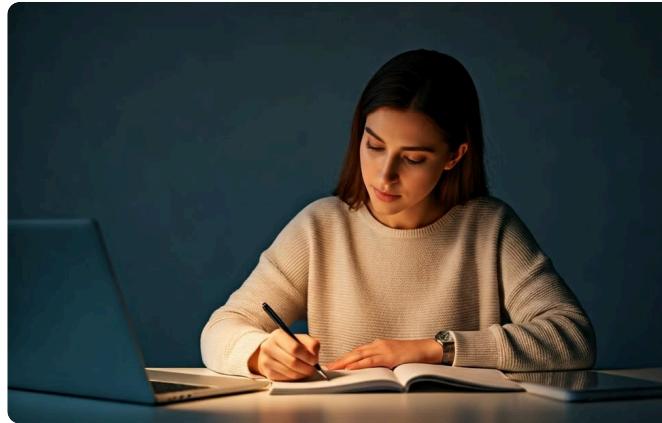
# Answer and Explanation

## 1 Correct answer: C

The Scrum Guide's definition of Scrum events is directly relevant and part of the official Scrum framework, which will be tested on the exam.



# Reflection



## Are you on track?

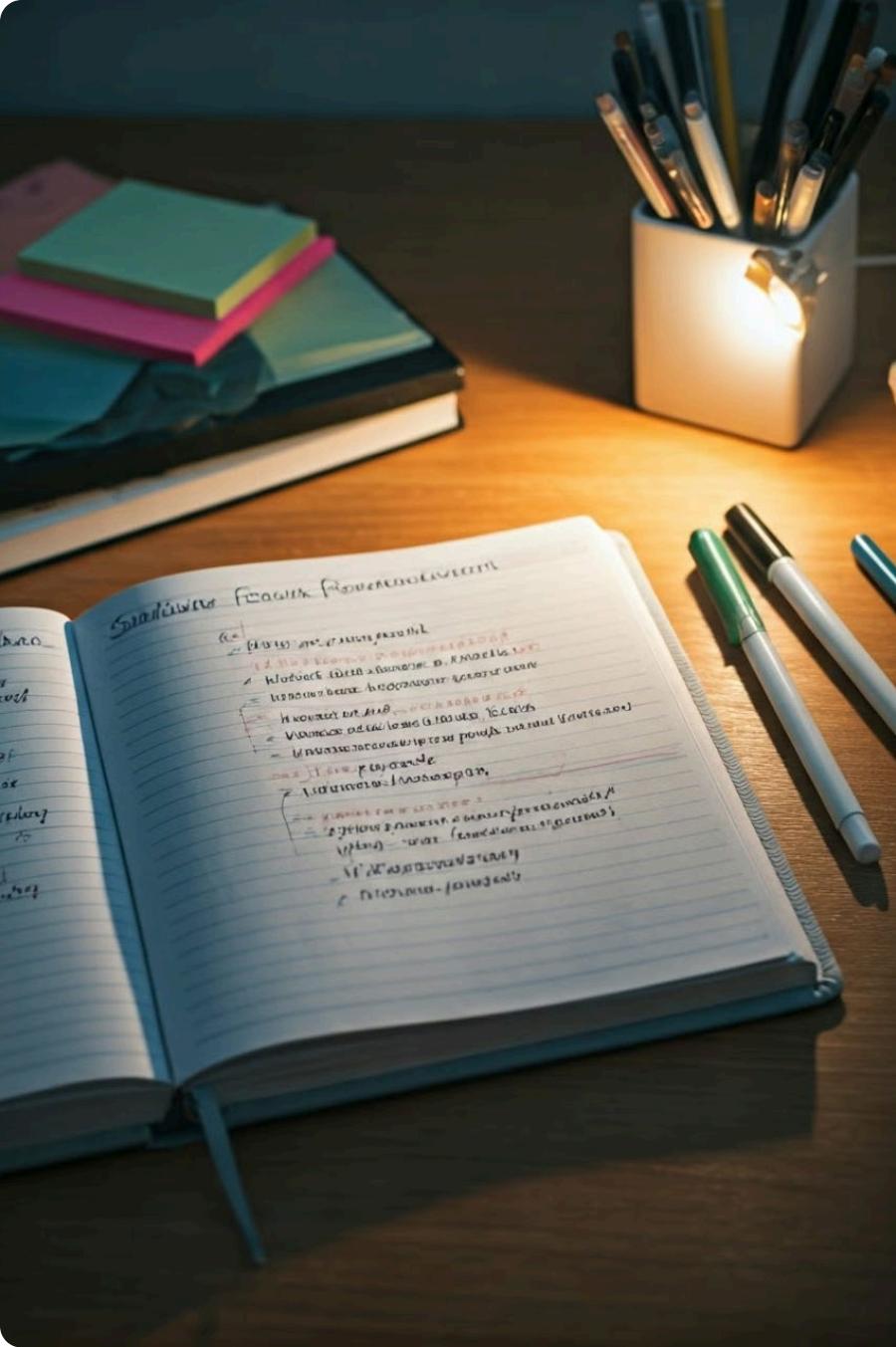
Consider whether you're spending time on topics not directly relevant to the exam. Refocus your efforts to align with the official Scrum framework.

## Realign your approach

Ensure your study plan is optimized to cover the key Scrum principles and practices needed to pass the exam.

## Stay the course

Maintain your discipline and commitment to mastering the Scrum framework. Consistent, focused effort will pay off on exam day.



# Remember

## 1 Stay focused on official Scrum framework

Avoid deviating from the core Scrum principles and practices.

## 3 Consistently review key topics

Regularly revisit the fundamental Scrum concepts to solidify your understanding.

## 2 Avoid distractions

Keep your attention on the key topics to ensure efficient preparation.

## 4 This approach helps efficient prep and exam success

Staying disciplined and focused will enable you to effectively prepare for the Scrum exam.

# **Thank You**

For your attention, and looking forward to continuing our exploration of Scrum!