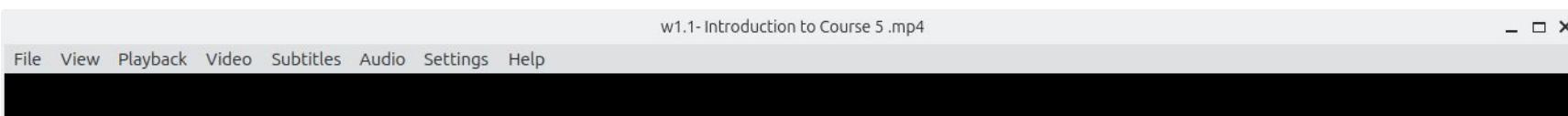
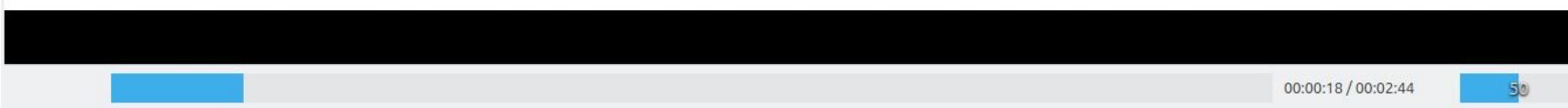


w1.1- Introduction to Course 5



Previous learnings

- Foundations of project management
- What it takes to be a project manager
- Phases of the project life cycle
- Tools and techniques for managing and communicating your plans



The screenshot shows a video player interface with a dark theme. At the top, the title "w1.1- Introduction to Course 5.mp4" is displayed. Below the title is a menu bar with options: File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area contains the text "In this course" followed by a bulleted list. The list includes "History of Agile", "Introduction to Scrum" (which is highlighted with a red border), and "Core roles that make up a Scrum Team". At the bottom of the player, there is a progress bar showing a blue segment, the text "00:01:26 / 00:02:44", and a volume control icon with the number "50".

In this course

- History of Agile
- **Introduction to Scrum**
- Core roles that make up a Scrum Team

00:01:26 / 00:02:44 50

Course Overview

The course follows a foundational program on project management, which covered:

- The basics of what it takes to be a project manager.
- The project lifecycle phases: **initiation, planning, execution, and closing.**
- Tools and techniques for managing plans and handling challenges, risks, and issues.

The new course focuses on the **Agile** approach, which the instructor describes as one of the most popular, interesting, and flexible ways to deliver projects.

What You Will Learn About Agile

- Agile is presented as an overarching philosophy to deliver value to customers, not a specific methodology itself.
- The course will prepare students for a career in Agile project management and will cover:
 - The history of Agile.
 - A specific Agile delivery framework called Scrum.
 - The core roles that make up a Scrum Team.
 - Best practices and real-world scenarios for using the Agile approach.

(all encompassing and comprehensive)

Instructor Introduction

The instructor, **Sue**, is a **Senior Technical Program Manager** at Google's support platform.

- She has been at Google since **2014**, working on **product reliability** and now building products for user support.
- Prior to Google, she managed projects using various methods, including **Waterfall** and **Agile**.
- She started her career as a **software engineer** but found her passion in **program management** because it brings disciplines together to deliver great outcomes for both customers and the business.
- She expresses excitement to share the '**aha**' moment she had when discovering Agile.

The video concludes by inviting the viewer to the next video to start learning the basics of Agile.

w1.1-Introduction to Course 5.mp4

File View Playback Video Subtitles Audio Settings Help



And I should probably introduce myself.
My name is Sue,

00:01:38 / 00:02:44 50

w1.1- Introduction to Course 5 .mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Sue
Senior Technical Program Manager

00:01:41 / 00:02:44 50

A video player window showing a woman with blonde hair, wearing a green top and a necklace, looking slightly down and to her left. The video is titled "w1.1- Introduction to Course 5 .mp4". The player interface includes a menu bar with File, View, Playback, Video, Subtitles, Audio, Settings, and Help. A subtitle overlay on the left says "Subtitles off". At the bottom, there is a red bar with the text "Sue" and "Senior Technical Program Manager". The video progress bar shows 00:01:41 / 00:02:44 and a volume icon showing 50%.

w1.2- A brief history of Agile

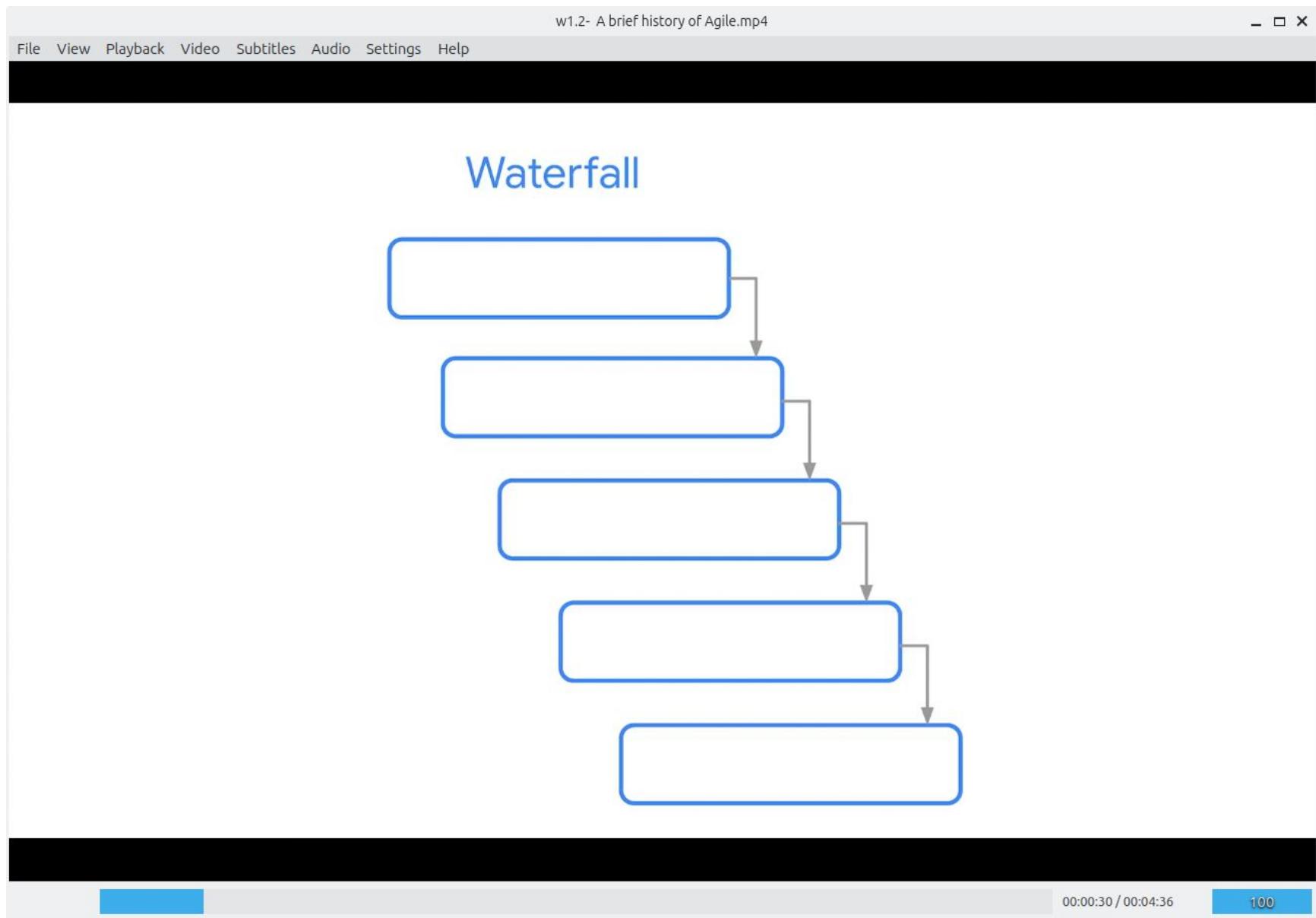
w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

In this video

- History of Agile
- Introduce the Agile values and principles
- Agile in different industries

00:00:25 / 00:04:36 100



w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Waterfall

The sequential or linear ordering of phases

00:00:37 / 00:04:36 100

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Agile

- Being able to move **quickly and easily**
- Flexibility and the **willingness and ability** to change and adapt

00:00:57 / 00:04:36 100

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Iterative approach

The diagram illustrates the iterative approach of Agile methodology. It features a large grey circle with a clockwise arrow indicating the flow of the process. Inside the circle, the word "Agile" is written in blue. Four white rectangular boxes with blue borders are positioned at the top, bottom, left, and right points of the circle, representing iterative cycles or phases. The entire diagram is set against a white background.

00:01:00 / 00:04:36 100

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Agile- Mini Waterfall

The diagram illustrates the concept of 'Agile- Mini Waterfall'. It features eight small waterfall icons arranged in two columns of four. Each icon depicts a green mountain-like base with a blue waterfall cascading down it. Above each waterfall, there is a sequence of three horizontal bars: a yellow bar at the top, followed by a blue bar, and another yellow bar at the bottom. Arrows point from the end of one bar to the start of the next, indicating a sequential flow. The entire diagram is set against a white background.

You can think of it as a lot of mini waterfalls for each activity.

00:01:30 / 00:04:36 100

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Agile's **iterative approach** enables a project to move quickly, as well as making it adaptive to change.

00:01:33 / 00:04:36 100

A screenshot of a video player window titled "w1.2- A brief history of Agile.mp4". The window has a menu bar with options: File, View, Playback, Video, Subtitles, Audio, Settings, and Help. A black bar covers the subtitle area, with the text "Subtitles off" visible. The main video frame displays a blue text quote: "Agile's **iterative approach** enables a project to move quickly, as well as making it adaptive to change." The word "iterative approach" is highlighted with a red rectangular box. At the bottom of the window, there is a progress bar showing "00:01:33 / 00:04:36" and a volume slider set to "100".

Waterfall vs. Agile: Core Differences

Feature	Waterfall	Agile
Process	Sequential or linear ordering of phases. One phase must be completed before the next begins (like a river flowing down a mountain).	Iterative approach. Project processes are repeated in many shorter blocks of time called iterations .
Flexibility	Does not encourage changing the process once it has started.	Defined by flexibility, repetition, and openness to change ; incorporates necessary changes throughout the process.
Execution	The project is treated as one long process with distinct phases .	The team completes a subset of all project activities in each iteration ("mini- watertails

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Agile project management

An approach to project and team management
that embodies “agility” and is based on the
Agile Manifesto

00:01:48 / 00:04:36 100

The image shows a screenshot of a video player window titled "w1.2- A brief history of Agile.mp4". The window has a menu bar with options: File, View, Playback, Video, Subtitles, Audio, Settings, and Help. Below the menu is a large black video frame. In the center of the slide, the title "Agile project management" is displayed in a large blue font. Below the title, a definition of Agile project management is given in blue text: "An approach to project and team management that embodies “agility” and is based on the Agile Manifesto". The word "Agile Manifesto" is enclosed in a red rectangular box. At the bottom of the slide, there is a progress bar showing "00:01:48 / 00:04:36" and a volume control showing "100".

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

The Manifesto is a collection of four values and
12 principles that define the mindset that all Agile
teams should strive for.

00:01:55 / 00:04:36 100

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Agile in historical context

The timeline illustrates the historical context of Agile development, spanning from the 1880s to 2014-15, categorized into four main phases: Foundations, Emergence, Democratization, and Modernization.

Foundations: 1880s (Study of organizational behavior begins) and 1930s (Toyota Way put into practice).

Emergence: 1970 (Waterfall model created), 1990s (Agile and Lean Thinking emerges), 1995 (Scrum codified and put into use), and 2001 (Agile Manifesto & principles created).

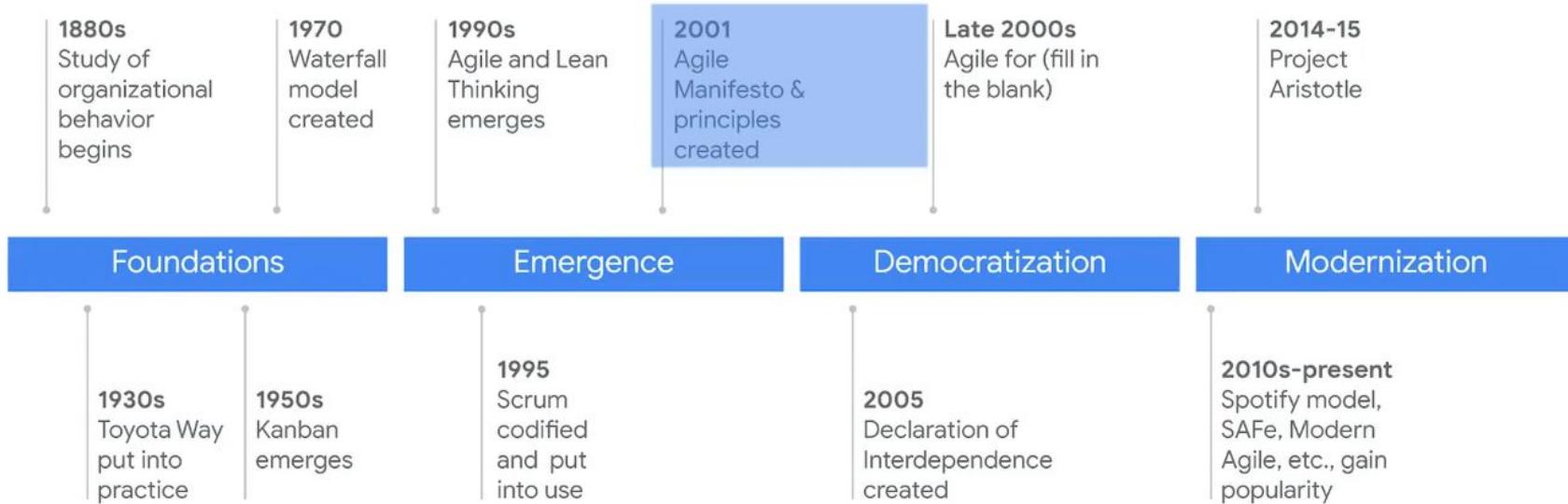
Democratization: 2005 (Declaration of Interdependence created) and Late 2000s (Agile for (fill in the blank)).

Modernization: 2014-15 (Project Aristotle) and 2010s-present (Spotify model, SAFe, Modern Agile, etc., gain popularity).

Source: Unify Consulting 'Agile in Context' Timeline

00:02:37 / 00:04:36 100

Agile in historical context



Source: Unify Consulting 'Agile in Context' Timeline

The screenshot shows a video player window titled "w1.2- A brief history of Agile.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area displays a slide with the text "Pleasing their customers" in blue, followed by a subtitle "pleasing their customers." in black. The bottom of the screen shows a progress bar at 00:03:25 / 00:04:36 and a volume indicator at 100%.

w1.2- A brief history of Agile.mp4

File View Playback Video Subtitles Audio Settings Help

Pleasing their customers

pleasing their customers.

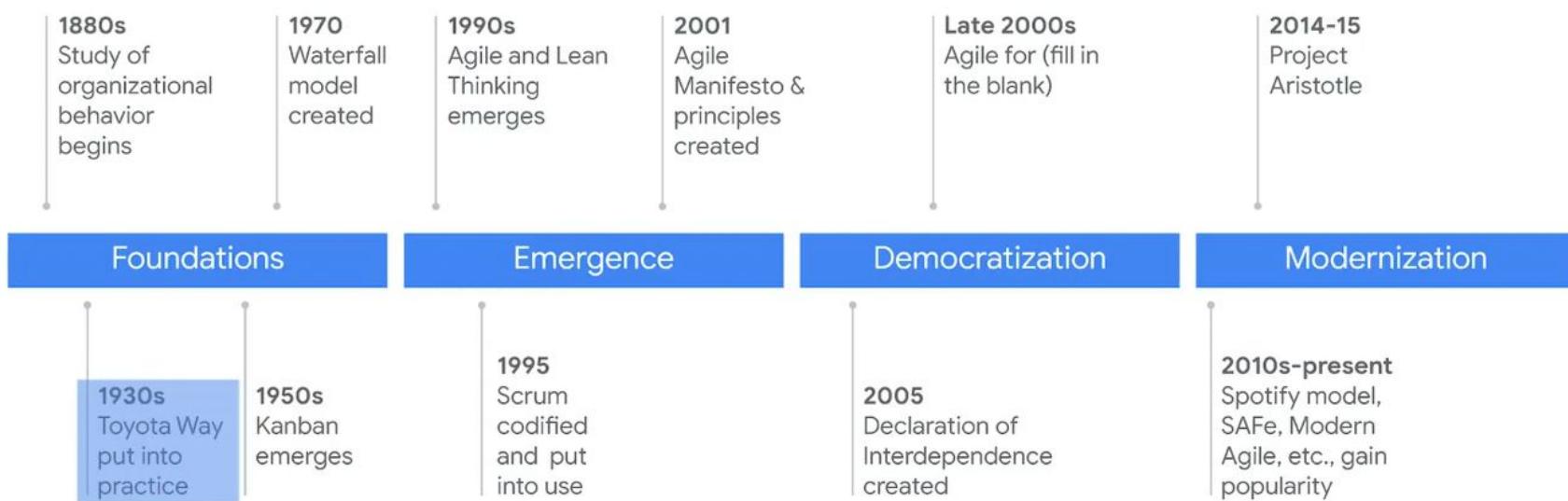
00:03:25 / 00:04:36 100

The image shows a screenshot of a video player window titled "w1.2- A brief history of Agile.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays a list of bullet points. The first two bullet points are highlighted with red boxes. The video player has a black bar at the bottom and a blue progress bar.

- Agile values, principles, and frameworks have been applied to every industry
- Agile methods draw heavily on Lean manufacturing principles

00:04:01 / 00:04:36 100

Agile in historical context



Source: Unify Consulting 'Agile in Context' Timeline

History and Adoption of Agile

Origin

- **Emergence:** Agile methodologies emerged organically in the **1990s** as the **software industry boomed**, with companies needing faster ways to build better software to stay competitive.
- **Agile Manifesto (2001):** Thought leaders from various new processes came together to find common ground and solve a core problem: companies were too focused on planning and documentation and were **losing sight of pleasing their customers**.
- **Core Philosophy:** The Manifesto guides teams to prioritize **flexibility** and focus on **people** (the team and the users) over final products and documentation.

Widespread Use

- While originating in software (which includes apps, websites, code for medical devices, agriculture, etc.), Agile's principles are now applied to **nearly every industry** due to its success.
- Agile methods draw heavily on **Lean manufacturing principles**, which originated at **Toyota's car factories in the 1930s**.
- Today, Agile is adopted across industries like **aeronautical, healthcare, education, and finance**.

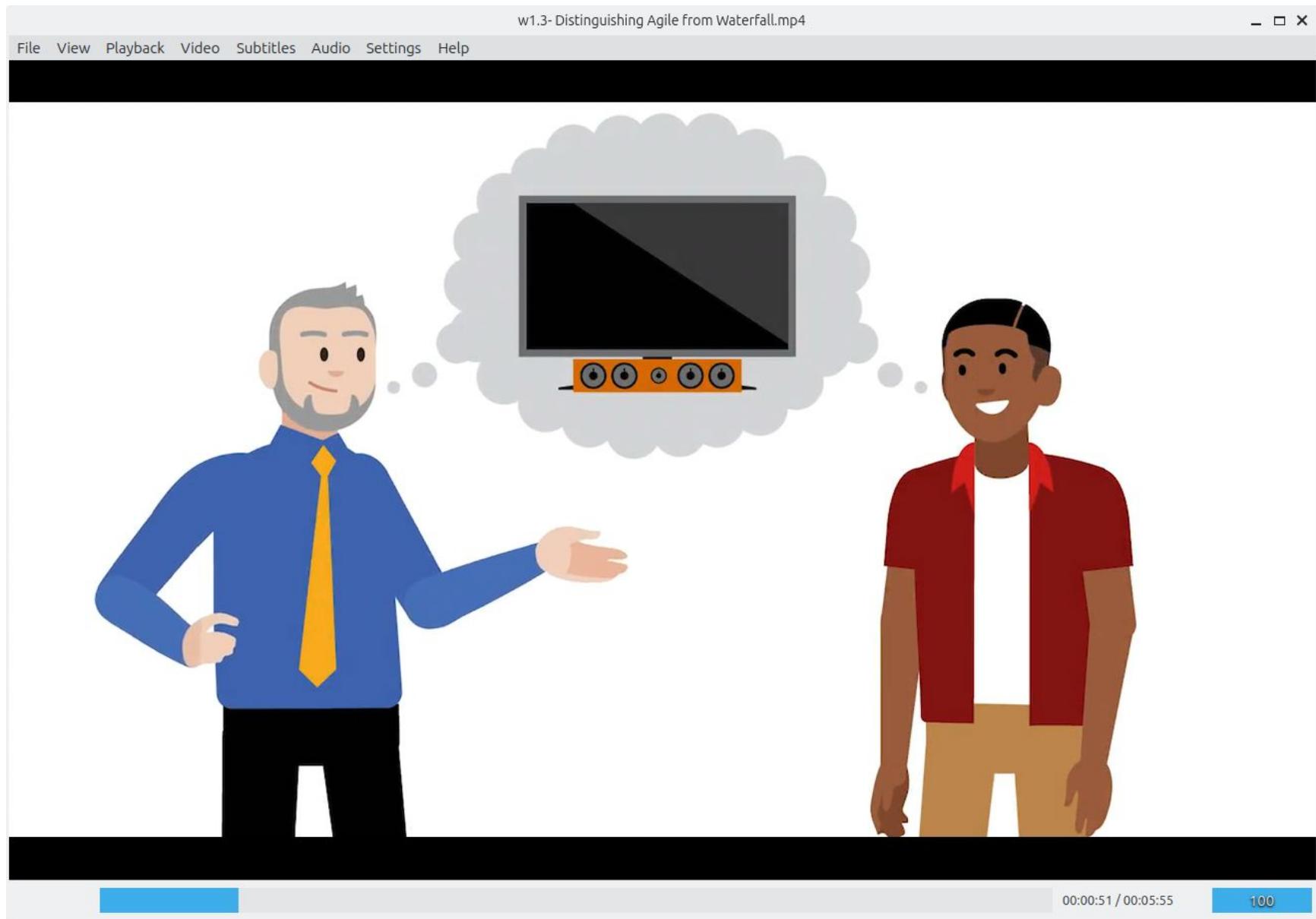
w1.3- Distinguishing Agile from Waterfall

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help

- Compare Agile and Waterfall
- Explain key elements of Agile that distinguish it from Waterfall
- Agile was created in response to the strict linear process of Waterfall
- Agile embraces the reality that the world, customer markets, and users are uncertain and unpredictable

00:00:46 / 00:05:55 100



w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help

- Agile aims to get customer feedback more quickly
- Working with an Agile mindset means always finding ways to work more efficiently by focusing on streamlining the process without reducing product quality or value
 - (# Unnecessary documentation, don't like the features)
- Agile aims to reduce waste

00:01:21 / 00:05:55 100

The Mindset: Change and Efficiency

The instructor highlights that **Agile** was created in response to the rigid, linear structure of **Waterfall**:

- Waterfall aims for **predictability** and tries to avoid change. 
- Agile embraces the reality that markets and users are **uncertain and unpredictable**. It aims to get **customer feedback more quickly** to ensure the team is building what the customer truly wants. 
- The Agile mindset focuses on **efficiency** by **reducing waste**, such as unnecessary documentation or spending time on features customers won't like. This is achieved through increased **team and stakeholder collaboration** for earlier feedback. 

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help

Aspects of a project

- Requirements
- Documentation
- Deliverables

00:01:58 / 00:05:55 100

The screenshot shows a video player interface with a presentation slide. The title bar reads 'w1.3- Distinguishing Agile from Waterfall.mp4'. The menu bar includes 'File', 'View', 'Playback', 'Video', 'Subtitles', 'Audio', 'Settings', and 'Help'. The main content area displays the slide with the title 'Aspects of a project' and a bulleted list: 'Requirements', 'Documentation', and 'Deliverables'. The word 'Deliverables' is highlighted with a red rectangular border. At the bottom, there is a progress bar showing '00:01:58 / 00:05:55' and a volume slider set to '100'.

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help

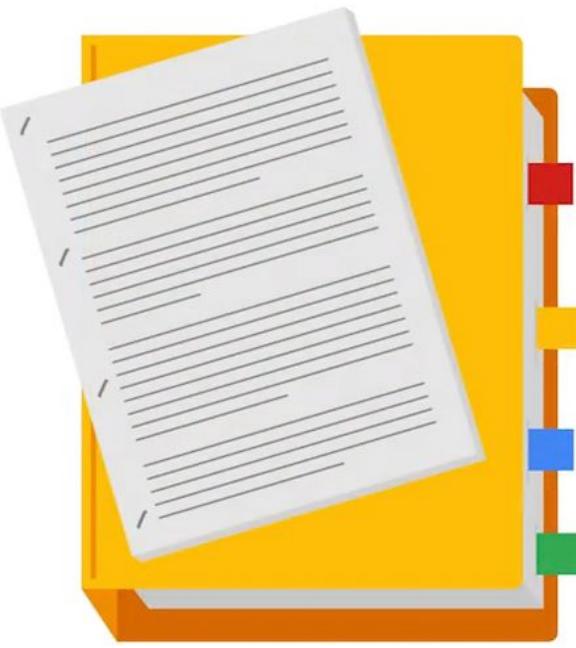
Requirements

Conditions that must be met or tasks that must be finished to ensure the successful completion of the project

00:02:05 / 00:05:55 100

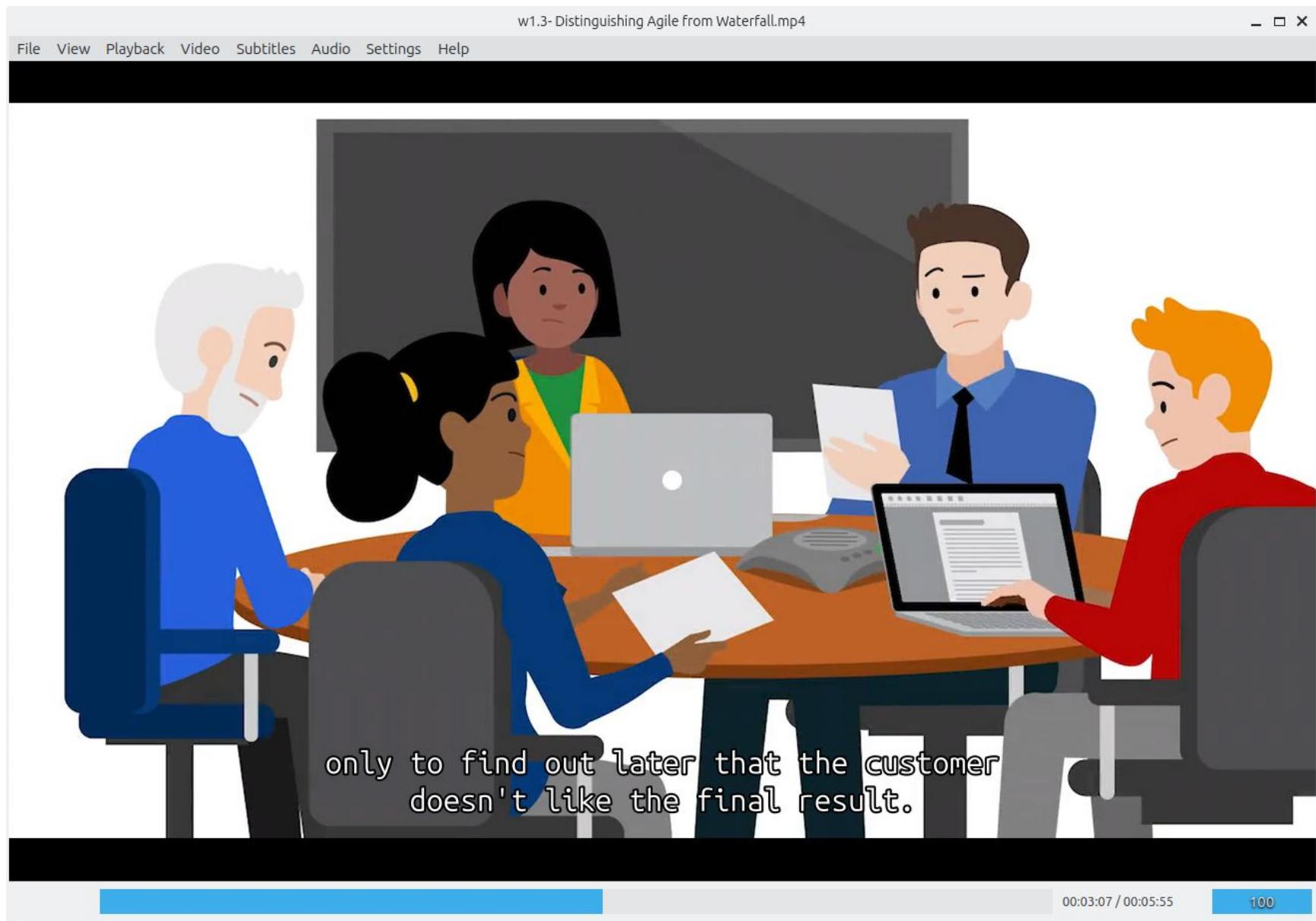
w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help



goals based on mandated regulation.

00:02:57 / 00:05:55 100



w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help

The diagram shows a vertical stack of horizontal bars representing a backlog. The top half of the stack is blue, and the bottom half is yellow. A blue curved arrow originates from the right side of the yellow bar stack and points to two circles on the right. One circle is yellow and labeled 'New idea'. The other is blue and labeled 'User changed mind'. A dashed orange arrow also points from the 'User changed mind' circle back towards the backlog.

Backlog

New idea

User changed mind

but that list of requirements and features is continuously growing and

00:03:24 / 00:05:55 100

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help

User changed mind

New idea

Backlog

changing throughout the project.

00:03:26 / 00:05:55 100

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help



always ~~moving the most urgent~~ or
valuable items to the top of the list.

00:03:34 / 00:05:55 100

Comparison of Key Project Aspects

1. Requirements (What needs to be built)

Waterfall

Requirements are generally set at the start and formalized in documents like the **Product Requirements Document (PRD)**.

The process is **strict** and often involves a **Change Control Board** to manage changes, aiming to prevent scope creep.

This works well when the end product is **known and understood** (e.g., mandated regulation).

Agile

Requirements are treated as **dynamic** and **expected to change** based on feedback and new information.

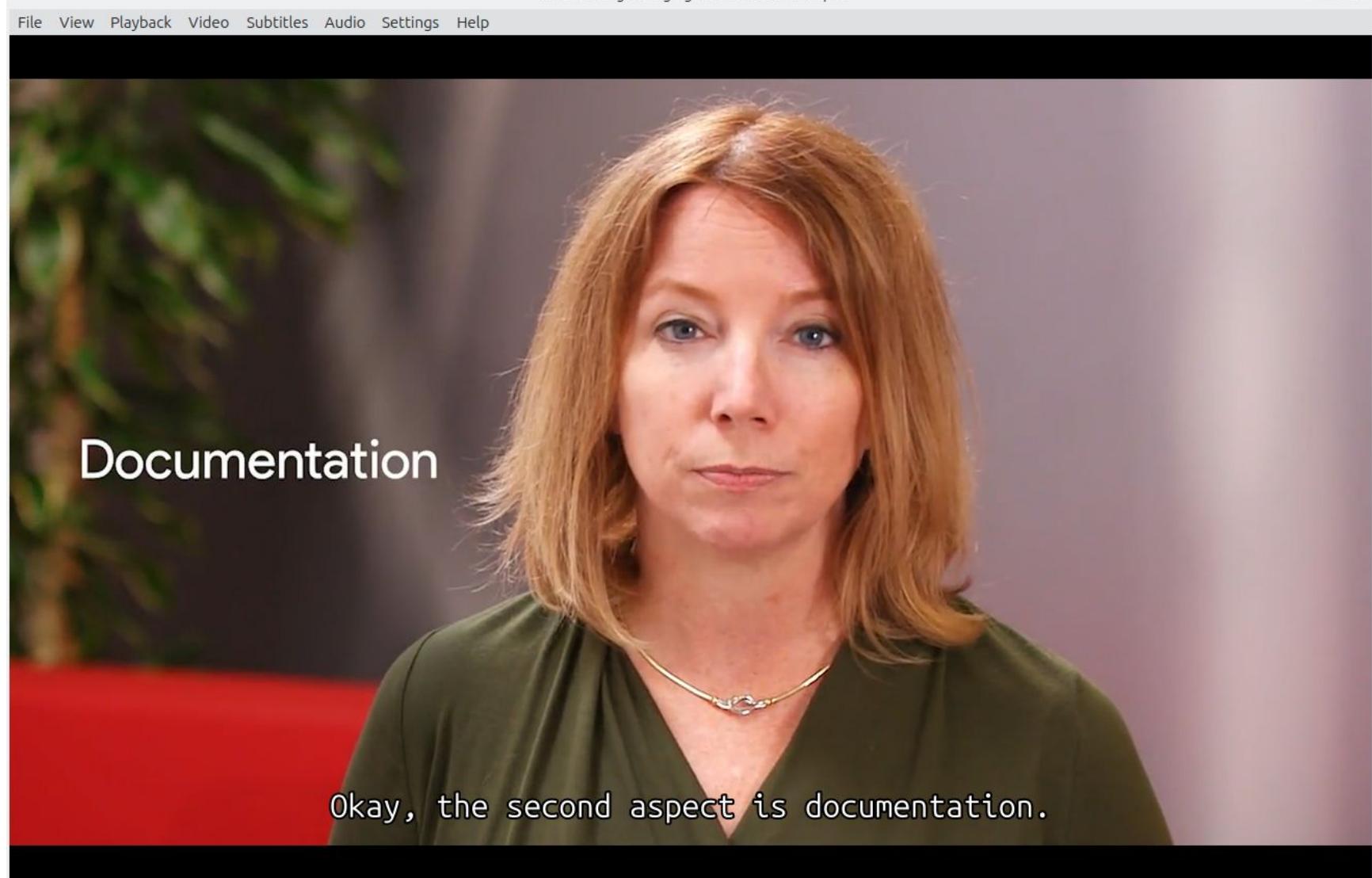
The team works with stakeholders to **continuously prioritize requirements**, focusing on the most urgent or valuable items.

The team works on requirements in **iterations**, getting **quick and frequent feedback** to make necessary adjustments.

A PRD defines what a product should do and why it should be built, guiding the entire development process from design to launch.

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help



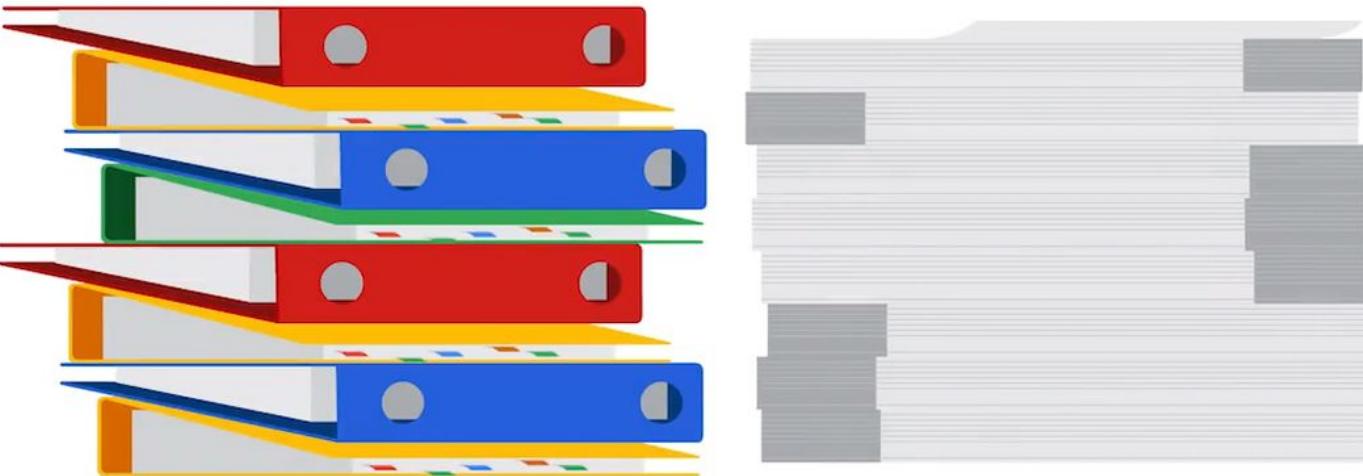
Documentation

Okay, the second aspect is documentation.

00:03:57 / 00:05:55 100

w1.3- Distinguishing Agile from Waterfall.mp4

File View Playback Video Subtitles Audio Settings Help



of handoffs between phases and handoffs among different teams within the project.

00:04:09 / 00:05:55 100

2. Documentation (How the work is recorded)

Waterfall

Uses lots of **documentation** (plans, schedules, charters) **due to handoffs** between phases and different teams, and because work is done in bigger chunks.

Documents are typically large, formal, and subject to a rigorous change management and approval process.

Agile

Emphasizes **real-time, person-to-person conversations** over rigid, formal documents.

Documentation is **shorter**, written **only as needed**, and contains **just enough detail for** the reader to get their job done.

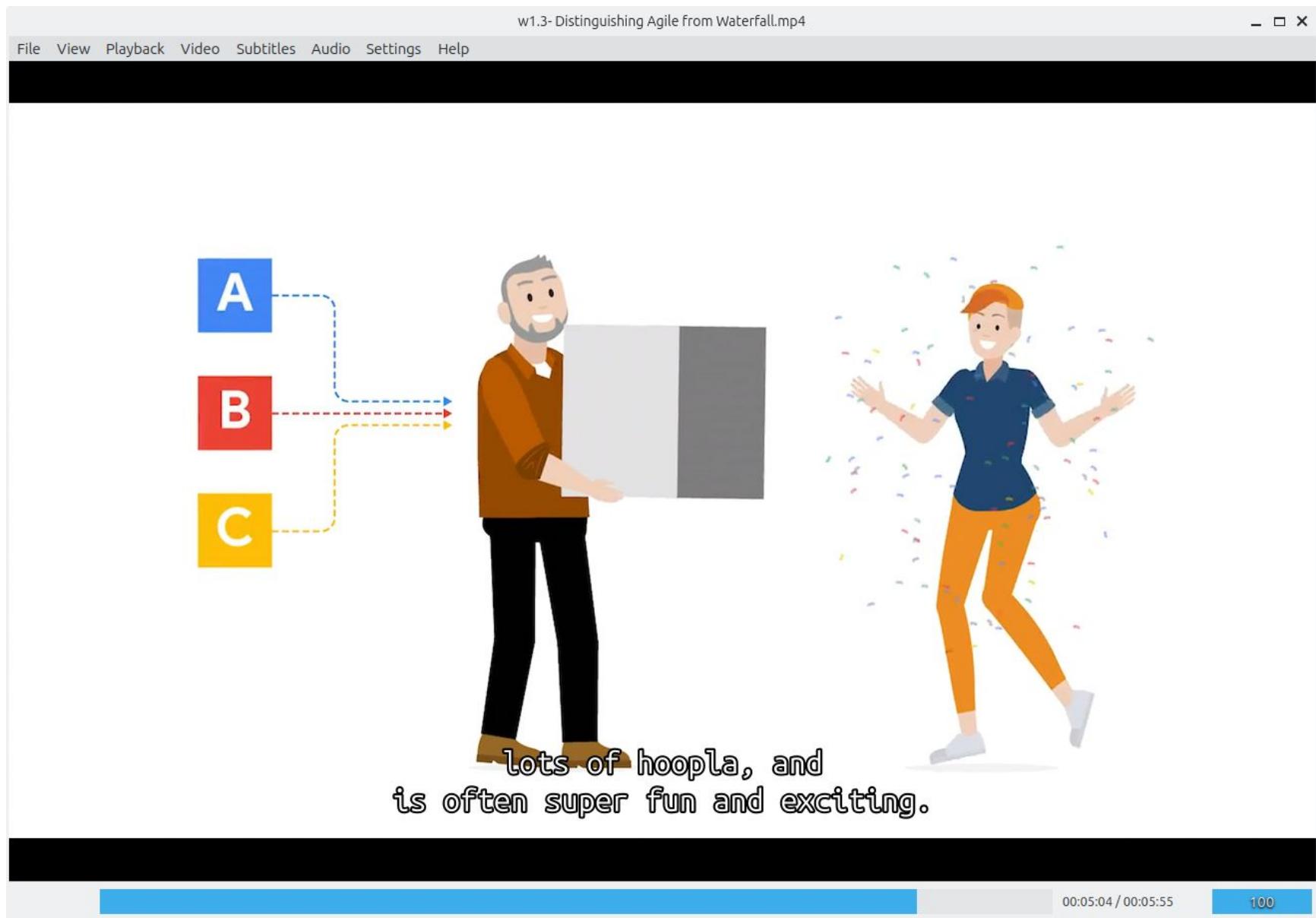
The screenshot shows a video player interface with a presentation slide. The title bar reads "w1.3- Distinguishing Agile from Waterfall.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays the following text:

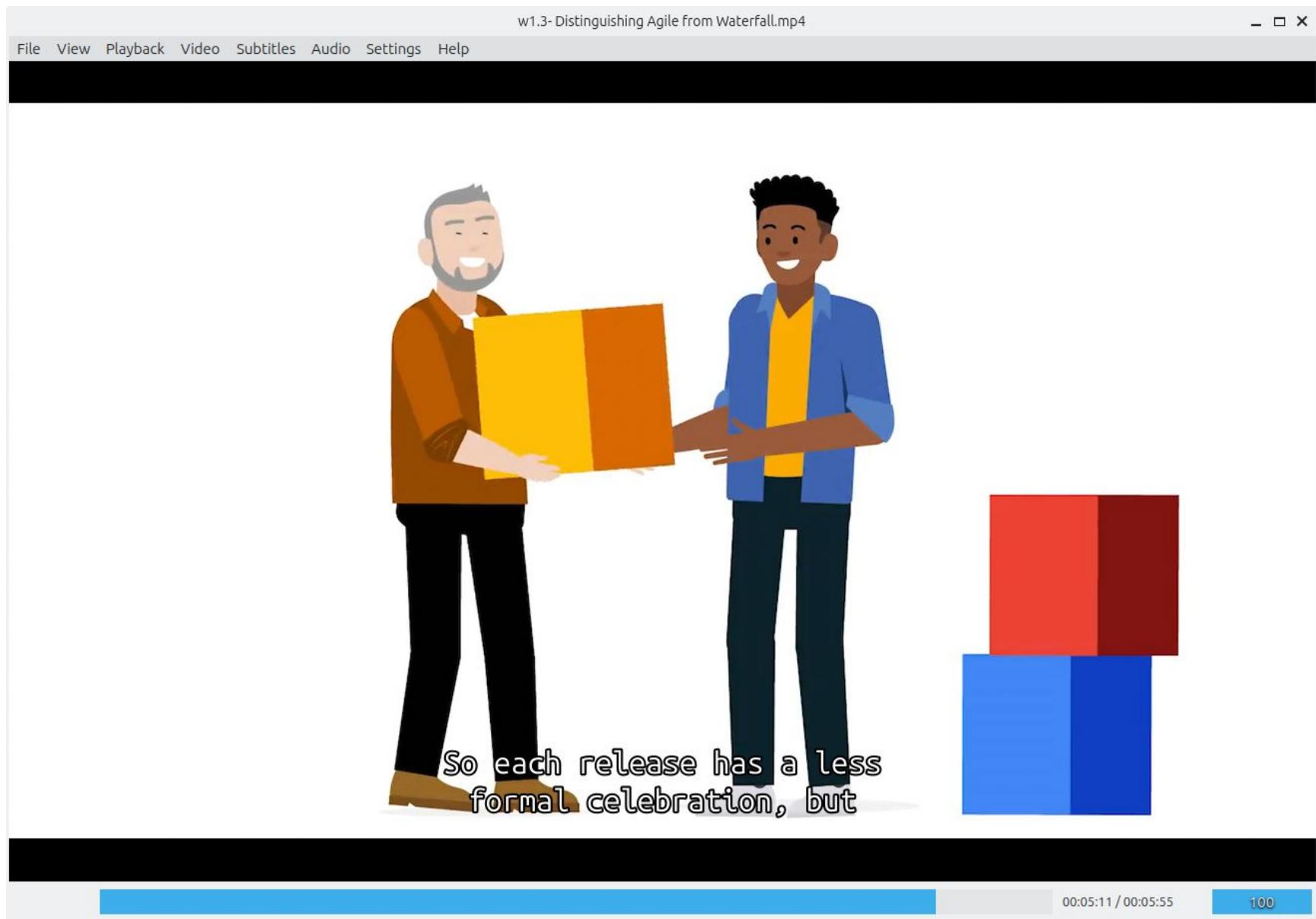
Deliverable

A tangible outcome from a project

A deliverable is a tangible outcome from a project.

At the bottom, there is a black bar with a blue progress bar, a timestamp "00:04:51 / 00:05:55", and a "100" percentage indicator.





3. Deliverables (The final output)

Waterfall

The deliverable is often **released once at the very end of the project**, which is a major, celebratory event.

Risk: Without regular feedback, the team risks delivering a product the customer doesn't want.

Agile

Features are released in **smaller, more frequent releases**.

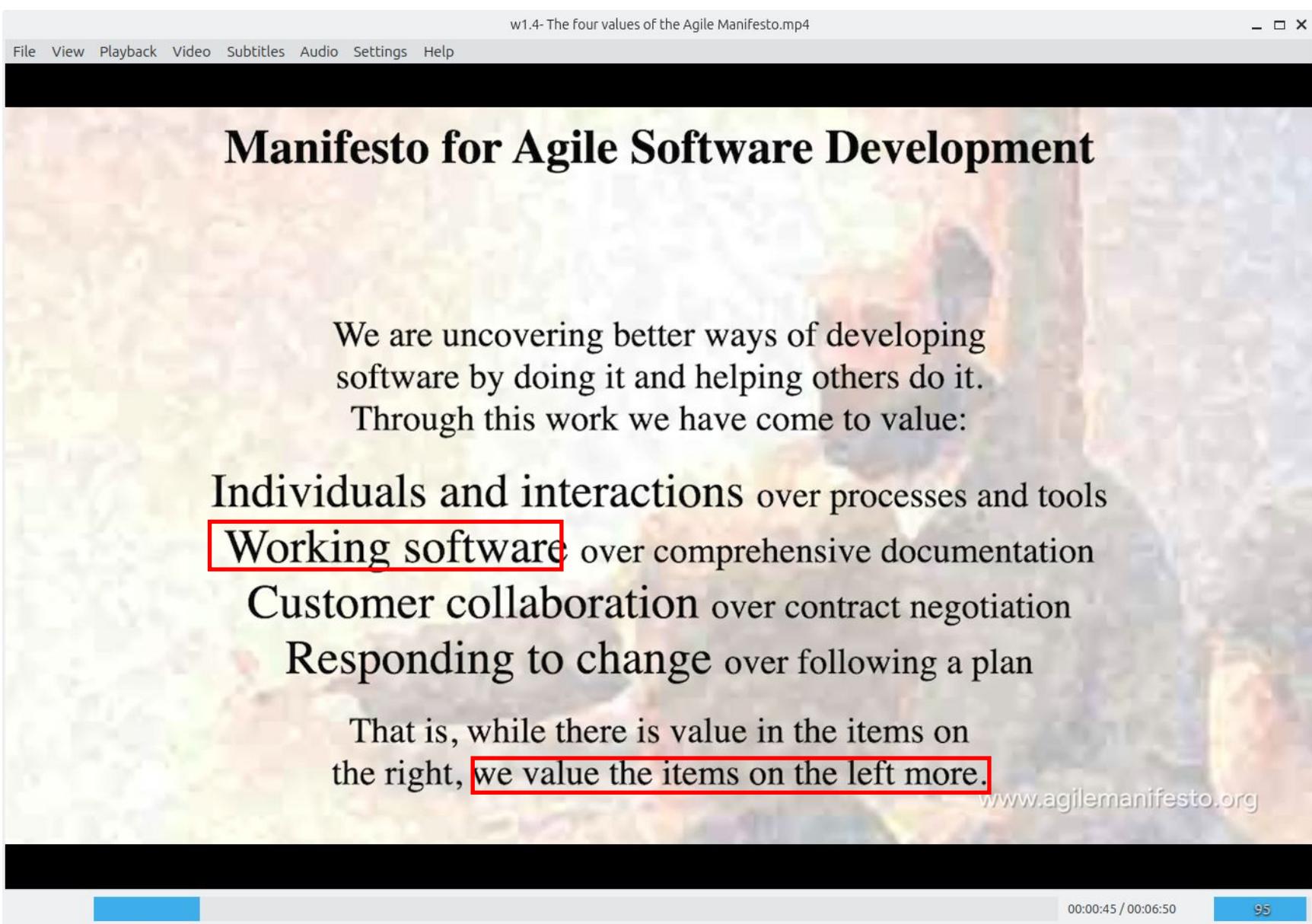
Benefit: Regular feedback from these steady releases allows the team to **learn and adapt as they go**, especially in new or uncertain markets.

w1.4- The four values of the Agile Manifesto

w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Manifesto for Agile Software Development



We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software** over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

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

www.agilemanifesto.org

00:00:45 / 00:06:50 95

Agile Manifesto

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

Through this work we have come to value:

(With brief conversation)

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

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

<http://agilemanifesto.org/>

Kent Beck
Mike Beedle
Arlie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

© 2001, the above authors this declaration may be freely copied in any form, but only in its entirety through this notice.

Detailed Breakdown of the Values

1. Individuals and Interactions over Processes and Tools

- **Emphasis:** Stresses that direct **communication and collaboration** between people are more efficient and effective than relying on rigid processes and tools (e.g., a quick conversation is better than a long email chain).
- **Goal:** Processes and tools should **facilitate** teamwork and project management, not become a barrier to good collaboration.

2. Working Software over Comprehensive Documentation

- **Emphasis:** Prioritizing the creation of the final **value-generating product/deliverable** (e.g., a legal brief, office design, or working software) over spending excessive time debating, writing, and reviewing documentation.
- **Goal:** It's more important to deliver the valuable item the customer wants than to perfectly document the process used to create it.

3. Customer Collaboration over Contract Negotiation

- **Emphasis:** Placing **customer satisfaction** as the highest priority and seeking opportunities to **collaborate early and often** with customers and stakeholders (e.g., initial product testing, feedback on prototypes). 
- **Goal:** To quickly adapt to customer needs rather than **waiting for formal sign-offs**, renegotiating contract terms, or filing change requests, which are viewed as less flexible.

The meaning of a formal sign-off is that an authorized individual is accepting responsibility and confirming that the work meets the required standards, is complete, and can proceed to the next stage

4. Responding to Change over Following a Plan

- **Emphasis:** Acknowledging that **change is inevitable** and that complex, long projects contain high uncertainty.
- **Goal:** The most successful projects are those that **can smoothly integrate change**. Agile Project Managers still create and value plans, but they must be prepared to revise and adapt those plans at any point to **maximize customer value**, even if it affects the initial schedule or budget. 

w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Agile values and principles inform the why, how, and what of Agile project management planning and processes.

00:01:59 / 00:06:50 95

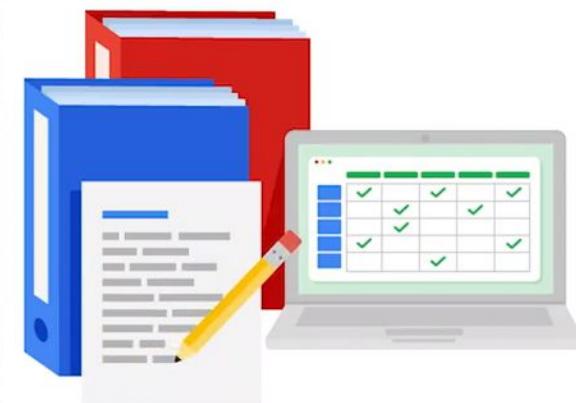
A screenshot of a video player window titled "w1.4- The four values of the Agile Manifesto.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays a blue text slide: "Agile values and principles inform the why, how, and what of Agile project management planning and processes." The words "the why, how," and "what of Agile project" are highlighted with red boxes. A black bar covers the top and bottom portions of the slide. At the bottom, there is a progress bar showing "00:01:59 / 00:06:50" and a page number "95".

w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help



- Work together.
- Collaborate.
- Help each other achieve the best outcomes possible.
- Value individual perspectives and creativity as important contributions to the success of the project.



00:02:49 / 00:06:50

95

w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

(deliver products to customer)
(Not debating)

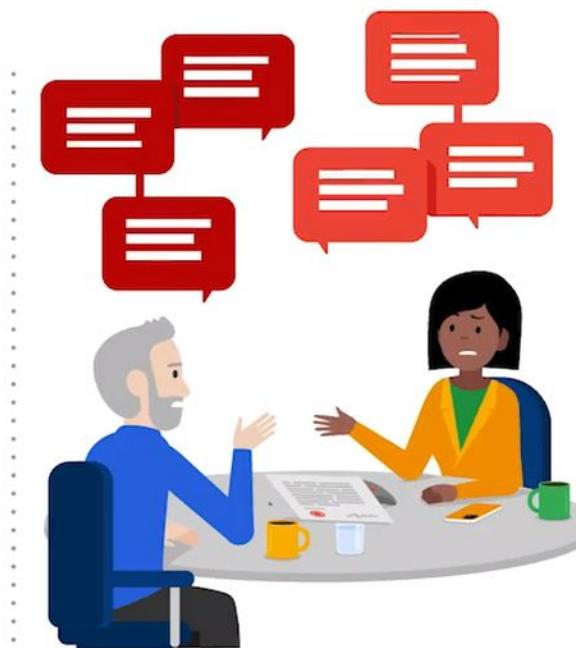
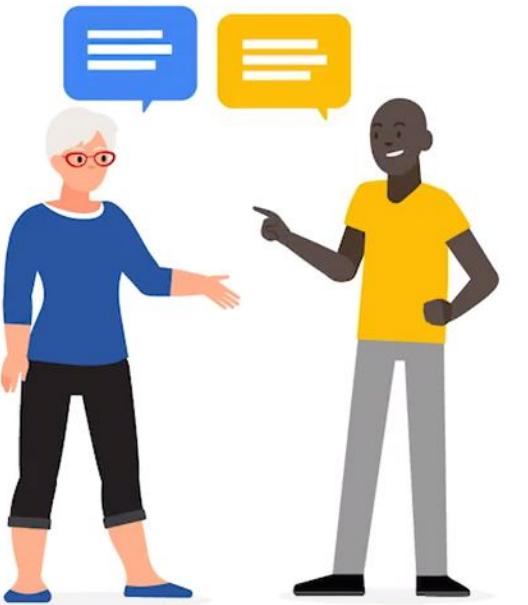
- Work on things that create value.

actually create value and avoid spending any more time than they really need on

00:03:16 / 00:06:50 95

w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

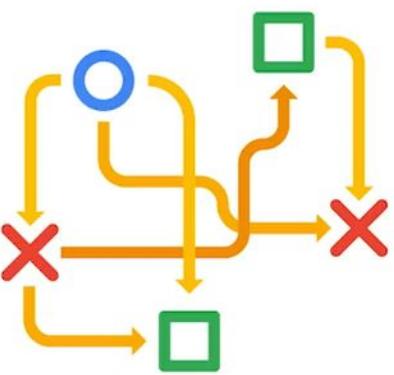


- Collaborate with customers early and often.
- React and adapt to feedback.

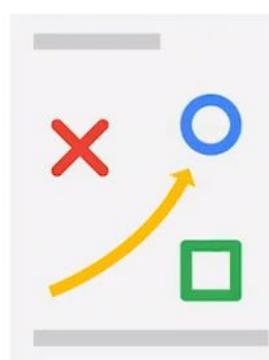
00:04:34 / 00:06:50 95

w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help



- Acknowledge that change is inevitable.
- Adapt to changes at any time during the project.



00:05:50 / 00:06:50 95

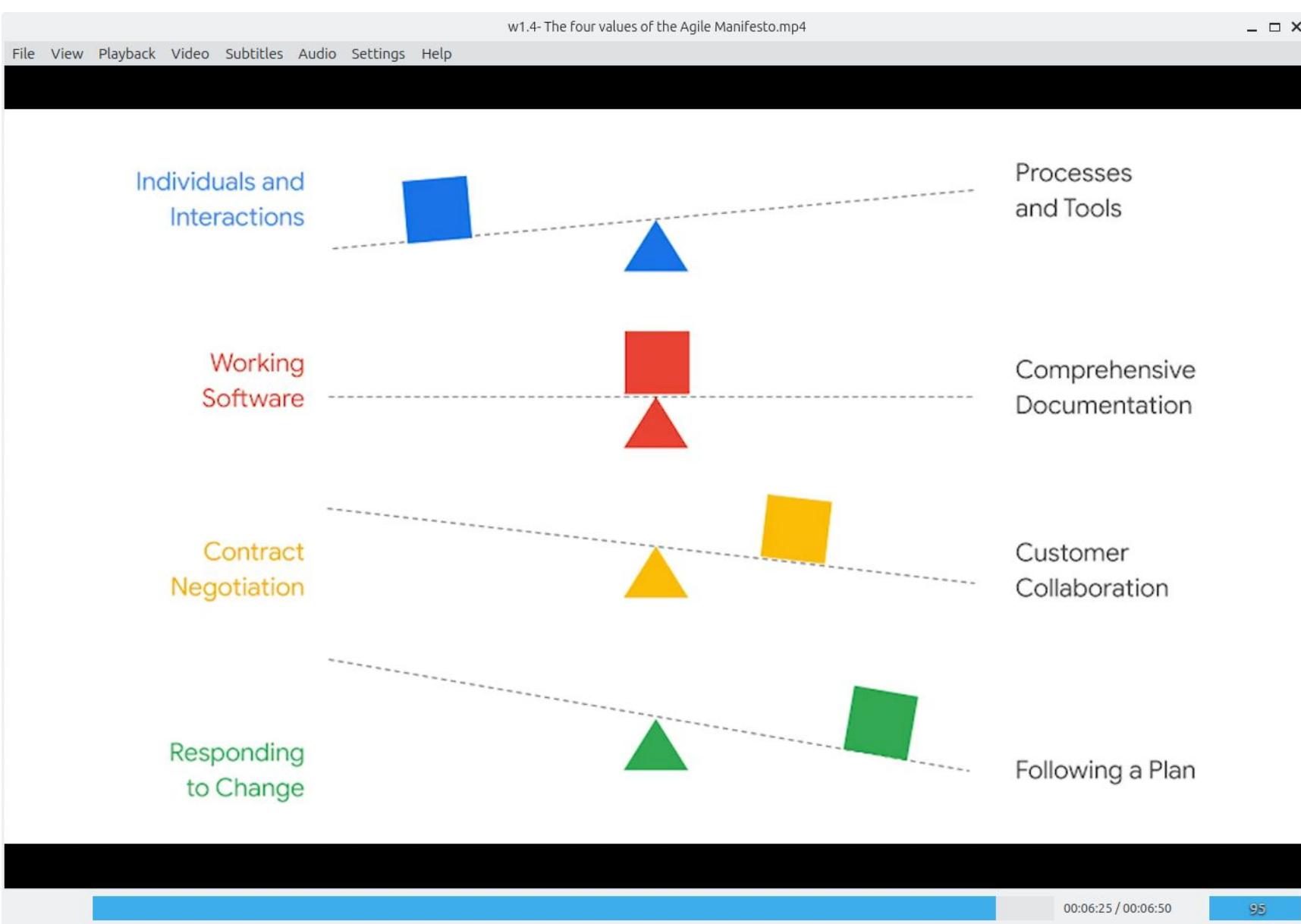
w1.4- The four values of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

The 4 Agile values

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

00:06:18 / 00:06:50 95



w1.5- The 12 principles of the Agile Manifesto

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Value Delivery

How do Agile teams deliver highly valuable products to their customers?

Business Collaboration

How do Agile teams collaborate with their business partners and stakeholders to create business value to the organization?

Team Dynamics and Culture

How does a team create and maintain the right interpersonal and team dynamics to deliver value for the customers and the business?

Retrospectives and Continuous Learning

How does the project learn to continuously increase performance of an organization and business?

00:00:37 / 00:07:19 50

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.

Working software is the **primary measure of progress.**

Simplicity—the art of maximizing the amount of work not done—**s essential.**

Continuous attention **to technical excellence** and good design enhances agility.

00:01:15 / 00:07:19

50

"Working" software is not just code that compiles. It means the feature is fully integrated, thoroughly tested (unit, QA, and acceptance tests), and meets the team's Definition of Done (DoD). It must be in a deployable or potentially shippable state.

The Four Themes of Agile Principles

1. Value Delivery (5 Principles)

This theme is about delivering the project's work **as quickly as possible** to gain **feedback** and mitigate the risk of building the wrong thing.

- **Core Idea:** Value isn't realized until the work is delivered. By replacing "software" with "deliverables" or "solutions," these principles apply to any project (e.g., "Deliver working solutions frequently").
- **Simplicity:** The theme emphasizes simplicity to allow the team to focus on what matters most and avoid adding unnecessary complexity that confuses the user.
- **Example in Action:** Prioritizing getting **early feedback on a prototype**, ensuring the team only works on approved features, and dedicating about 10% of time to bug fixing or process polishing to speed up future iterations.

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Value delivery

Delivering the work as quickly as possible in order
to get feedback and mitigate time risk

00:01:32 / 00:07:19 50

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Our highest priority is to satisfy the customer through early and continuous delivery **of valuable solutions.**

Deliver working solutions frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.

Working solutions is the primary measure of progress.

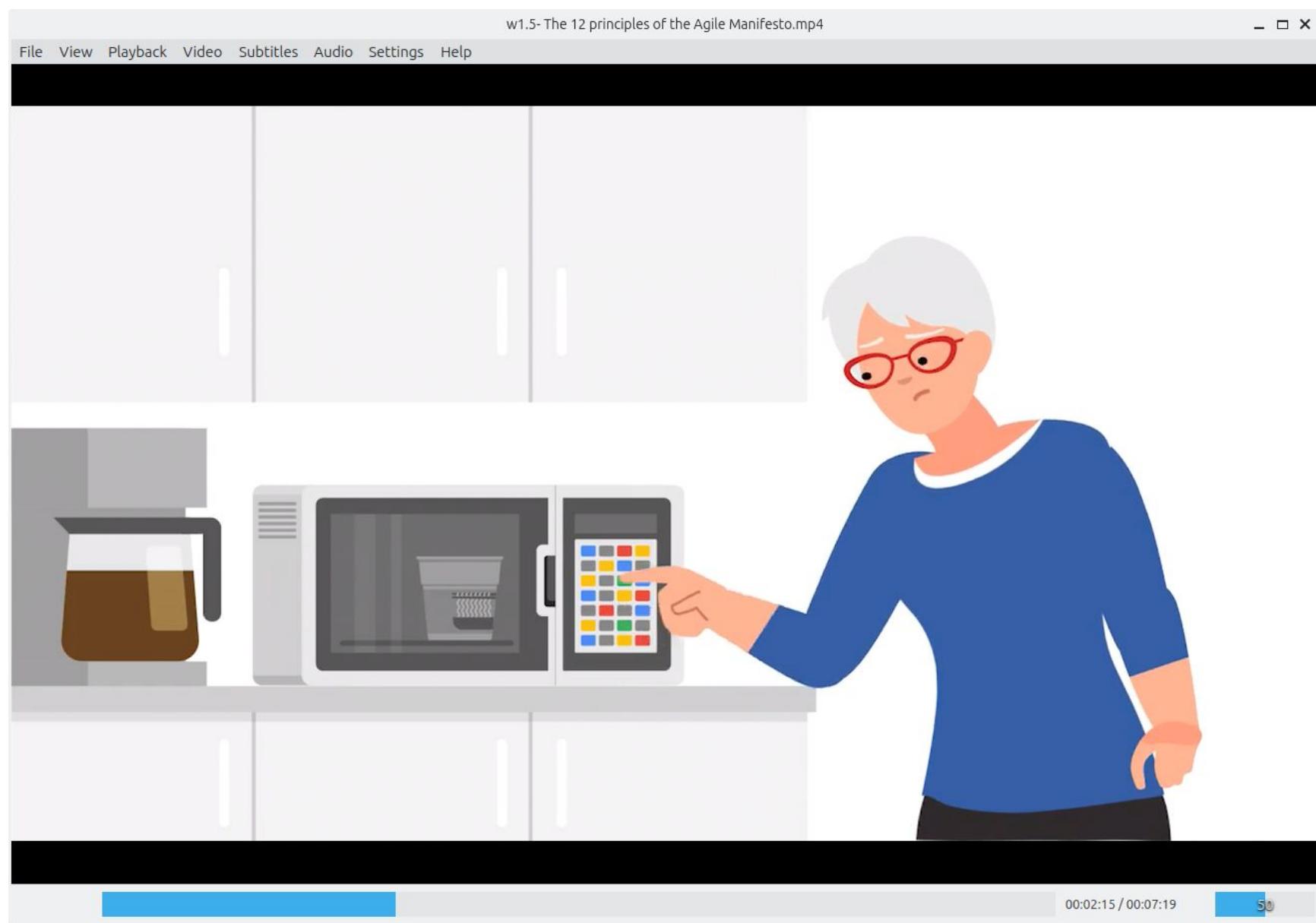
Simplicity—the art of maximizing the amount of work not done—is essential.

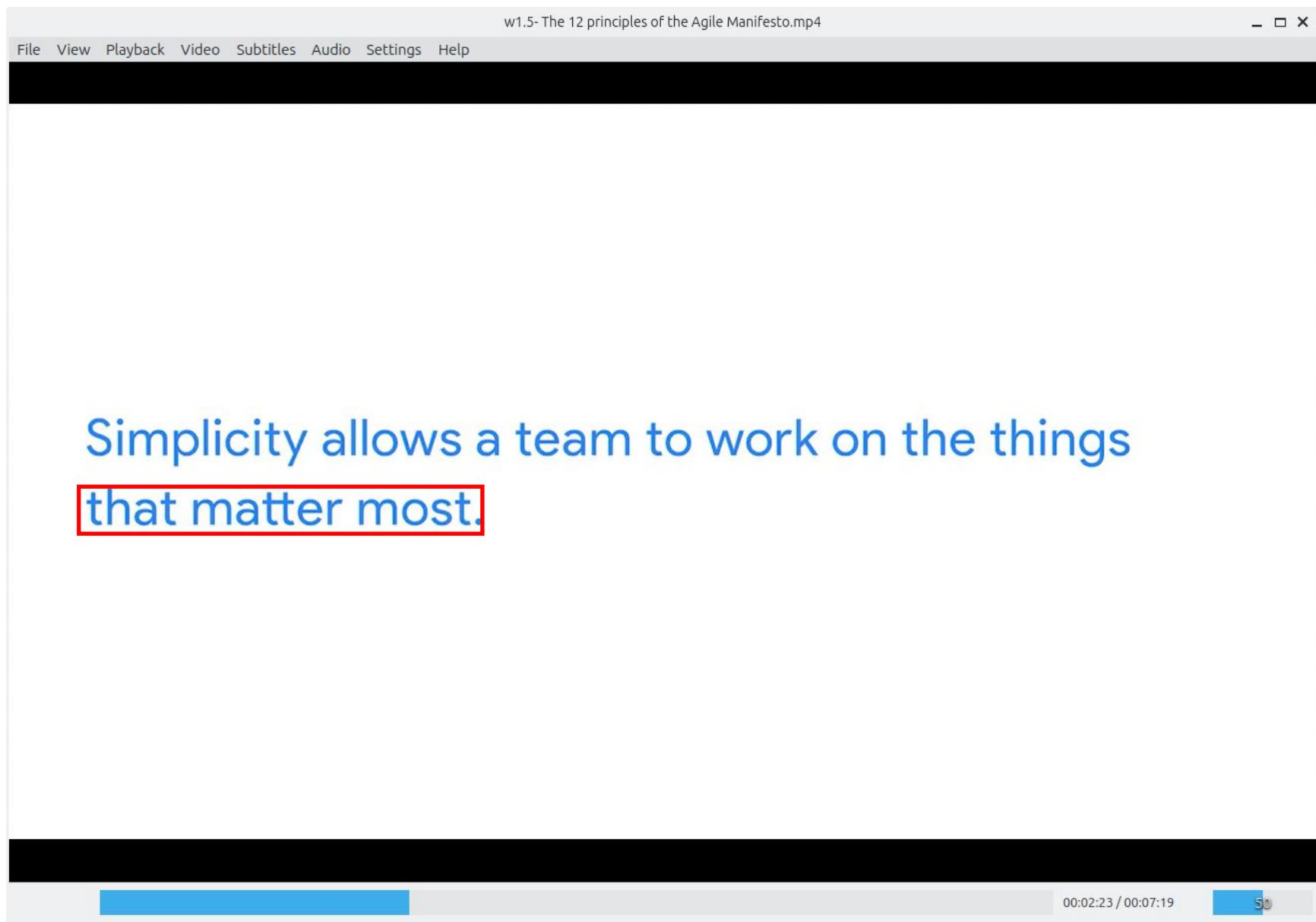
Continuous attention to solution excellence and good design enhances agility.

00:02:02 / 00:07:19 50

Simplicity—the art of maximizing the amount of work not done—is essential."

The goal is to avoid doing unnecessary work, focusing only on what is essential to deliver value.





w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Business people and developers must work together daily throughout the project.

00:02:56 / 00:07:19 50

2. Business Collaboration (2 Principles)

This theme focuses on how the development team (those creating the product) collaborates with **business people** (those involved with sales, marketing, and customer support).

- **Core Idea:** Continuous **customer collaboration** provides critical business information instantly, allowing the team to adapt quickly.
- **Goal:** Enable easy access between business people and developers. This can be achieved by:
 - Co-locating teams (in the same office or virtual space).
 - Encouraging instant messaging.
 - Creating a **weekly huddle** where customers and business people can explore feedback and new ideas with the team without fearing scope creep.

(# Weekly huddle- Recurring meeting)

(# Scope creep- Uncontrolled expansion of project's scope, features)

w1.5- The 12 principles of the Agile Manifesto.mp4

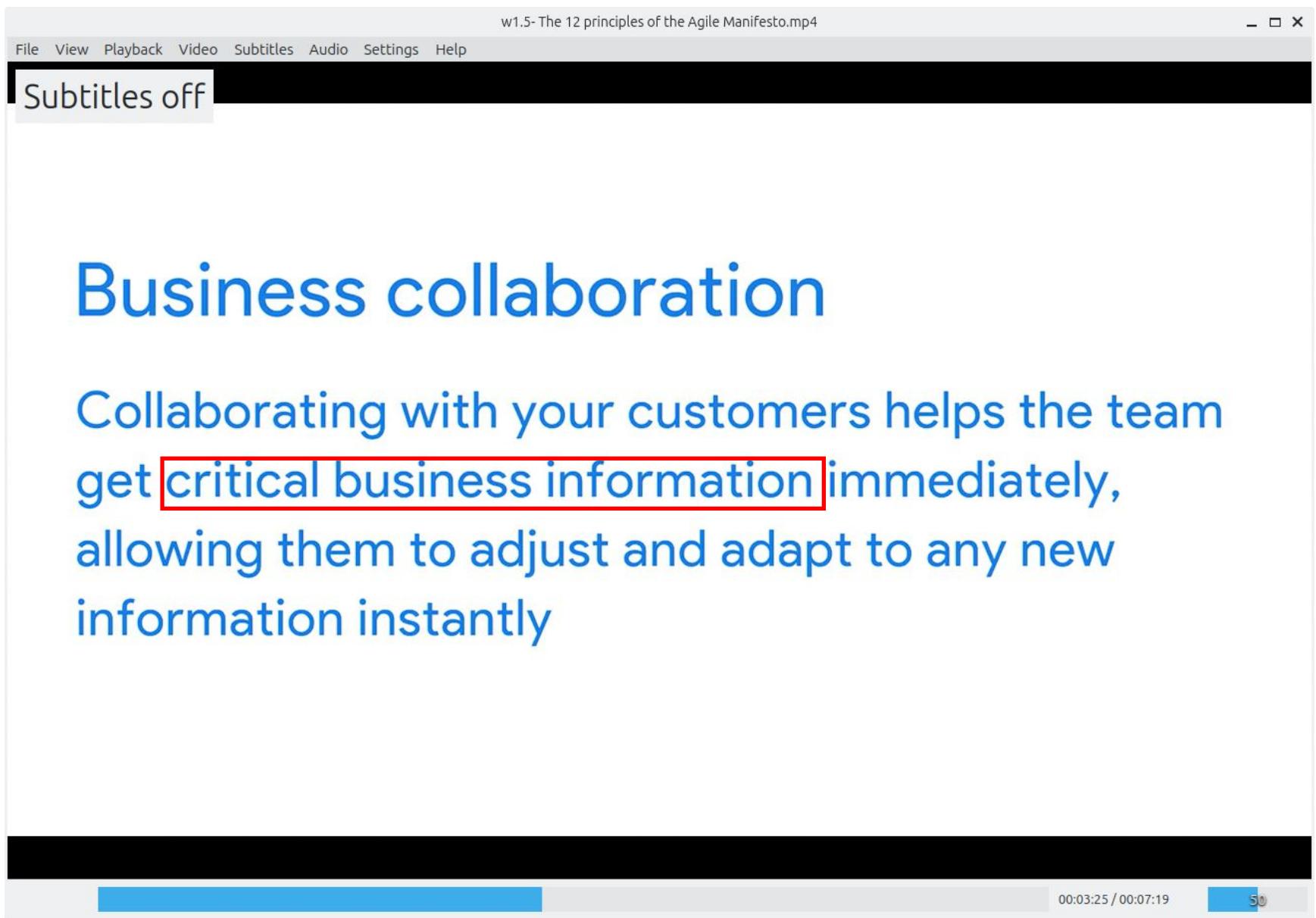
File View Playback Video Subtitles Audio Settings Help

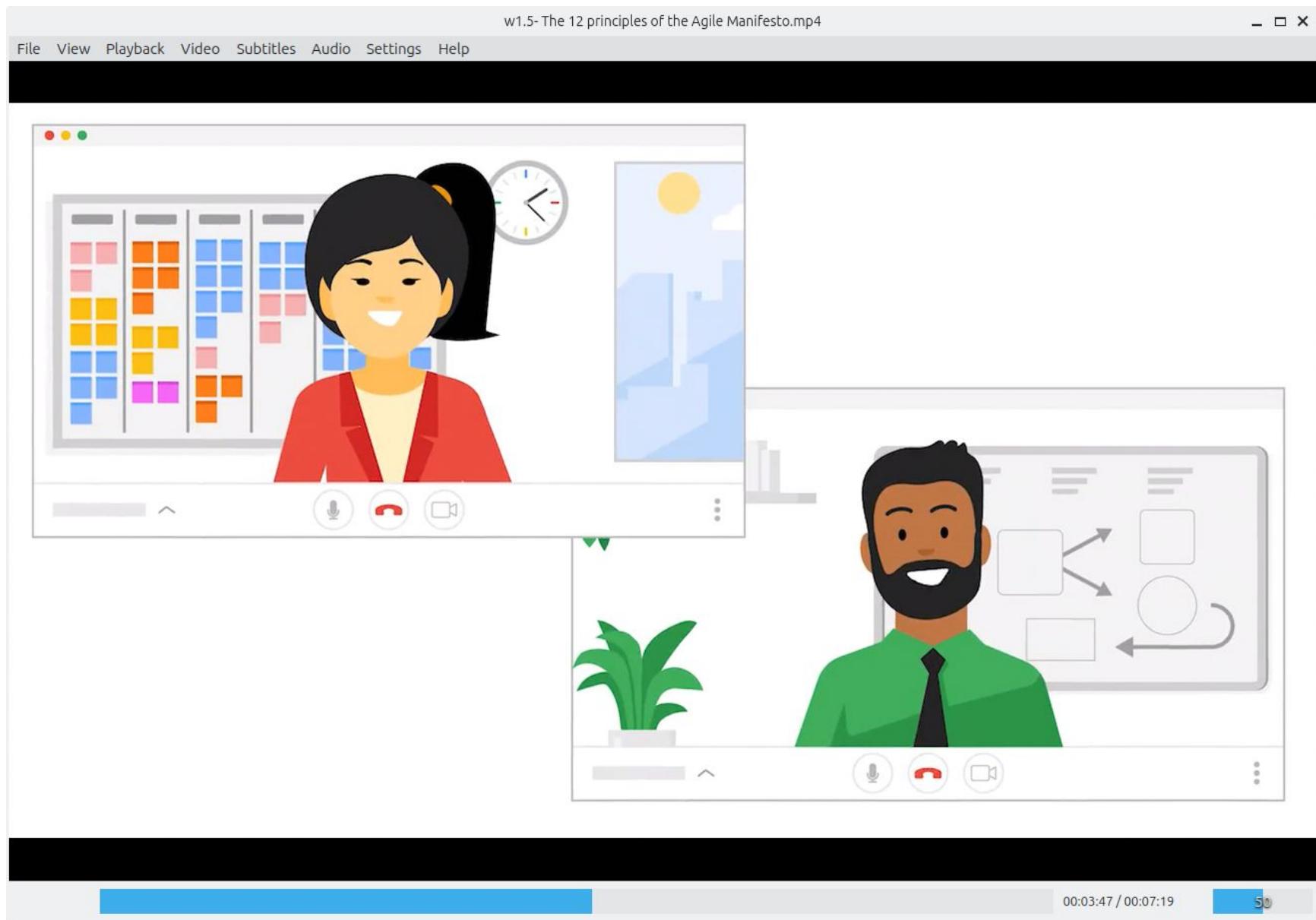
Subtitles off

Business collaboration

Collaborating with your customers helps the team get critical business information immediately, allowing them to adjust and adapt to any new information instantly

00:03:25 / 00:07:19 50





w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Team Dynamics and Culture

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

The best architectures, requirements, and designs emerge from self-organizing teams.

00:04:35 / 00:07:19 50

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Team dynamics and culture

Create an effective team culture that is inclusive,
supportive, and empowering

00:05:02 / 00:07:19

50

Make sure your team

- Is motivated to do the **right thing**
- Feels trusted to do the right thing
- Has the resources and space to work closely together on their goals
- Works at a sustainable pace

3. Team Dynamics and Culture (4 Principles)

Reflecting the first Agile value ("Individuals and interactions over processes and tools"), this theme is about building an **effective team culture** that is inclusive, supportive, and empowering.

- **Core Idea:** An effective team culture is essential for project success. These principles ensure the team is:
 - **Motivated and trusted** to do the right thing.
 - Working at a **sustainable pace**.
 - Equipped with the necessary resources.
- **Example in Action:** Asking the team what equipment they need, allowing teams to write **their own processes and templates**, and making space for their input to be valued. This builds trust and empowers the team to work more productively.

(# empowering- granting authorities)

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

The video player window displays a slide with a yellow vertical bar on the left. The main text on the slide is "Retrospectives and Continuous Learning". To the right of the slide, there is a block of text: "At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly." The word "become more effective" is highlighted with a red rectangular box.

At the bottom of the video player, there is a progress bar showing the time "00:05:58 / 00:07:19" and a volume control icon with the number "50".

At the very top of the slide area, there is a black bar with the text "Subtitles off" in white.

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Retrospectives and continuous learning

Strive to continuously learn and adapt to what's working and what's not

00:06:16 / 00:07:19

50

4. Retrospectives and Continuous Learning (1 Principle)

This theme highlights the importance of regular reflection and adaptation for performance improvement.

- **The Principle:** "At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly."
- **Goal:** Agile teams must continuously learn and adapt. Dedicated time should be set aside after each iteration to focus entirely on improvement, considering questions like:
 - Are the customers happy?
 - Are the processes working?
 - Are we accumulating technical or process "**debt**" (anything that slows the team down)?

w1.5- The 12 principles of the Agile Manifesto.mp4

File View Playback Video Subtitles Audio Settings Help

Questions for improvement

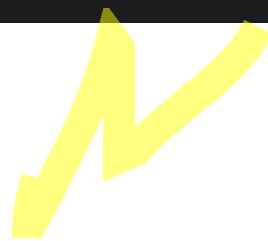
- How is the team doing?
- Are the customers happy?
- Are there processes **we could optimize?** Are our tools working for us?
- Are we following the values?
- Are we accumulating any debt, technical or otherwise?

00:06:47 / 00:07:19 50

The 12 Principles of Agile

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

7. **Working software is the primary measure of progress.** (Focus on Value Delivery)
8. **Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.** (Focus on Team Culture)
9. **Continuous attention to technical excellence and good design enhances agility.** (Focus on Value Delivery)
10. **Simplicity—the art of maximizing the amount of work not done—is essential.** (Focus on Value Delivery)
11. **The best architectures, requirements, and designs emerge from self-organizing teams.** (Focus on Team Culture)
12. **At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.** (Focus on Retrospectives & Continuous Improvement)



w1.6- Adopting an Agile mindset

w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

- Different scenarios in which to adopt an Agile mindset
- Introduction to VUCA

00:00:19 / 00:04:25 50

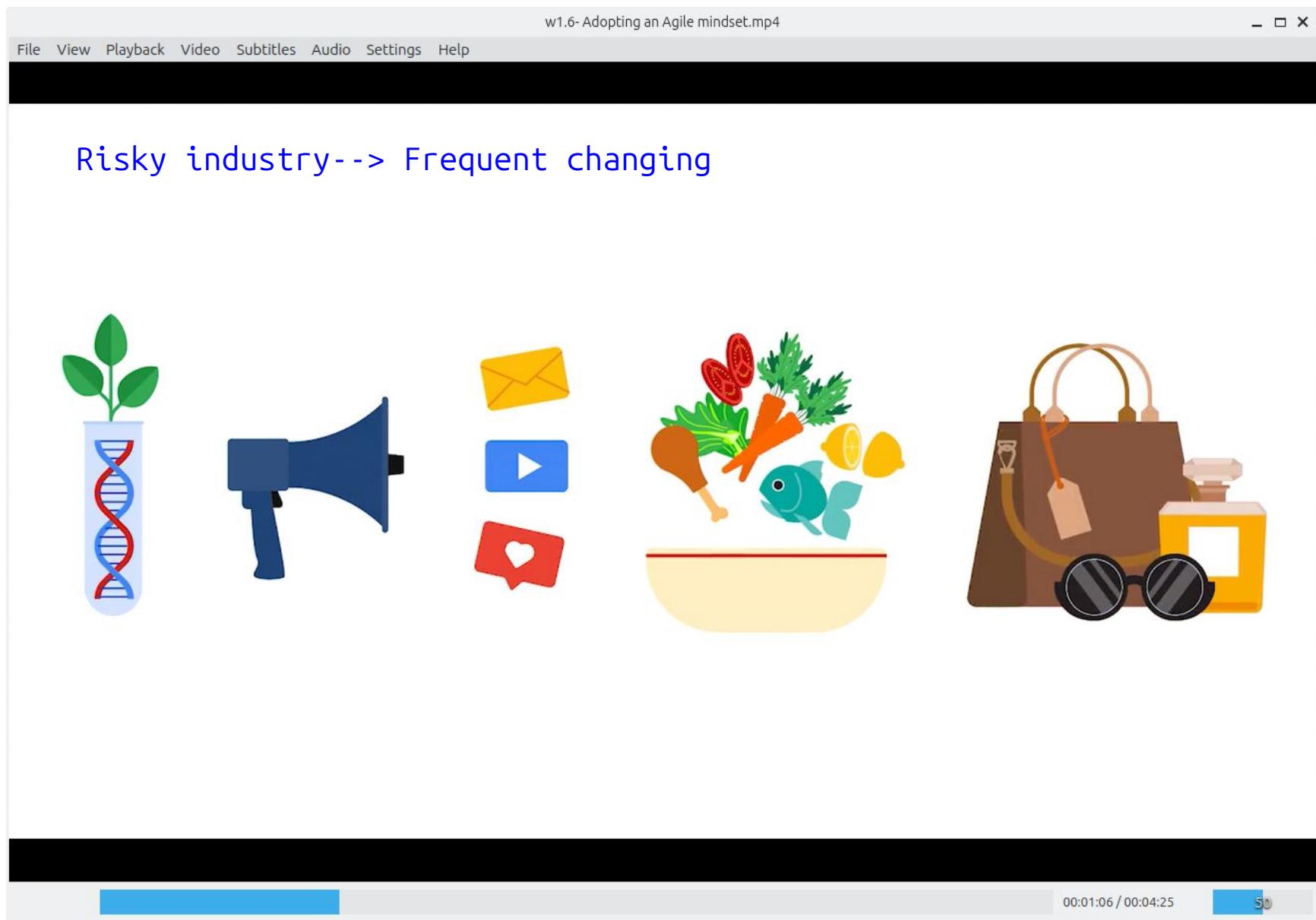
w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

Agile is about delivering value in a world with high degrees of uncertainty, risk, and competition.

Agile works best in industries or projects that are susceptible to or that encourage change and uncertainty.

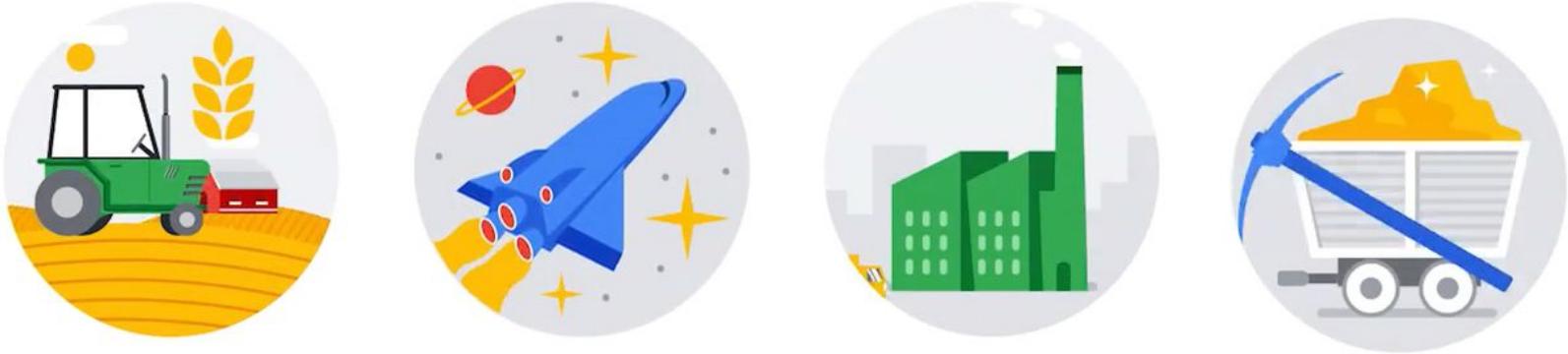
00:00:40 / 00:04:25 50



w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

Stable industries



AGRICULTURE AEROSPACE MANUFACTURING MINING

00:01:20 / 00:04:25 50

- **Examples of industries** (besides software) that deal with lots of change include: **biotechnology** (new vaccines, treatments), **media** (new content sharing methods), the **food industry** (latest crazes), and **fashion** (shifting trends).
- While some industries like agriculture, aerospace, manufacturing, and mining might seem stable, they must still adapt to changes from new laws, regulations, and unforeseen issues.

w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

V U C A

An acronym that defines the conditions that affect organizations in a changing and complex world

00:01:58 / 00:04:25 50

A screenshot of a video player window titled "w1.6- Adopting an Agile mindset.mp4". The window has a menu bar with options: File, View, Playback, Video, Subtitles, Audio, Settings, and Help. A subtitle overlay says "Subtitles off". The main content area displays a large blue acronym "V U C A" followed by a definition: "An acronym that defines the conditions that affect organizations in a changing and complex world". The word "conditions that affect" is highlighted with a red rectangular box. At the bottom of the screen, there is a progress bar showing "00:01:58 / 00:04:25" and a volume control showing "50".



w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

(disruption- added, deleted some features)

Volatility refers to the rate of change and churn in a business or situation.

00:02:26 / 00:04:25 50

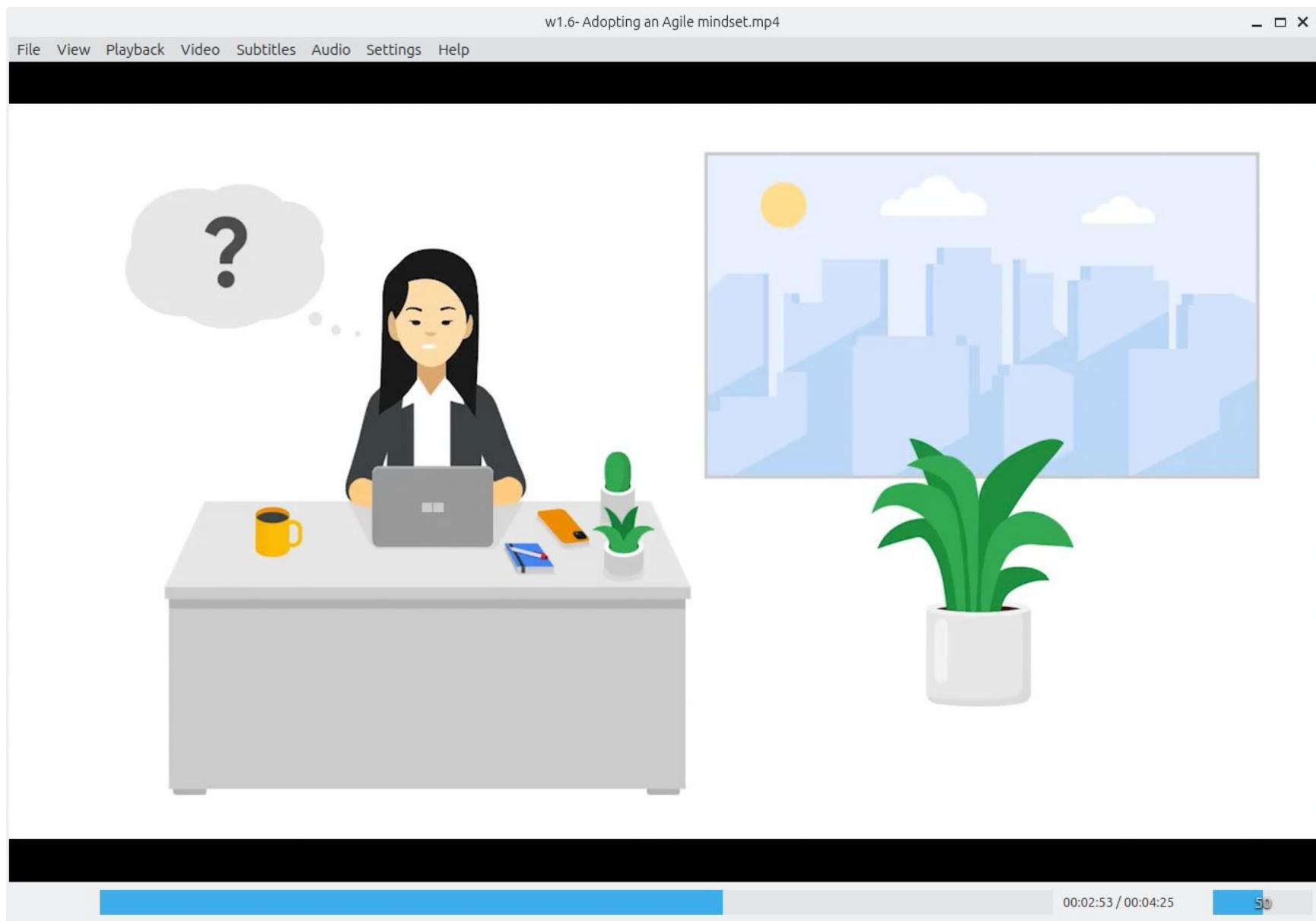
2. Project Management Context (Especially Agile)

In the context of projects, particularly **Agile** development and Scrum, **churn** refers to **unplanned changes** being made to the project scope, task list, or user stories within an iteration or sprint.

- **Agile Churn:** When a team adds, removes, or **changes tasks in a sprint backlog** after the sprint has officially started.
- This is generally undesirable because it disrupts the team's rhythm, affects their ability to meet commitments, and forces them to re-estimate their workload.

The image shows a screenshot of a video player window titled "w1.6- Adopting an Agile mindset.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays a blue text slide: "Uncertainty refers to the lack of predictability or high potential for surprise." The word "lack of predictability" is highlighted with a red rectangular box. The bottom of the window shows a progress bar at 00:02:43 / 00:04:25 and a volume slider set to 50.

Uncertainty refers to the lack of predictability or high potential for surprise.



w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Complexity refers to the high number of interrelated forces, issues, organizations, and factors that would influence the project.

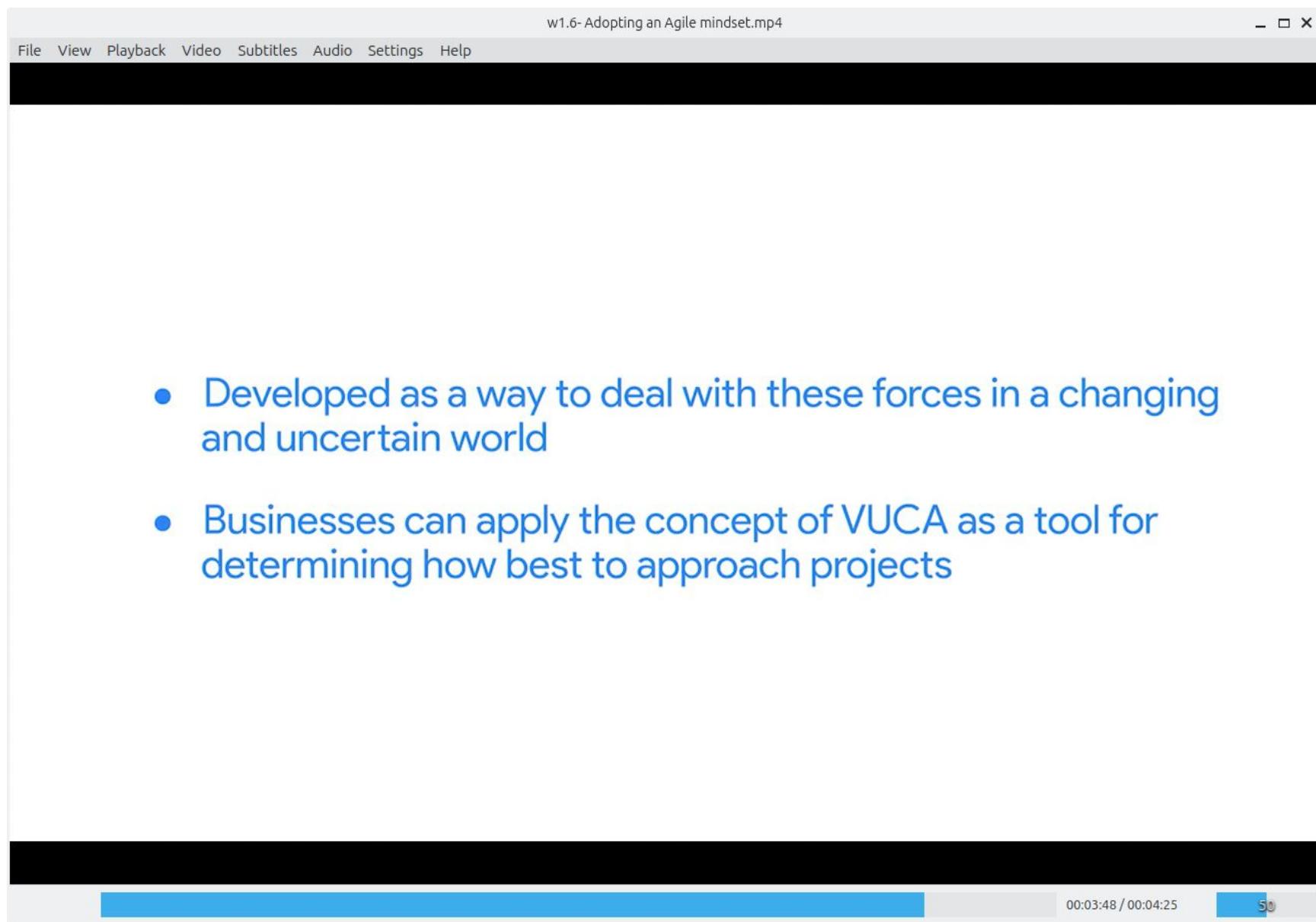
00:02:59 / 00:04:25 50

The image shows a screenshot of a video player window titled "w1.6- Adopting an Agile mindset.mp4". The window has a standard OS X-style interface with a menu bar at the top. The main area displays a large blue text block: "Complexity refers to the high number of interrelated forces, issues, organizations, and factors that would influence the project." A red rectangular box highlights the word "would influence the project.". At the bottom of the window, there is a progress bar showing the video's duration as "00:02:59 / 00:04:25" and a volume slider set to "50".

The screenshot shows a video player interface with a presentation slide overlaid. The video title is "w1.6- Adopting an Agile mindset.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area displays a blue text slide: "Ambiguity refers to the possibility of misunderstanding [the conditions and root causes of events or circumstances.]". A red rectangular box highlights the phrase "the conditions and root causes". The bottom of the screen shows a progress bar at 00:03:17 / 00:04:25 and a volume slider set to 50%.

Ambiguity refers to the possibility of
misunderstanding [the conditions and root causes
of events or circumstances.]

00:03:17 / 00:04:25 50



w1.6- Adopting an Agile mindset.mp4

File View Playback Video Subtitles Audio Settings Help

- Developed as a way to deal with these forces in a changing and uncertain world
- Businesses can apply the concept of VUCA as a tool for determining how best to approach projects

00:03:48 / 00:04:25 50

The **VUCA** framework was developed by the U.S. military to define the conditions that affect organizations in a changing and complex world, helping factor the forces of change and uncertainty into projects and businesses.

VUCA stands for:

- **Volatility:** The **rate of change** and churn in a situation, where disruptions are frequent and things don't settle into a normal rhythm.
- **Uncertainty:** The **lack of predictability** or high potential for surprise, making it difficult to create future plans not based on many assumptions.
- **Complexity:** The **high number of interrelated forces, issues, and factors** influencing a project (e.g., relying on diverse and global suppliers).
- **Ambiguity:** The **possibility of misunderstanding** the conditions and root causes of events, making it difficult to pinpoint causes of delays and design mitigation plans.

w1.7- Applying Agile in a VUCA environment

w1.7- Applying Agile in a VUCA environment.mp4

File View Playback Video Subtitles Audio Settings Help

- When starting on a new project, it's helpful to examine the environment and conditions in which the project exists before deciding the best approach to use
- If your project has high levels of VUCA parameters, it's a good sign you should consider an Agile approach
- An Agile approach will lead to better outcomes by giving you and your team tools and systems to mitigate VUCA risks

00:00:53 / 00:03:36 50

w1.7- Applying Agile in a VUCA environment.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off



The logo features three stylized green plants of increasing height from left to right. Below the plants, the word "Office" is written in a bold, dark gray sans-serif font, and "Green" is written in a slightly smaller, bold, dark gray sans-serif font. A horizontal green line extends from the bottom of the "O" in "Office" to the end of the "e" in "Green".

00:01:29 / 00:03:36

50

w1.7- Applying Agile in a VUCA environment.mp4

File View Playback Video Subtitles Audio Settings Help

The illustration depicts a person with dark skin and short hair, wearing a blue t-shirt, sitting at a wooden desk. They are smiling and looking towards the camera. On the desk, there is a laptop, a red mug, and a small orange cat curled up on top of a brown wooden cabinet. To the right of the desk, a large green plant in a blue pot labeled 'Office Green' is visible. In the background, there is a window showing a green landscape with trees and a building. A shelf above the desk holds several books and a small potted plant. The overall theme is a modern, eco-friendly office environment.

They wanted to react fast to
a potentially huge market opportunity and

00:01:47 / 00:03:36 50

w1.7- Applying Agile in a VUCA environment.mp4

File View Playback Video Subtitles Audio Settings Help



Virtual
Verde

Your goal is to deliver their new service, called Virtual Verde.

00:02:39 / 00:03:36 50

w1.7- Applying Agile in a VUCA environment.mp4

File View Playback Video Subtitles Audio Settings Help

- Volatility
 - Major and abrupt change **to business plans**
- Uncertainty
 - Lack of predictability; difficult to create concrete plans for the future
- Complexity
 - Interrelated factors
- Ambiguity
 - Not being able to determine or control changes

00:03:05 / 00:03:36 50

The Office Green Scenario: Virtual Verde

Office Green LLC, a commercial landscaping company specializing in plant design for offices, noticed a major market shift as more workers transitioned to **home offices**.

- To react fast and avoid losing revenue from their old service model, Office Green decided to pivot and capture the new home office market by creating a new service called **Virtual Verde**.
- This sudden market shift meant they had no time for extensive prep work or creating concrete plans. They needed to maximize the opportunity quickly.
- The user is assigned as the project manager for the new Agile team tasked with delivering **Virtual Verde**.

How VUCA Applied to Office Green

The new project environment for Virtual Verde was clearly defined by high VUCA factors:

- **Volatility:** A major, disruptive change to their core business model (commercial landscaping).
- **Uncertainty:** A lack of predictability, making it difficult to create concrete future plans.
- **Complexity:** High levels of interrelated external factors like global suppliers and the overall economy. ↗
- **Ambiguity:** Inability to determine or control what might cause future changes and their effects.

By adopting an **Agile approach**, Office Green was able to **embrace the changing market** and remain flexible, addressing the high VUCA factors affecting their business instead of being eroded by them. The course will continue to follow the progress of the Virtual Verde project.

w1.8- Introduction to Scrum

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

● Origins of Scrum

● Basics of Scrum methodology

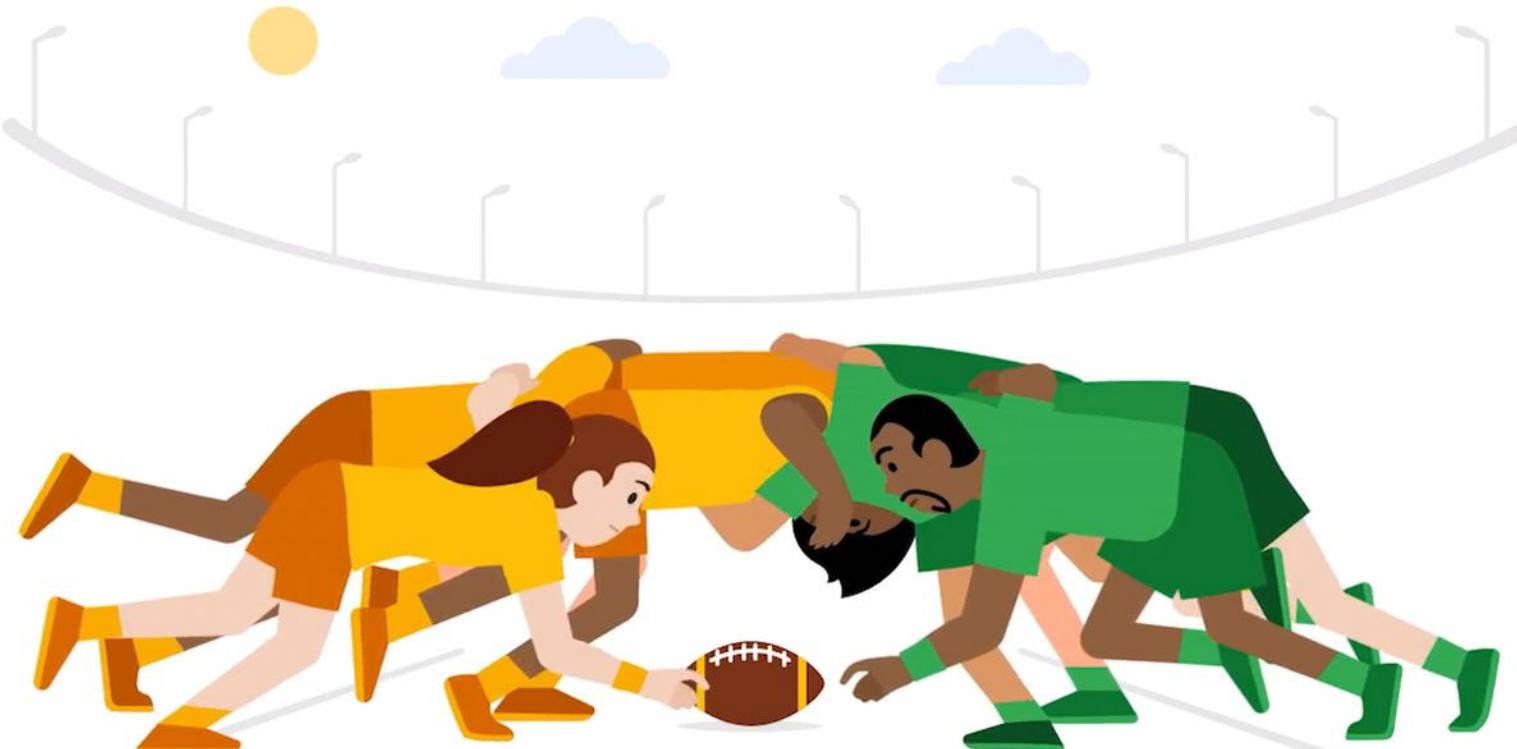
00:00:25 / 00:05:17

65

A screenshot of a video player window titled "w1.8- Introduction to Scrum.mp4". The window has a dark header bar with menu options: File, View, Playback, Video, Subtitles, Audio, Settings, and Help. A black bar at the top of the video frame contains the text "Subtitles off". On the left side of the video frame, there is a list of bullet points: "● Origins of Scrum" and "● Basics of Scrum methodology". The "Origins of Scrum" point is highlighted with a red rectangular box. The video frame shows a woman with blonde hair, wearing a green top and a necklace, speaking. The bottom of the video frame has a progress bar showing "00:00:25 / 00:05:17" and a page number "65" on the right.

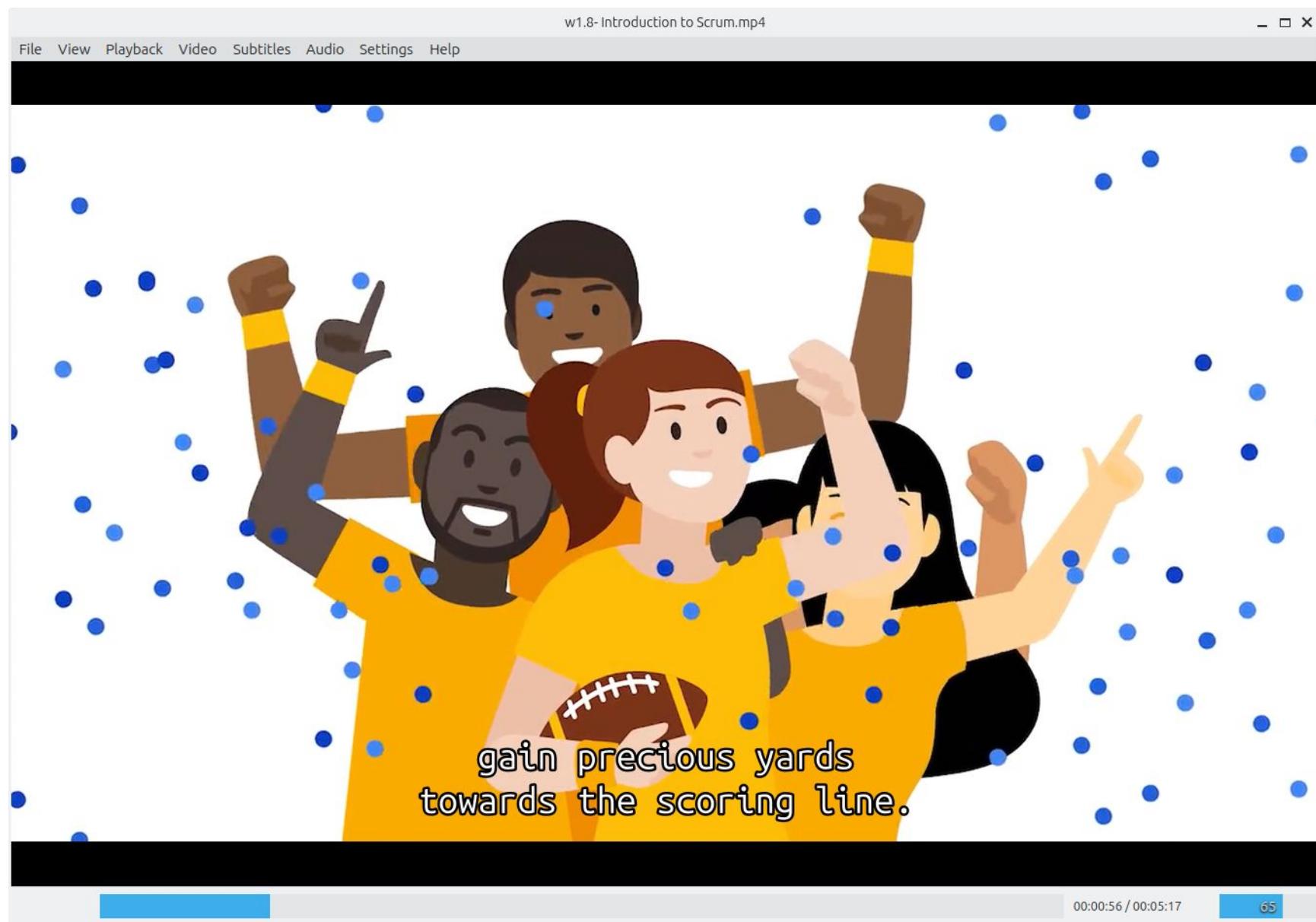
w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help



Scrum refers to a formation in rugby where all of the players on the team lean

00:00:49 / 00:05:17 65



Scrum: The Most Popular Agile Methodology

The video introduces **Scrum** as the most popular Agile methodology, noting that it is **not an acronym** but is named after the close-knit **rugby formation** where players work as one unit to move the ball forward. Scrum is highly used, with a 2019 report showing 72% of teams using Agile utilized Scrum or a hybrid approach. 

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

AGILE METHODS AND PRACTICES

AGILE METHODOLOGIES USED

Scrum and related variants continue to be the most common Agile methodologies used by respondents' organizations.

A pie chart illustrating the distribution of Agile methodologies used by respondents' organizations. The chart shows the following percentages:

Methodology	Percentage
Scrum	58%
ScrumBan	10%
Other/Hybrid/Multiple Methodologies	9%
Scrum/XP hybrid	8%
Kanban	7%
Iterative Development	4%
Don't Know	3%
Lean Startup	1%
Extreme Programming (XP)	1%

Total exceeds 100% due to rounding.

The screenshot shows a video player interface with the title "w1.8- Introduction to Scrum.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. A black bar at the top has the text "Subtitles off". The main content area displays the text "Product Backlog" in large blue letters, followed by a definition in blue text: "The central artifact in Scrum, where all possible ideas, deliverables, features, or tasks are captured for the team to work on". A red box highlights the word "central artifact". The bottom of the screen shows a black bar, a blue progress bar, and a status bar with "00:01:54 / 00:05:17" and a page number "75".

Subtitles off

Product Backlog

The central artifact in Scrum, where all possible ideas, deliverables, features, or tasks are captured for the team to work on

00:01:54 / 00:05:17 75

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Sprint

A time-boxed iteration in Scrum where work is done

00:02:11 / 00:05:17 75

The screenshot shows a video player interface with a slide about Daily Scrum. At the top, the title bar reads "w1.8- Introduction to Scrum.mp4". Below it is a menu bar with "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". A black bar covers the "Subtitles" and "Help" menu items. The main content area features a large blue header "Daily Scrum" and a blue text definition: "A meeting of 15 or fewer minutes every day of the Sprint". The word "15" is highlighted with a red box. The bottom of the screen shows a black bar and a progress bar indicating the video is at 00:02:25 / 00:05:17, with the number "75" in a blue box.

Subtitles off

Daily Scrum

A meeting of 15 or fewer minutes every day of the Sprint

00:02:25 / 00:05:17 75

Core Concepts, Artifact, and Event

Scrum uses short cycles to quickly develop and test deliverables. 

- **Backlog:** The central artifact. It is a prioritized list of all possible ideas, deliverables, features, or tasks for the team to work on. It is continuously managed by the **Product Owner**.
- **Sprint (or Iteration):** The time-boxed period during which work is done, typically lasting **one to four weeks**, with two weeks being most common. 
- **Daily Scrum (or Stand-up)** **A 15-minute or less** meeting held every day of the Sprint for the team to inspect their progress toward the Sprint goal. 

Item Type	Item Description	Estimated Size (Story Points)	Business Value
Feature	As a user, I want to securely log in using Face ID/Touch ID so that I can access my accounts quickly without typing a password.	3	High
Bug Fix	Fix critical error where account balance sometimes displays as zero immediately after a transfer is completed.	5	Very High
Feature	As a user, I want to view the last 5 transactions on my main dashboard without navigating to the history page.	2	High
Technical Debt	Refactor the legacy payment processing API to improve transaction speed by 15%.	8	Medium-High

The term "**central artifact**" is most frequently used in the context of **Agile project management**, particularly the **Scrum methodology**.

It refers to a key document or piece of information that is essential for the team, stakeholders, and the project as a whole, as it represents the work to be done, the value being created, or the progress achieved.

In the Context of Scrum (Agile)

In Scrum, the word "artifact" is used for crucial pieces of information that ensure **transparency, inspection, and adaptation**. When a speaker refers to **the central artifact**, they are usually highlighting one of the three main Scrum Artifacts, with the **Product Backlog** often being considered the *most central* or foundational:

Artifact	Centrality/Meaning
1. Product Backlog	Often referred to as the "central artifact" in the sense that it is the single source of truth for all planned work. It is a prioritized, constantly evolving list of features, requirements, enhancements, and fixes needed for the product.
2. Sprint Backlog	The list of items from the Product Backlog selected for development during a single Sprint, plus the plan for delivering the Increment. It's central to the current iteration's work.
3. Increment	The usable, valuable, and potentially releasable product resulting from the completed work of a Sprint, plus the sum of all previous Increments. It is central to the value being delivered.

In General Project Management

In a broader sense, a **central artifact** is any **key document, deliverable, or output** created during a project's lifecycle that guides the work, tracks progress, and aligns the team with business goals.

Examples of central artifacts in general project management include:

- **Project Charter:** The foundational document that formally authorizes the project and defines its high-level scope and objectives.
- **Project Plan:** The comprehensive roadmap detailing how the project will be executed, monitored, and controlled.
- **Requirements Documentation/Log:** The definitive source detailing all the necessary functions and features of the final product.

In both contexts, "**central artifact**" means a piece of documentation or information that is **essential for communication, decision-making, and tracking the core work of the project.**

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help



Scrum Master

- Responsible for ensuring the team lives Agile values and principles
- Responsible for ensuring the team follows the processes and practices that the team agreed to
- Responsible for sharing information to the larger project team
- **Responsible for helping the team focus on doing their best work**

Development Team

- Responsible for how a team will deliver that product

Product Owner

- Responsible for maximizing the **value of the product** and the work of the team
- Responsible for the inventory of work and has final say on how to prioritize the work

00:03:11 / 00:05:17

75

Key Roles

Scrum defines three distinct roles while emphasizing the power of the team as a whole:



- **Scrum Master:** Responsible for ensuring the team adheres to Agile values and principles, follows agreed-upon processes, and focuses on doing their best work. They are a *servant-leader*.
- **Product Owner:** Responsible for **maximizing the value** of the product and the work of the team. They own the Backlog and have the final say on prioritizing the work.
- **Development Team:** Responsible for *how* the product is delivered.

A **Servant Leader** in Scrum is a leader who prioritizes the needs of the **team** and the **Product Owner** over their own authority. This leadership style, most notably embodied by the **Scrum Master**, is centered on **support, enablement, and facilitation** rather than command and control.

The core purpose of a Servant Leader in Scrum is to **maximize the ability of the Development Team to deliver value** by focusing on these areas:

- **Serving the Team:** They help the Development Team organize and manage itself. They believe in the team's expertise and focus on removing obstacles so the team can concentrate on development work.
- **Serving the Product Owner:** They assist the Product Owner in defining the product goal, managing the **Product Backlog**, and practicing empirical product planning (making decisions based on real-world data).
- **Impediment Removal**: They actively work to **remove roadblocks** (technical, organizational, or interpersonal) that hinder the team's progress. They shield the team from external distractions and interferences.
- **Coaching and Teaching:** They coach the team and the organization on the practices, theory, and values of Scrum and Agile principles.

The screenshot shows a video player window titled "w1.8- Introduction to Scrum.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays a large blue title "Reasons for Scrum's popularity" followed by a bulleted list of reasons. The fourth reason is highlighted with a red box around the first part of the sentence. The video player has a black bar at the bottom and a progress bar showing "00:03:45 / 00:05:17". The page number "75" is in the bottom right corner.

Reasons for Scrum's popularity

- Clear roles and responsibilities, while continuously emphasizing the power of the team as a whole
- Regular and predictable meeting and delivery schedules, formats, and outcomes
- Supports and reinforces the Agile values and principles, while adding structure and foundations that help new Agile teams get started and more experienced teams get better
- Free and open for all to use. Huge amounts of online guidance and support, as well as Scrum-specific training and certifications.

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help

a Scrum team should be cross-functional,
with around three to nine team members.

00:04:04 / 00:05:17 75

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help

(# 3 to 9 members)

Some call this a "pizza-size team" because it has the same amount of

00:04:08 / 00:05:17

75

w1.8- Introduction to Scrum.mp4

File View Playback Video Subtitles Audio Settings Help

A cartoon illustration of a team meeting. A woman in a red jacket stands behind a grey conference table, gesturing towards a whiteboard that displays a bar chart with four bars of increasing height in yellow, green, blue, and red. Five team members are seated around the table: a man in a green shirt, a woman in a yellow top, a man in a blue shirt, a woman in a blue shirt, and a man in a red shirt. The man in the red shirt is leaning forward, pointing at the whiteboard. A laptop is open on the table. The scene is set in a modern office environment.

Lastly, Scrum works best for projects where the team and management are open-minded,

00:04:27 / 00:05:17

75

Benefits and Team Characteristics

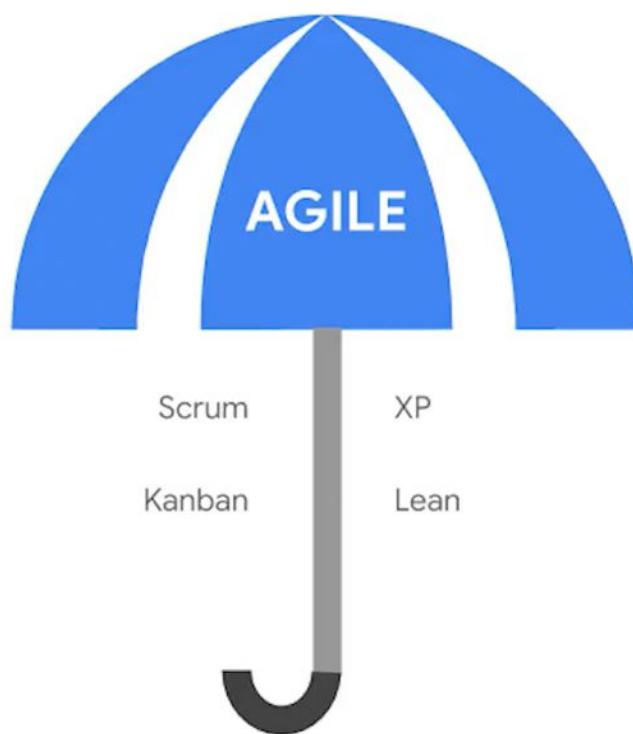
Scrum is popular because it provides structure and clear expectations while reinforcing Agile values. 

- **Benefits:** Clear roles, regular and predictable meetings, predefined agendas, ease of teaching, reinforcement of Agile principles, and free/open-to-use with a large amount of support and training available.
- **Ideal Team Size:** Scrum teams should be **cross-functional** and small, ideally **three to nine members** (a "pizza-size team"). 
- **Adaptability:** Although it originated in software development, Scrum can be applied to a wide variety of projects, **including wedding planning** and house moves, but only if the team and management are **open-minded, adaptable, and value continuous learning**. 



w1.9- Introduction to Kanban, XP, and Lean

Subtitles off



w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Kanban

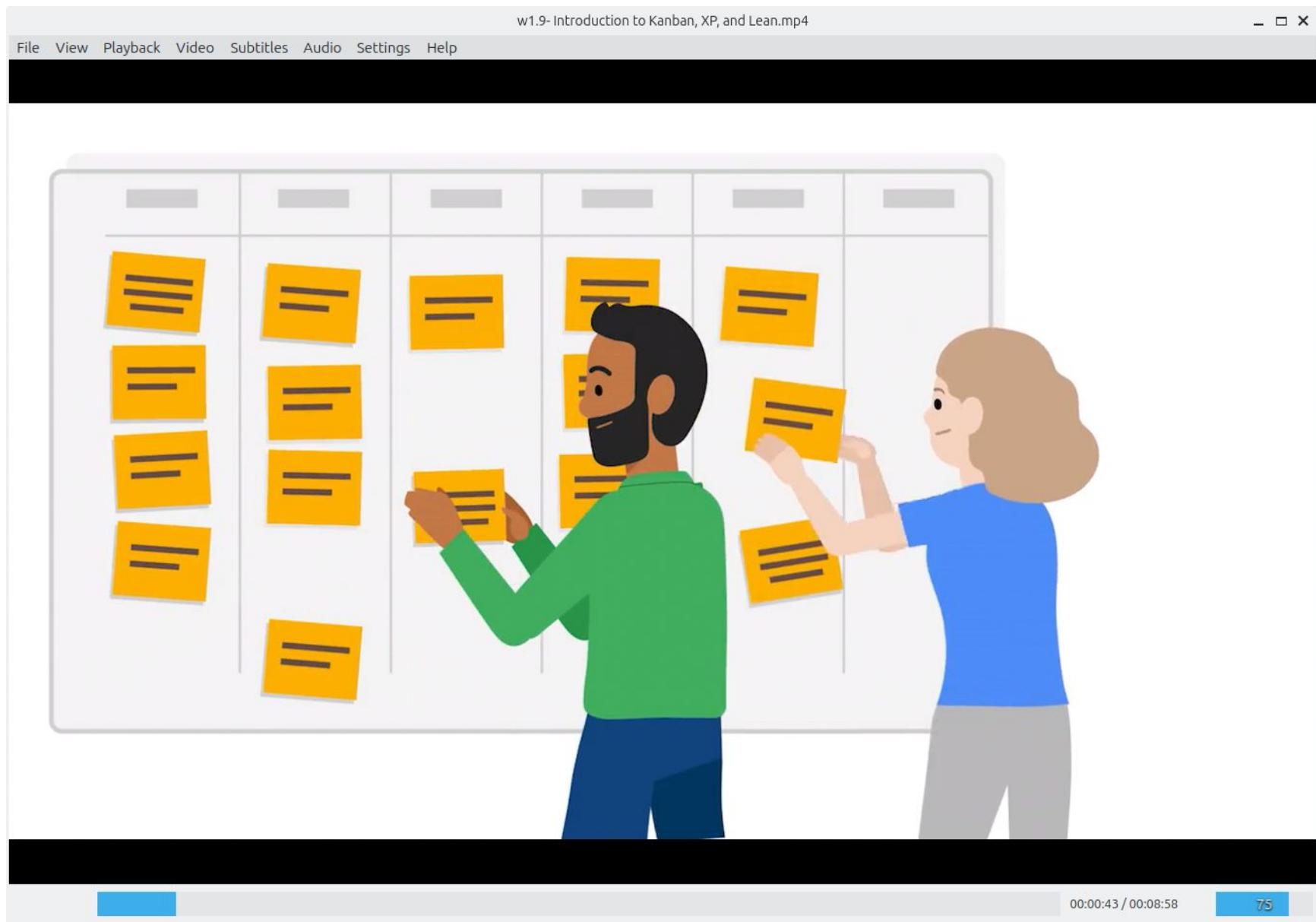
From Japanese

Kan 看 = “sign”

Ban 板 = “board”

00:00:37 / 00:08:58

75



w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Benefits of Kanban

- Provides transparent visual feedback
- Ensures that the project team only accepts a sustainable amount of in progress work

00:01:13 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Work-in-Progress (WIP) Limit

Tasks are limited to what the team can actually handle during a certain amount of time.

00:01:23 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

BACKLOG	CREATING (WIP:8)	REVIEW (WIP:4)	PUBLISHING (WIP:4)	DONE
2 yellow	2 orange	2 yellow		1 yellow
2 blue	2 yellow	2 orange		
	2 yellow	2 yellow		
	2 blue	2 blue		
	2 red	2 red		

The team members add new tasks to be completed only after they finished their

00:01:40 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

BACKLOG

CREATING (WIP:8)

REVIEW (WIP:4)

PUBLISHING (WIP:4)

DONE

previous task and
are below the WIP limit.

00:01:45 / 00:08:58

75

The image shows a screenshot of a video player window titled "w1.9- Introduction to Kanban, XP, and Lean.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. A subtitle bar at the top says "Subtitles off". The main content area displays a slide with the word "Flow" in large blue letters. Below it, a statement reads "A core principle of Kanban that aims to maximize efficiency", where "efficiency" is highlighted with a red border. At the bottom of the slide, there is a black horizontal bar. The video player interface at the bottom shows a blue progress bar, the text "00:01:59 / 00:08:58", and a page number "75".

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Flow

A core principle of Kanban that aims to maximize
efficiency

00:01:59 / 00:08:58 75

1. Kanban

Kanban (Japanese for "sign board") is a methodology that focuses on **visual transparency** and **limiting work in progress**.

- **Kanban Board:** The most famous feature, it provides visual feedback on work status, typically displayed in columns like "To Do," "In Progress," and "Done."
- **Work-in-Progress (WIP) Limit:** The core principle. The team decides on a sustainable amount of work they can handle at any given time and accepts new tasks only after finishing a previous one.
- **Flow:** The primary goal is to **maximize efficiency** by focusing on less work, thereby getting tasks done faster. Once a task is started, the entire team prioritizes getting it to "Done."

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

Extreme
Programming
(XP)

00:02:08 / 00:08:58

75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

(#Test first development)

Extreme Programming (XP)

- Aims to improve product quality and the ability to respond to changing customer needs
- Takes best practices for the development process to "extreme" levels

00:02:54 / 00:08:58 75

2. Extreme Programming (XP)

XP originated in the software industry and aims to improve **product quality and responsiveness** to customer needs by taking best development practices to an "extreme" level. It is applicable to non-software environments as well.

The screenshot shows a video player window titled "w1.9- Introduction to Kanban, XP, and Lean.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area displays a large blue title "XP activities" followed by a bulleted list: "• Designing". At the bottom, there is a black redacted bar, a blue progress bar, and a status bar showing "00:03:38 / 00:08:58" and a page number "75".

XP activities

- Designing

00:03:38 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

The illustration shows a man with orange hair and glasses, wearing a blue polo shirt, standing next to a large grey frame. Inside the frame, there is a graphic design for a soda advertisement. The design includes the word 'SODA' in green, a bottle, a glass, and several orange slices. Below the main image are wireframe versions of a website and a mobile phone screen. The man has his hand to his chin, looking thoughtful.

maybe the main pieces are the artwork,
the copy and the ad by plan.

00:03:52 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

XP wants to ensure that all of the pieces of the product will fit together properly, so it stresses simplicity.

00:04:00 / 00:08:58 75

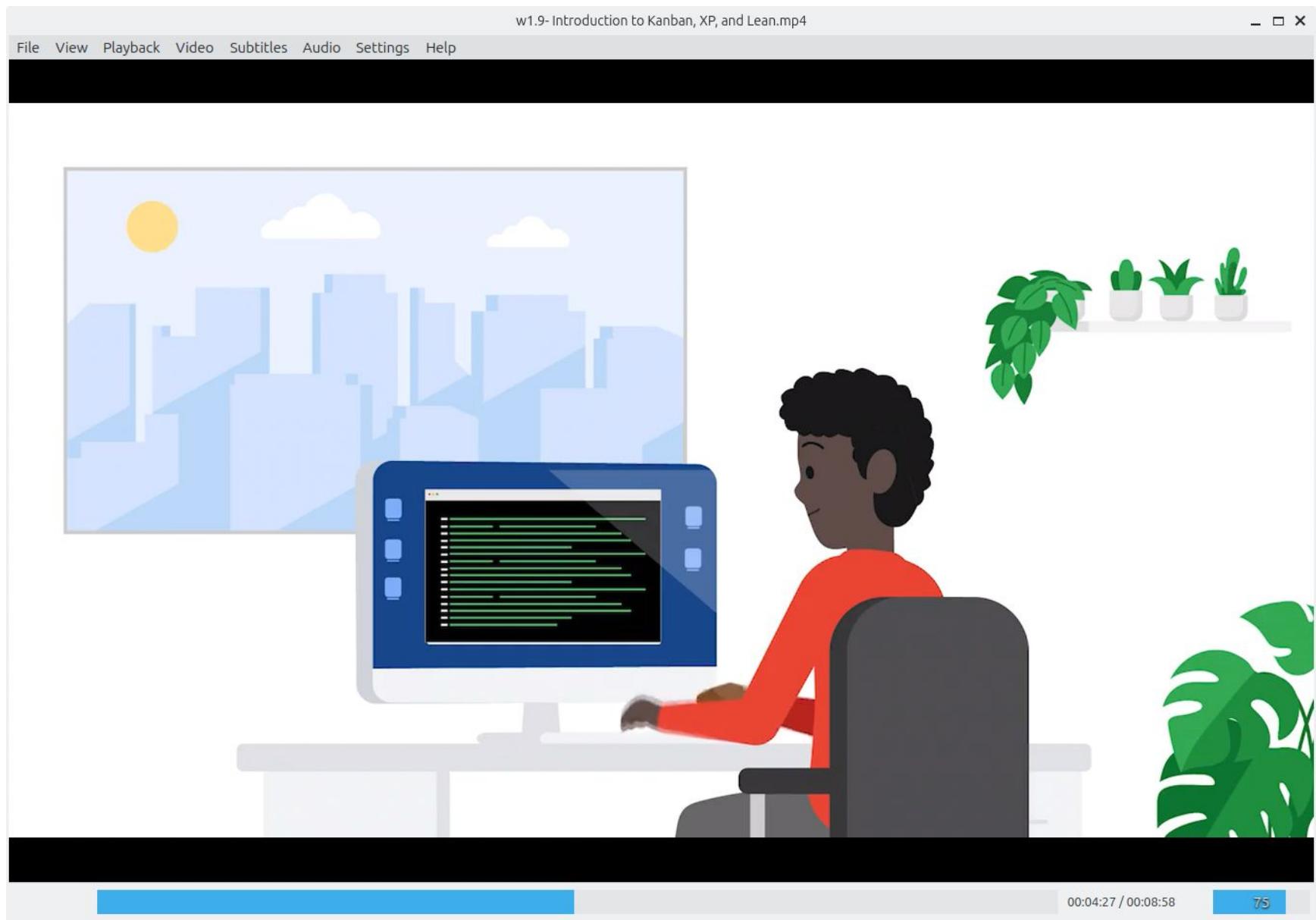
The image shows a screenshot of a video player window titled "w1.9- Introduction to Kanban, XP, and Lean.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays a slide with the following text: "XP wants to ensure that all of the pieces of the product will fit together properly, so it stresses simplicity." The words "so it stresses" and "simplicity." are highlighted with red rectangular boxes. At the bottom of the screen, there is a progress bar indicating the video is at 00:04:00 of 00:08:58, and a page number "75" in a blue box.

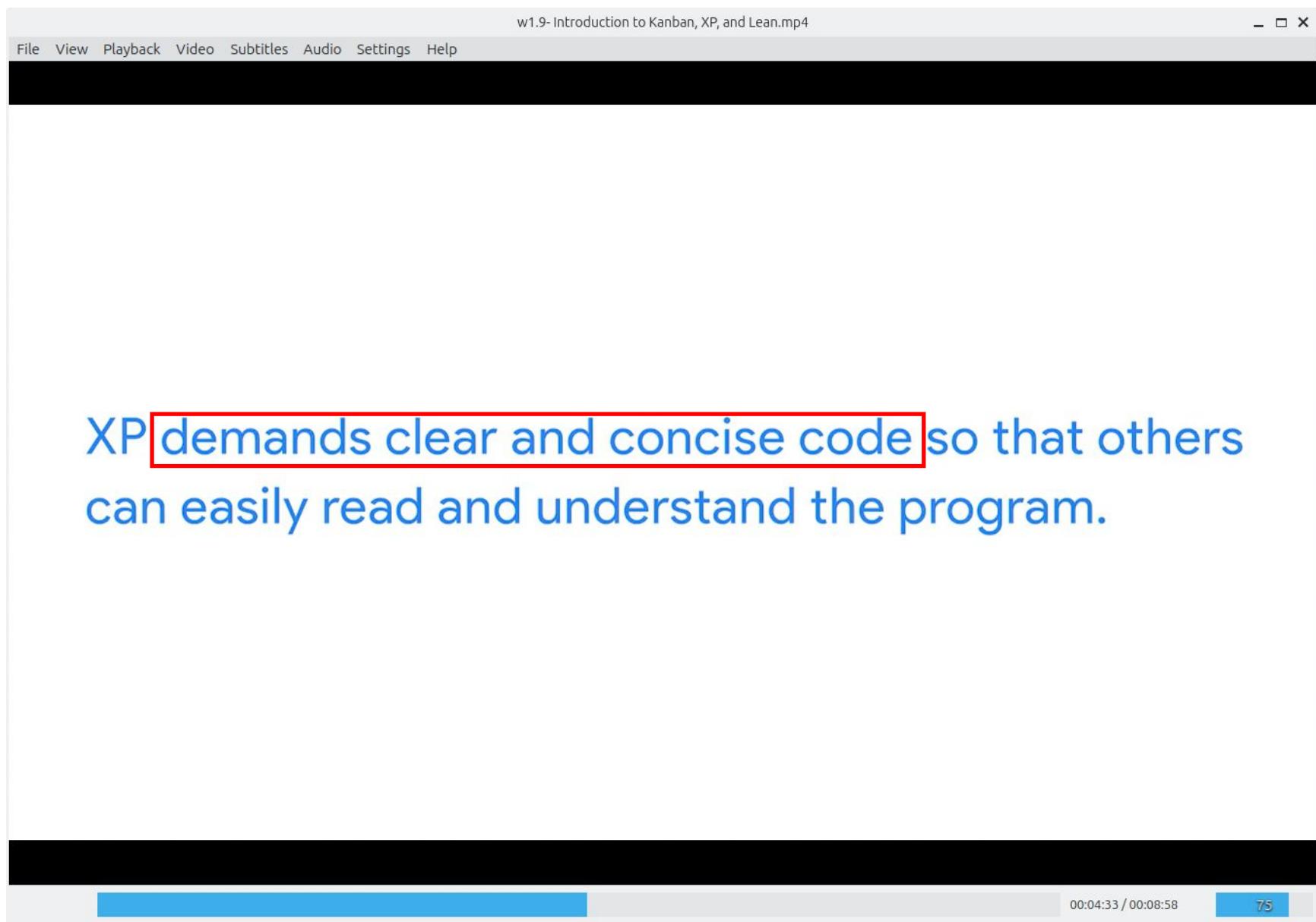
The image shows a screenshot of a video player window titled "w1.9- Introduction to Kanban, XP, and Lean.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area displays a slide with the title "XP activities" in large blue text, followed by a bulleted list: "• Designing" and "• Coding". A black redaction bar covers the bottom portion of the slide content. At the bottom of the player window, there is a progress bar, a timestamp "00:04:19 / 00:08:58", and a page number "75".

XP activities

- Designing
- Coding

00:04:19 / 00:08:58 75





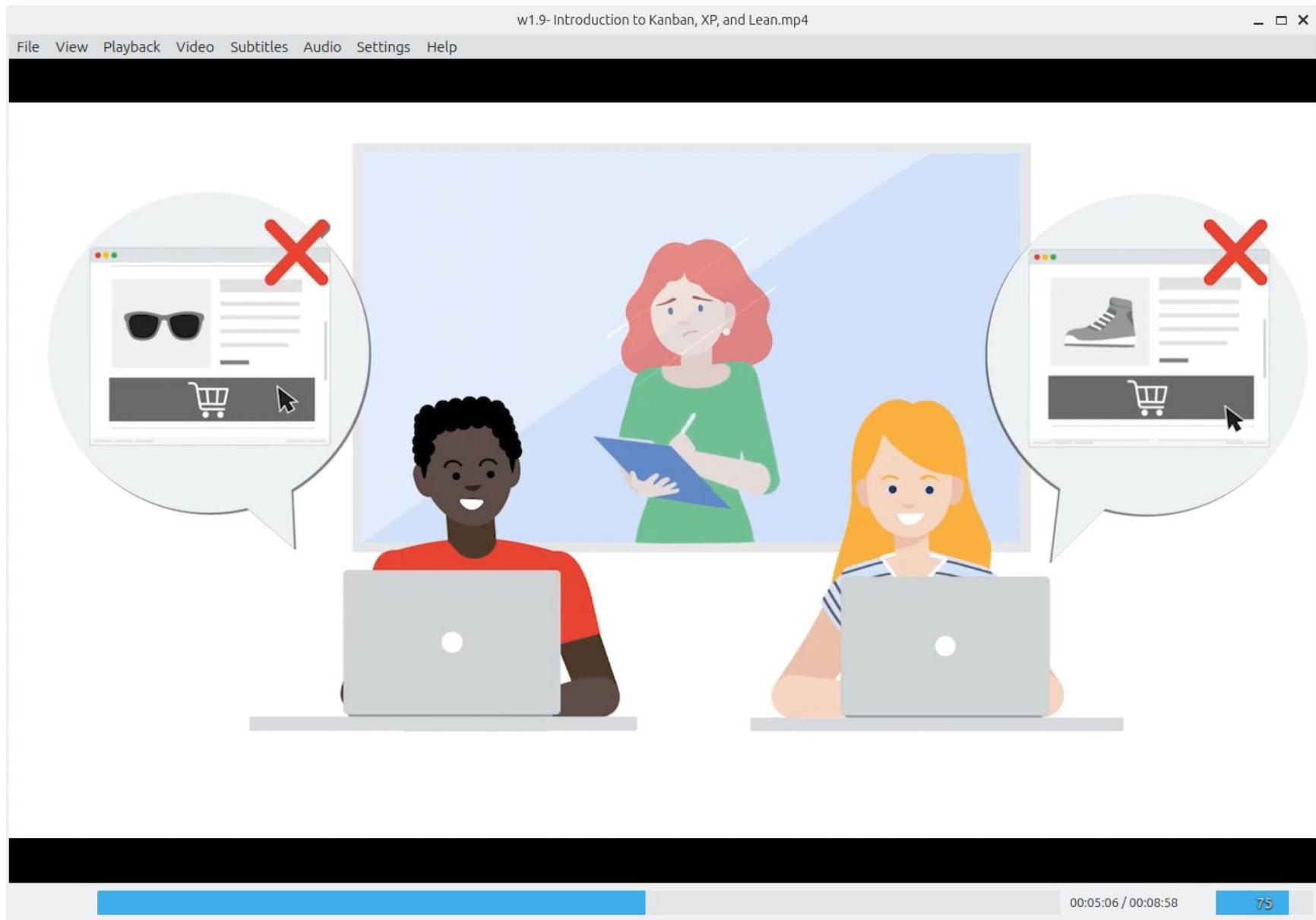
w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

XP Activities

- Designing
- Coding
- Testing

00:04:52 / 00:08:58 75



w1.9- Introduction to Kanban, XP, and Lean.mp4

The goal is to test for and **eliminate any flaws** in a feature before building it and continuing on.

00:05:10 / 00:08:58 75

A screenshot of a video player window titled "w1.9- Introduction to Kanban, XP, and Lean.mp4". The window has a menu bar with "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main area displays a slide with the following text in blue: "The goal is to test for and **eliminate any flaws** in a feature before building it and continuing on." The word "eliminate any flaws" is highlighted with a red rectangular box. At the bottom of the video player, there is a progress bar showing "00:05:10 / 00:08:58" and a page number "75".

The screenshot shows a video player interface with a presentation slide. The title bar reads "w1.9- Introduction to Kanban, XP, and Lean.mp4". The menu bar includes "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". The main content area displays a large blue heading "XP Activities" followed by a bulleted list of activities: "Designing", "Coding", "Testing", and "Listening". A black redaction bar is at the bottom. The bottom right corner shows the video progress bar with "00:05:19 / 00:08:58" and a page number "75".

XP Activities

- Designing
- Coding
- Testing
- Listening

00:05:19 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Listening to the customer and ensuring that the requirements are integrated into the product.

00:05:25 / 00:08:58 75

The image shows a screenshot of a video player window titled "w1.9- Introduction to Kanban, XP, and Lean.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area displays a slide with the text "Listening to the customer and ensuring that the requirements are integrated into the product." This text is highlighted with a red rectangular border. At the bottom of the screen, there is a progress bar indicating the video is at 00:05:25 of a total duration of 00:08:58, and a page number "75" is visible.

Four Basic Activities XP Enhances

1. **Designing:** Stresses **simplicity**. Start with a basic design to meet core requirements, and continuously improve it. The practice of **Avoid Big Design Up Front** ensures the design is just enough to get started.
2. **Coding:** Demands **clear and concise code** (or clear instructions/processes in non-software environments) to make troubleshooting easier.
3. **Testing:** Advocates for **more is better**. Testing should be frequent and thorough, checking smaller features and eliminating flaws *before* continuing work. The idea of **test first development** is taken to the extreme.
4. **Listening:** Crucial for integrating customer requirements, reflecting Agile's value of **customer collaboration, frequent communication, and regular feedback**. The practice **Write Tests, Not Requirements** means the test plan serves both as the requirement definition and the comparison criteria.

You are referring to the Extreme Programming (XP) practice of **Test-Driven Development (TDD)**, which is often summarized by the phrase: "**Write tests, not requirements**" or more accurately, "**Write tests first, then write the code to satisfy the tests.**"

This principle focuses on a tight, fast feedback loop to drive development, which is fundamentally different from a lengthy documentation approach.

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

XP Innovative Practices

- Pair Programming
 - Two team members work together at the same time on one task
- Continuous Integration and Continuous Refactoring
 - Merging product changes into a shared version of the product
- Avoid Big Design Up Front
 - Design is just enough to get started and should be continuously improved as the product evolves
- Write Tests, Not Requirements
 - Embed product requirements into the test plan

00:06:22 / 00:08:58

75

The phrase "**shared version**" in the context of the XP practice **Continuous Integration** refers to the single, central repository where all team members merge their latest work.

Meaning of Shared Version

- **Single Source of Truth:** It is the official, working copy of the project (usually the code base, but can apply to other project artifacts) that every team member uses as their foundation.
- **Constant Merging:** In Continuous Integration, team members frequently merge their product changes (e.g., new features or bug fixes) into this shared version **several times a day.**
- **Purpose:** The goal of merging into this shared version so often is to:
 1. **Get Quick Feedback:** Immediately discover and fix conflicts or quality issues when the changes are small and recent, rather than letting problems build up.
 2. **Maintain Consistency:** Ensure everyone is working off the latest and most stable base, which prevents integration problems later on.

Essentially, the "shared version" is the live, up-to-date state of the product that the entire team is continuously building toward.

The Core Concept

- **Refactoring (The Action):** Making a series of small, behavior-preserving transformations to simplify and improve the existing design of a product's underlying components (like source code, a workflow, or a set of instructions).
- **Continuous (The Frequency):** Integrating this tidying process into the daily, ongoing development work, rather than putting it off for a massive, dedicated "cleanup" phase later.

Continuous improvement= Continuous refactoring (Clean codes)

Continuous refactoring is the practice of regularly improving the internal structure of code without changing its external behavior. It's done in small, incremental steps as part of the normal development cycle, rather than in one large, dedicated effort.

Here's an example of a simple piece of code undergoing continuous refactoring:



Example: Calculating Discounted Price

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

5 Principles of Lean

1. Define value
2. Map value stream
3. Create flow
4. Establish pull
5. Pursue perfection

00:07:04 / 00:08:58 75

w1.9- Introduction to Kanban, XP, and Lean.mp4

File View Playback Video Subtitles Audio Settings Help

Subtitles off

1. **Define Value:** Identify and focus on what the customer wants and include the customer.
2. **Map Value Stream:** Map out the steps to production and challenge all wasted steps.
3. **Create Flow:** Ensure the product flows through the value stream efficiently, eliminating waste throughout the cycle.
4. **Establish Pull:** Ensure the customer is “pulling” on the product through this stream by asking for features and incremental deliveries.
5. **Pursue Perfection:** Push the team to continuously improve the first four process steps.

00:08:14 / 00:08:58 75

3. Lean

Agile was inspired by the manufacturing principles of **Lean**. It is a set of principles and a value system focused on eliminating waste and maximizing customer value.

The Five Principles of Lean

1. **Define Value:** Identify and focus on **what the customer wants**, including the customer in the process.
2. **Map Value Stream:** Map the entire process of producing value and **challenge/eliminate wasteful or unnecessary steps.**
3. **Create Flow:** Ensure the product moves through the value stream efficiently, eliminating interruptions, delays, and barriers.
4. **Establish Pull:** Ensure the **customer is "pulling"** for the product, features, or incremental deliveries throughout the value stream, allowing the team to present what they've been **working on at any time.**
5. **Pursue Perfection:** Continuously push the team to improve the first four process steps.

The phrase "**customer pulling**" in the context of the **Lean methodology** means that work is initiated *only* when the **customer explicitly requests or needs it.**

It describes a **demand-driven system**, which is one of the core principles of Lean.

Understanding "Establish Pull"

The Lean principle "**Establish Pull**" aims to avoid the waste and excess inventory created by a "push" system:

System	Description	Result in Agile
Pull System	Work is pulled into the process by customer demand. The customer's request (e.g., for a feature, an increment, or a product) acts as the signal that triggers the start of the next piece of work.	Your team only delivers what is asked for, reducing the waste of overproduction or building features the customer doesn't want.
Push System	Work is pushed through the process based on a schedule, often before the customer is ready for it or before its value is confirmed.	This can lead to building up unnecessary inventory or features that may need to be thrown away later.

w1.10- Blending project management approaches

w1.10- Blending project management approaches.mp4

File View Playback Video Subtitles Audio Settings Help

- Agile is a way of thinking about the project delivery process through the values and principles of the Manifesto.
- Agile values and principles can be achieved through certain project delivery frameworks and methods.
- The real power of Agile comes from not only adopting certain processes, but also from adopting a certain mindset that is different from traditional Waterfall models.

00:01:05 / 00:05:43 50

Agile as Mindset and Method

The video highlights two ways to view Agile:

1. **Agile as a Mindset:** It is a way of thinking about project delivery, **centered on the values and principles** outlined in the Agile Manifesto. This mindset can be applied even when using non-Agile processes like Waterfall.
2. **Agile as Methods:** It is applied through specific project delivery **frameworks** (like Scrum) and **methods** (like Kanban, XP, and Lean).

w1.10- Blending project management approaches.mp4

File View Playback Video Subtitles Audio Settings Help

Waterfall project management phases

- Initiation
- Planning
- Executing and completing tasks
- Closing out the project

00:01:56 / 00:05:43 50

A screenshot of a video player window titled "w1.10- Blending project management approaches.mp4". The window has a menu bar with "File", "View", "Playback", "Video", "Subtitles", "Audio", "Settings", and "Help". Below the menu is a large blue header with the title "Waterfall project management phases". Underneath the title is a bulleted list of five phases: "Initiation", "Planning", "Executing and completing tasks", "Closing out the project", and another "Executing and completing tasks" which is highlighted with a red border. At the bottom of the video player is a progress bar showing "00:01:56 / 00:05:43" and a volume slider set to "50".

w1.10- Blending project management approaches.mp4

File View Playback Video Subtitles Audio Settings Help

Reasons to blend Waterfall and Agile styles

- Your stakeholders, customers, or sponsors are more comfortable with traditional approaches and workflows, but your project team is already established in Scrum.
- Regulatory requirements insist on certain traditional work processes.
- A vendor is already following **a traditional approach** and the integration between the teams requires some blending of methods.

00:02:52 / 00:05:43 50

Reasons for Blending

Blending methods can be necessary to accommodate various project constraints:

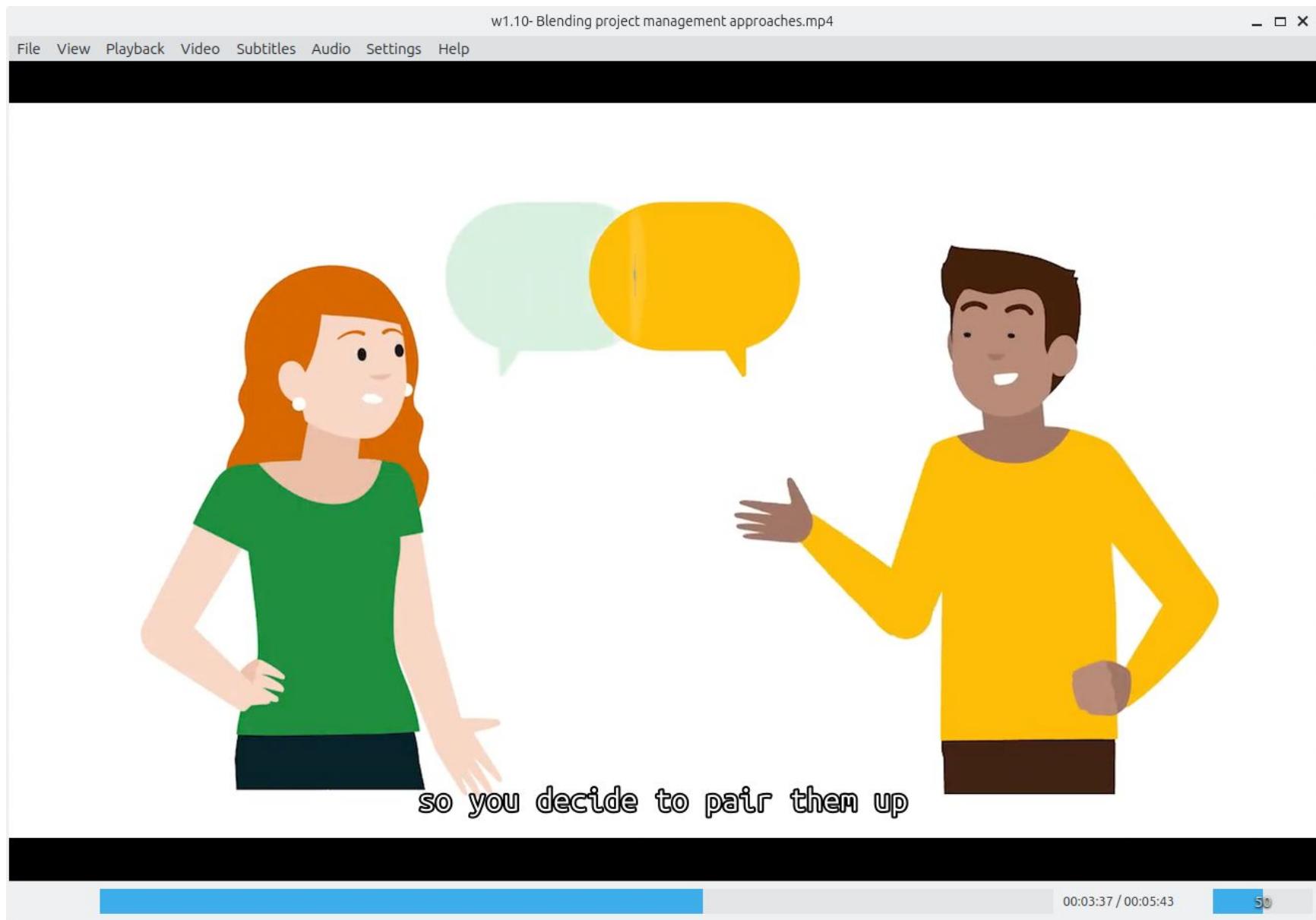
- **Stakeholder Comfort:** Stakeholders, customers, or sponsors may prefer and better understand **traditional workflows** or work products.
- **Regulatory Requirements:** Certain industries require **traditional work processes**, such as producing large requirement documents for certification.
- **Vendor Integration:** A major project vendor may already be committed to a **traditional approach**, requiring the project to bridge the methods for integration.
- **Team Preference:** A project team may be established in a method like **Scrum** and wish to continue using it, even if some external project factors require a traditional approach.

Examples of Blending Methods

Blending often involves integrating a *practice* from one methodology into the framework of another:

- **XP and Scrum:** A team using **Scrum** (with its **retrospectives** and **Sprints**) adopts **Pair Programming** from **XP** to help a team member learn a new feature from an expert.
- **Kanban and Scrum:** A **Scrum** team uses a **Kanban board** to visually track and manage the progress of tasks within their Sprint. 
- **Agile and Waterfall:** In the *Office Green* case study, while the delivery team may use an Agile method like Scrum, the project manager might use **traditional budget management controls** to satisfy the need for strict cost oversight. Similarly, they may need to involve vendors early on to gain buy-in, as some may not be ready for an Agile approach.

The key takeaway is to **mix things up** and blend styles when different parts of the project can benefit, as long as it doesn't negatively impact the project as a whole. Consistency in how a team works is also important to maintain effectiveness.



The image shows a screenshot of a video player window titled "w1.10- Blending project management approaches.mp4". The menu bar includes File, View, Playback, Video, Subtitles, Audio, Settings, and Help. The main content area displays a slide with a blue header "Key points" and a bulleted list of three items. The third item, "Agile and Waterfall are both effective ways of approaching project management.", is highlighted with a red rectangular box. The video player interface at the bottom shows a progress bar, the time "00:05:11 / 00:05:43", and a page number "50".

Key points

- Agile is a mindset.
- Agile values and principles can be achieved through certain frameworks and delivery methods.
- Agile and Waterfall are both effective ways of approaching project management.

Yes, the **Waterfall methodology** is considered the **traditional approach** when compared to **Scrum** (which is an Agile framework).

The distinction lies primarily in their approach to planning, flexibility, and delivery:

Feature	Waterfall (Traditional)	Scrum (Agile Framework)
Approach	Linear and Sequential (Plan → Build → Test → Deploy).	Iterative and Incremental (Build → Test → Deploy in short cycles).
Flexibility	Low. Requirements are fixed upfront; change is expensive and discouraged.	High. Requirements can be reviewed and adjusted at the start of every Sprint.
Customer Involvement	Low. Typically involved only at the beginning (requirements) and end (final review).	High. Customer/Product Owner is highly engaged, often reviewing work at the end of every Sprint Review.
Risk	High. Problems are often discovered late in the testing phase.	Lower. Risks are identified and addressed early due to continuous testing and frequent feedback.
Goal	Deliver the entire product in one big launch.	Deliver small, working pieces of the product frequently and continuously.

w1.11- Wrap-up

This introductory module was a whirlwind tour covering the foundational elements of the **Agile** approach to project management.

Topic	Deutsch (German)
Agile Fundamentals	You explored the history of Agile , the Agile Manifesto , and its four core values and 12 principles .
Mindset & Application	You learned <i>when</i> and <i>why</i> to adopt an Agile mindset and how to use the VUCA concept (Volatility, Uncertainty, Complexity, Ambiguity) to determine the best approach for a project.
Agile Methodologies	You were introduced to several Agile-friendly methodologies, including Kanban , Extreme Programming (XP) , and Lean .
Blending Approaches	You learned how to blend these modern Agile methods with traditional Waterfall practices.
Practical Application	The module concluded with your return to Office Green as the project manager for the new Agile project, Virtual Verde .