

Good morning everyone,

Thank you for being here today for my presentation.

My name is Long Nguyen, and I currently work as an Agile consultant. I'm thrilled to have the opportunity to share my knowledge about Agile and discuss how adopting an Agile mindset can benefit your project in the near future.

My presentation will focus on two main parts:

- ◆ Part 1: Onboarding the Team
- ◆ Part 2: Agile Practice
- ◆ Part 3: High Performing Agile Teams

I believe that my insights into the Agile mindset will open the door to a new way of efficiently managing your projects in the future.

If you have any questions, don't hesitate to let me know.

Part 1: Agile Benefits for the team

Challenge



Feedback from customers is obtained at a delayed stage.

Solution



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



The first key challenge that WorldVisitz faced is feedback from customers is obtained at a delayed stage, which can result in misaligned expectations and costly late-stage changes.

- To address this, Agile emphasizes the principle: *"Our highest priority is to satisfy the customer through early and continuous delivery of valuable software."* By delivering smaller, working increments of the product frequently, we ensure that customer feedback is gathered early and consistently. This enables us to align with their needs, make timely adjustments, and ultimately deliver a product that meets their expectations.

Part 1: Agile Benefits for the team

Challenge



The team lacks close collaboration, both in terms of physical presence and shared spirit.

Solution



In Agile, the most effective architectures, requirements, and designs are developed by self-organizing teams.



- Currently, work is planned and assigned by a single person, often based on individual preference, which creates inefficiencies, limits collaboration, and prevents the team from reaching its full potential.
- So the solution is adopt Agile mindset. In Agile, the most effective architectures, requirements, and designs emerge from self-organizing teams. By empowering team members to take ownership of their work, collaborate closely, and plan collectively, Agile creates an environment that fosters shared accountability and improves team efficiency.
- **Jane Doe - Project Manager** plans the work herself and assigns tasks based on her preference, limiting collaboration and team empowerment.
 - **Agile Solution:** Shift Jane's role from task-assigner to a facilitator and servant leader. Her focus should move towards enabling the team to self-organize, collaboratively plan sprints, and identify work priorities during team ceremonies like Sprint Planning.
 - **Benefit:** This transition empowers the team to take ownership of their work while enabling Jane to focus on removing obstacles and driving value delivery.
- **John Smith - Product Manager** works in silos, focusing heavily on upfront market

research and product planning, staying detached from the development team.

- **Agile Solution:** Transition John's role to **Product Owner**, where he actively collaborates with the development team throughout the sprint cycle. Instead of upfront planning, John will break down product requirements into a prioritized **Product Backlog** and continuously refine it based on feedback from both customers and the team.
- **Benefit:** As a Product Owner, John will work iteratively, focusing on delivering the **highest-value features** first while incorporating real-time feedback. This ensures that the product adapts to market needs quickly and reduces the risk of wasted efforts from over planning.
- **Jim Brady & Nathan Connor (Onshore Developers)** receive simpler tasks and lack opportunities to learn from offshore contractors. There's also inconsistency in coding standards.
 - **Agile Solution:** Implement pair programming and knowledge-sharing sessions to allow onshore developers to collaborate with offshore contractors. Adopt a collective code ownership practice to standardize coding processes.
 - **Benefit:** Onshore developers upskill, team-wide coding standards improve, and knowledge gaps are bridged.
- **Venkat Ragu & Ali Khan (Offshore Developers)** handle the complex deliverables individually but operate in isolation, missing opportunities to collaborate and transfer knowledge.
 - **Agile Solution:** Include offshore developers in daily stand-ups, backlog refinement, and sprint retrospectives to promote active collaboration. Encourage offshore developers to mentor onshore developers during pair programming and code reviews.
 - **Benefit:** Offshore developers share their technical expertise, improving team cohesion and overall knowledge transfer.

Part 1: Agile Benefits for the team

Challenge



Requirements and product planning are handled entirely at the beginning of the project.

Solution



In Scrum change is the norm, and requests are re-prioritized at Sprint boundaries

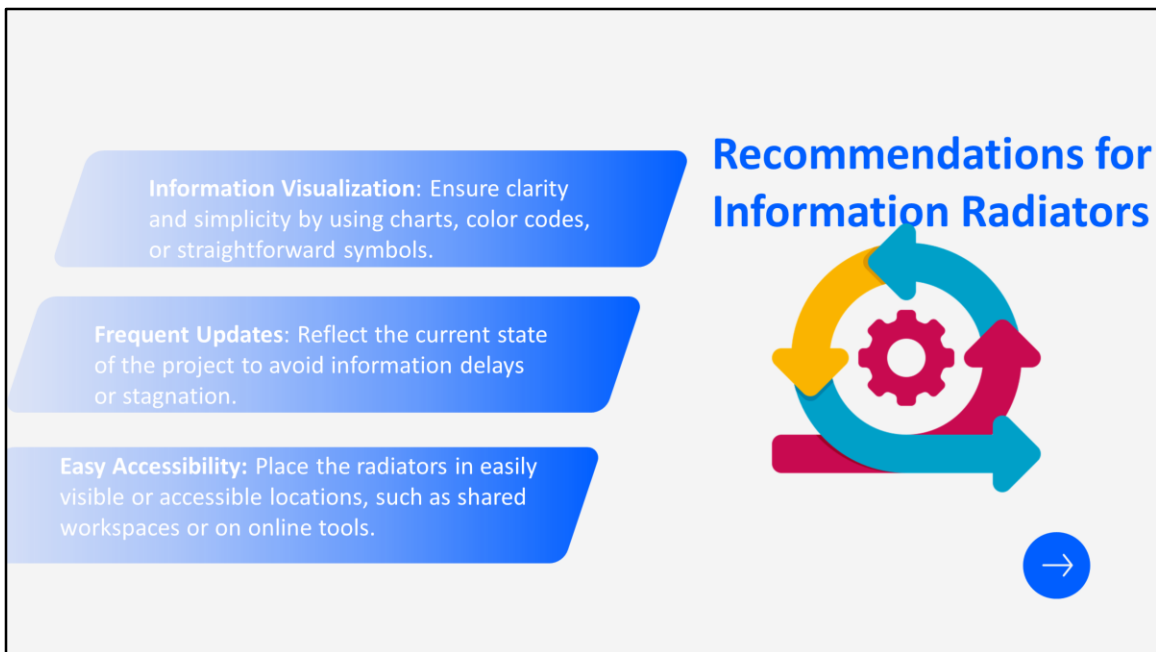


- Under the previous model, all requirements were gathered upfront and remained fixed throughout the development process, which often resulted in inflexibility when market conditions or customer needs evolved.
- By shifting to Agile and enabling John to act as the Product Owner, he can continuously prioritize and refine the **Product Backlog** based on real-time feedback from stakeholders and end-users. This iterative approach ensures that the team delivers high-value features early and frequently, addressing the most critical customer needs while adapting to emerging changes.
- Consequently, the project becomes more dynamic, customer-focused, and resilient to shifting priorities, ultimately leading to better outcomes for both the business and its users.

Part 2: Agile Practice



Let's shift our attention to **Part 2: Agile Practice**, where we will delve into two fundamental components of Agile: **Agile Radiators** and **Scrum Ceremonies**. These elements are crucial to fostering transparency, collaboration, and continuous improvement in Agile environments.



To ensure transparency, effective communication, and a shared understanding within the team and stakeholders, implementing **Information Radiators** is crucial in Agile practices.

When implementing **Information Radiators** in Agile, it's crucial to focus on their core purpose: fostering transparency, alignment, and collaboration within the team.

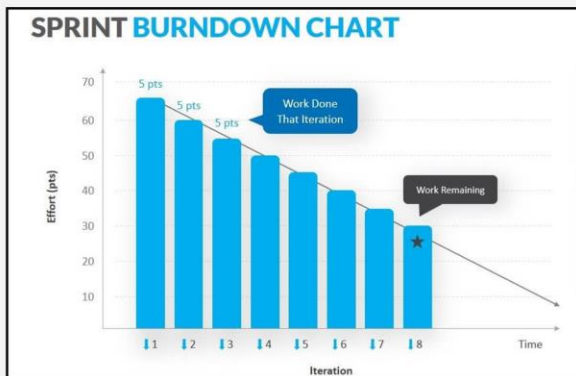
By making key project metrics and progress visible to everyone, these tools act as a shared source of truth, enabling teams to make informed decisions quickly.

In this section, we'll focus on two essential Information Radiators that can significantly enhance the flow of information:

- **Burndown Chart:** A tool to monitor Sprint progress and identify pacing issues.
- **Velocity Chart:** A tool to measure and predict the team's delivery capacity over time.

By leveraging these visual tools, Agile teams can make informed decisions, stay aligned on priorities, and continuously improve their processes. Let's delve deeper into how each chart works and how it can benefit your project.

Burndown chart



A **Burn Down Chart** is a key tool used in **Agile project management** to visually track the progress of a project and show how much work remains to be done over time. It is particularly useful in Scrum, one of the Agile frameworks, to monitor the completion of tasks or user stories during a sprint.

Key Elements of a Burn Down Chart:

- **X-Axis (Time):** The horizontal axis represents the time duration of the sprint, typically in days. It shows the time elapsed since the beginning of the sprint.
- **Y-Axis (Work Remaining):** The vertical axis shows the amount of work remaining, usually measured in story points or hours. This represents the effort needed to complete the user stories or tasks.
- **Ideal Burn Down Line:** This is a diagonal line that starts from the top-left corner (representing the total amount of work) and ends at the bottom-right corner (representing zero work remaining). It shows the ideal progress needed to complete all tasks by the end of the sprint.
- **Actual Burn Down Line:** This line represents the actual progress made throughout the sprint. It starts at the same point as the ideal burn down line but fluctuates based on how much work is completed each day.

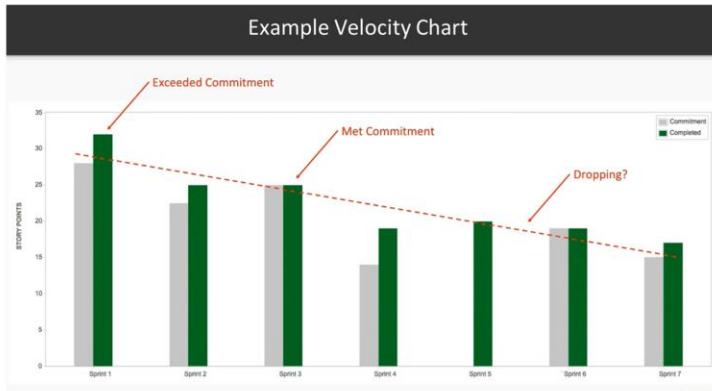
The chart provides a clear visual of how much work remains versus the time left in

the sprint or project. For WorldVisitz, this transparency ensures that everyone—developers, Jane as the project manager, and John as the product owner—can stay aligned on the project’s status and identify risks early.

By comparing the actual progress line to the ideal burndown line, the team can quickly detect if they are falling behind. This helps WorldVisitz proactively adjust priorities or allocate resources to avoid missing deadlines, particularly in a dynamic market.

The burndown chart keeps the team focused on delivering user stories that provide value. For WorldVisitz, where the transition to Agile emphasizes customer-centric outcomes, this alignment ensures that the team prioritizes high-impact tasks in line with the product backlog.

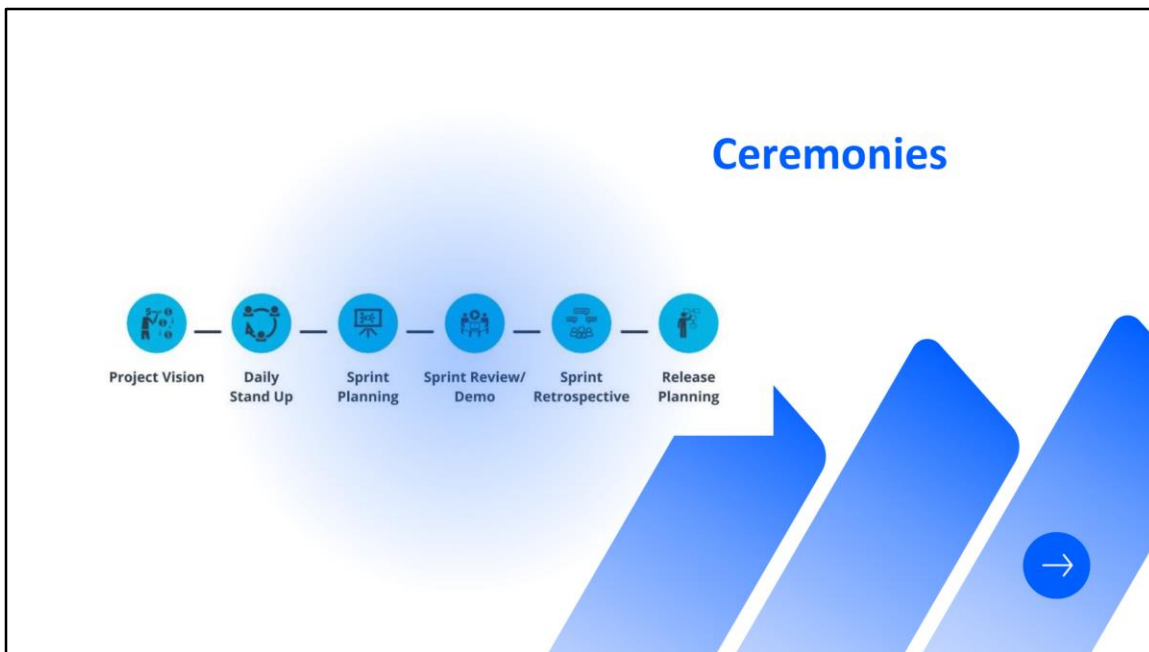
Velocity chart



The **Velocity Chart** is another essential Agile tool that can greatly benefit **WorldVisitz**, particularly as they transition from a traditional waterfall model to Agile practices.

Here's how the velocity chart proves useful:

The velocity chart displays the amount of work (e.g., story points) completed in each sprint. For **WorldVisitz**, this helps the team estimate their capacity for future sprints, making sprint planning more reliable and realistic. Over time, the team can develop a predictable velocity, ensuring better forecasting for deliverables and deadlines.



To guide our Agile implementation for WorldVisitz, I propose using the Scrum framework. To assist you in this transition, I'd like to present a sample Scrum Ceremony Schedule.

- Scrum is an agile project management framework that focuses on iterative development and continuous improvement.
- One of the key elements of Scrum is its ceremonies, which help teams stay organized, collaborate effectively, and maintain focus on their goals.
- These 6 ceremonies provide structure to the Scrum process and are essential for the successful execution of a Scrum project.
- Let's take a closer look at each of these ceremonies.

Project Vision: This ceremony is where business leaders define the project's objectives, success criteria, assumptions, and risks. It occurs once before the project launch and is time-boxed to 1 hour. The outcome is a statement of the desired future state.

Daily Stand Up: A 15-minute daily meeting where each team member answers three questions: What did I do? What will I do today? What obstacles am I facing? This promotes synchronization and transparency within the team.

Sprint Planning: The team collaborates to plan the work for the upcoming sprint. It is

time-boxed to 8 hours for a 1-month sprint, and the team works with the Product Owner to determine what to deliver, how to achieve it, and estimates for tasks.

Sprint Review/Demo: The team demonstrates the completed work to the Product Owner (PO) and business representatives to gather feedback. The PO determines if the deliverables meet the Acceptance Criteria. This ceremony occurs once at the end of each sprint and is time-boxed to 4 hours for a one-month sprint.

Sprint Retrospective: The team reflects on its performance during the sprint, discussing successes and areas for improvement. This meeting is held after the Sprint Review, time-boxed to 3 hours for a one-month sprint.

Release Planning: The team creates a plan for when a collection of features will be released to the customer. The agenda includes discussing critical dates, coordinating with other teams, and balancing business value and quality. This occurs only when a release is planned and is time-boxed to 20 minutes.

Ceremonies

Ceremony	Purpose	Key Activities	Outcome
Project Vision	Align the team on the project's goals, scope, and purpose.	<ul style="list-style-type: none">- Define the product vision and objectives.- Identify key stakeholders and their needs.	A shared understanding of the project goals.
Daily Stand Up	Synchronize the team's progress and identify obstacles.	<ul style="list-style-type: none">- Each member answers:<ol style="list-style-type: none">1. What did I do yesterday?2. What will I do today?3. What obstacles do I face?	Awareness of team progress and blockers.
Sprint Planning	Plan the work for the upcoming Sprint.	<ul style="list-style-type: none">- Discuss and prioritize the Product Backlog.- Define Sprint Goal.- Break down stories into tasks.	A clear Sprint Goal and task list.

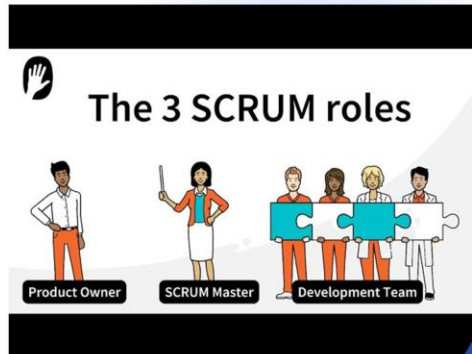


Ceremonies

Ceremony	Purpose	Key Activities	Outcome
Sprint Review/Demo	Showcase the work completed in the Sprint.	<ul style="list-style-type: none">- Demonstrate completed user stories.- Collect stakeholder feedback.- Discuss potential adjustments.	Stakeholder feedback and next steps.
Sprint Retrospective	Reflect on the Sprint to identify improvements.	<ul style="list-style-type: none">- Discuss what went well, what didn't, and what can be improved.- Develop action items for improvement.	Actionable steps for continuous improvement.
Release Planning	Reflect on the Sprint to identify improvements.	<ul style="list-style-type: none">- Define the timeline and scope for releases.- Prioritize features for the release.- Address risks..	A realistic release plan aligned with goals.



Core Roles



In Scrum, there are three key roles, each with specific responsibilities to ensure the team works effectively:

- **Product Owner:** The Product Owner is responsible for defining the project's goals, managing the product backlog, and ensuring that the team works on the most valuable tasks. They prioritize work based on business needs and customer feedback and act as the main liaison between the business and the Scrum team.
- **Scrum Master:** The Scrum Master serves as a facilitator and coach for the Scrum team. They ensure that Scrum practices are followed, help remove any obstacles the team may face, and support the team in improving its processes. They also assist the Product Owner in managing the backlog and ensure that the team remains focused on delivering value.
- **Development Team:** The Development Team is composed of professionals who work together to deliver the product increment. They are self-organizing, cross-functional, and responsible for completing the tasks defined in the sprint backlog. The team collaborates closely to ensure the quality and functionality of the product.

Part 3: High Performing Agile Teams



Let's shift our attention to **Part 3: High Performing Agile Team**, where we will delve into how adopt Agile in our project.

Necessary roles based on the Scrum framework

Team Member	Agile Role	Rationale
John Smith	Product Owner (PO)	Manages the Product Backlog, prioritizes work to align with business goals, and ensures focus on customer needs.
Jane Doe	Scrum Master	Facilitates Scrum ceremonies, removes impediments, and fosters collaboration between onshore and offshore teams.
Jim, Nathan, Venkat, Ali	Developers	Work on deliverables collaboratively, adhering to coding standards and ensuring knowledge sharing.
Kathy Qualls	QA Specialist/ Tester	Ensures quality by being involved in sprint activities rather than testing only post-development.
Jerry Holden	Business Analyst (Team Member)	Supports PO in refining backlog items and ensuring business value in each increment.
James Cowx	UX Designer (Team Member)	Works closely with developers to iterate UX designs and integrate customer feedback during development.
Holly Vogt	Subject Matter Expert (Advisor)	Provides guidance to PO and Scrum Team as needed to refine and validate key features.

- When transitioning to an Agile approach, clearly defining roles within the team is a critical step to ensure collaboration, accountability, and focus. Unlike traditional models where roles are rigidly siloed, Agile empowers individuals by giving them ownership and responsibility for specific aspects of the process.
- In the case of WorldVisitz, assigning Agile roles will help streamline communication, enhance decision-making, and foster a shared commitment to delivering value. By aligning each team member's strengths with their new Agile responsibilities, we create a balanced team dynamic that enables continuous progress and innovation.
- Let's take a closer look at how each member's role evolves to support the Agile transformation.

Appendix 1 - Skill Gaps

Team Member	Skill Gaps
John Smith	John Smith's detached approach to product planning creates a lack of alignment between the Product Manager and development team.
Jane Doe	Jane Doe assigns tasks based on her preferences, which hinders team self-organization and empowerment.
Development Team	Onshore and offshore developers lack trust and camaraderie. Offshore developers handle complex tasks without transferring knowledge to onshore counterparts.
Kathy Qualls	Kathy Qualls provides feedback only after code completion, delaying issue resolution.
Jerry Holden	Jerry Holden freezes requirements upfront, making it difficult to adapt to changing needs.
James Cowx	James builds the UX based solely on predefined specifications and requirements, without incorporating iterative customer feedback during the design phase.
Holly Vogt	Holly Vogt's insights are shared through one-on-one conversations, leaving the team without shared documentation or accessible guidance.

The case study of WorldVisitz highlights several skill gaps that could hinder the team's Agile transformation and delivery success. Addressing these gaps will be critical to ensure a smooth transition to Agile and effective collaboration. Here's an analysis of the skill gaps for each role:

Appendix 2 - Training & Coaching Plan

Team Member	Topic	Key areas to improvement
John Smith (Product Manager)	Becoming a Product Owner	<ul style="list-style-type: none">- Managing and prioritizing the Product Backlog.- Writing effective user stories.- Collaborating with stakeholders and developers.
Jane Doe (Project Manager)	Transition to Scrum Master Role	<ul style="list-style-type: none">- Agile mindset and servant leadership.- Facilitation of Scrum ceremonies.- Coaching on self-organization and team empowerment.
Development Team	Coding Standards and Agile Mindset	<ul style="list-style-type: none">- Establishing consistent coding standards.- Knowledge-sharing (onshore-offshore collaboration).- Pair programming and code reviews.
Kathy Qualls (Tester)	Agile Testing Practices	<ul style="list-style-type: none">- Test-driven development (TDD) and behavior-driven development (BDD).- Shift-left testing.- Writing automated test scripts.

The training and coaching plan focuses on addressing skill gaps, fostering Agile practices, and promoting collaboration within the team. Each session targets specific roles, improving both technical and soft skills essential for Agile transformation.

Appendix 2 - Training & Coaching Plan

Team Member	Topic	Key areas to improve
Jerry Holden (Business Analyst)	Agile Requirements Gathering	<ul style="list-style-type: none">- Refining and evolving requirements with the Product Owner.- Writing and updating user stories.- Engaging stakeholders for continuous feedback.
James Cowx (UX Designer)	Agile UX Design	<ul style="list-style-type: none">- Iterative design and rapid prototyping.- Collaborating with the team during Sprints.- Incorporating feedback from Sprint Reviews.
Holly Vogt (Subject Matter Expert)	Collaborative Agile Practices	<ul style="list-style-type: none">- Refining priorities with the Product Owner.- Providing feedback during Sprint Reviews.- Sharing expertise during backlog refinement sessions.

And all team should focus on Agile Ceremonies and Collaboration Coaching

How the current skill gaps can be fixed with a coaching plan

Role	Skill Gaps	Stage	Coaching Plan
Jane Doe (Project Manager)	Focuses on assigning tasks individually, lacks Agile mindset.	Shu	Provide foundational Agile training on Scrum roles, ceremonies, and values. Introduce her to the concept of servant leadership and team empowerment.
	Finds it hard to relinquish control over detailed planning.	Ha	Mentor her on backlog refinement and collaborative sprint planning. Encourage delegation of task assignments to the team.
John Smith (Product Manager)	Detached from developers due to traditional upfront planning practices.	Shu	Educate on Agile product ownership, emphasizing continuous backlog prioritization and regular engagement with the team.
	Unfamiliar with breaking requirements into smaller, deliverable chunks.	Ha	Provide hands-on coaching for backlog grooming and user story writing. Focus on creating valuable, incremental deliverables with the team.

- Now that we've identified the key skill gaps and challenges within the WorldVisitz team, it's clear that bridging these gaps requires a structured and systematic approach. To address this, I propose leveraging the **Shu-Ha-Ri model**—a proven framework for skill development and mindset transformation.
- The Shu-Ha-Ri model guides individuals and teams through three stages of learning: **Shu (Follow)**, where foundational knowledge is established; **Ha (Adapt)**, where practices are refined and tailored; and finally, **Ri (Innovate)**, where the team achieves mastery and autonomy.
- By applying this model, we can ensure that each team member progresses effectively, building the skills, collaboration, and trust required for a successful Agile transformation at WorldVisitz.



How the current skill gaps can be fixed with a coaching plan

Role	Skill Gaps	Stage	Coaching Plan
Development Team	Lack of knowledge-sharing processes and inconsistent coding standards.	Shu	Conduct workshops on Agile engineering practices (e.g., pair programming, code reviews, and consistent coding standards).
	Offshore and onshore developers feel isolated and disconnected.	Ha	Facilitate team-building exercises and cross-location collaboration through virtual tools and ceremonies. Foster a shared ownership culture within the team.
Kathy Qualls (Tester)	Joins testing late in the process, limiting feedback and integration.	Shu	Train on Agile testing principles, including test-driven development (TDD) and integrating QA earlier in the sprint (shift-left approach).
	Feedback loop is delayed, reducing efficiency.	Ha	Encourage active participation in daily standups, backlog refinement, and sprint planning to provide earlier feedback.

How the current skill gaps can be fixed with a coaching plan

Role	Skill Gaps	Stage	Coaching Plan
Jerry Holden (Business Analyst)	Focuses on freezing detailed requirements upfront, limiting adaptability.	Shu	Train on Agile requirements management and incremental delivery. Introduce user stories and story mapping as tools for dynamic prioritization.
	Lacks collaboration with developers and UX designer.	Ha	Embed Jerry into the Scrum team to work collaboratively with developers and UX in backlog grooming and sprint planning.
James Cowx (UX Designer)	Receives customer feedback late, causing inefficiencies in design improvements.	Shu	Provide training on Lean UX and incorporating feedback in iterative cycles.
	Works based on static project requirements rather than dynamic collaboration.	Ha	Foster close collaboration with the Product Owner and developers. Embed UX in Agile ceremonies (e.g., sprint reviews and retrospectives).

How the current skill gaps can be fixed with a coaching plan

Role	Skill Gaps	Stage	Coaching Plan
Holly Vogt (Subject Matter Expert)	Provides input in a siloed manner, relying on one-on-one interviews rather than collaborative forums.	Shu	Educate on Agile team dynamics and shared knowledge. Introduce SME participation in team ceremonies, such as sprint reviews or backlog grooming.
	Limited direct interaction with the full development team.	Ha	Encourage regular knowledge-sharing sessions with the team and active involvement in cross-functional discussions.

