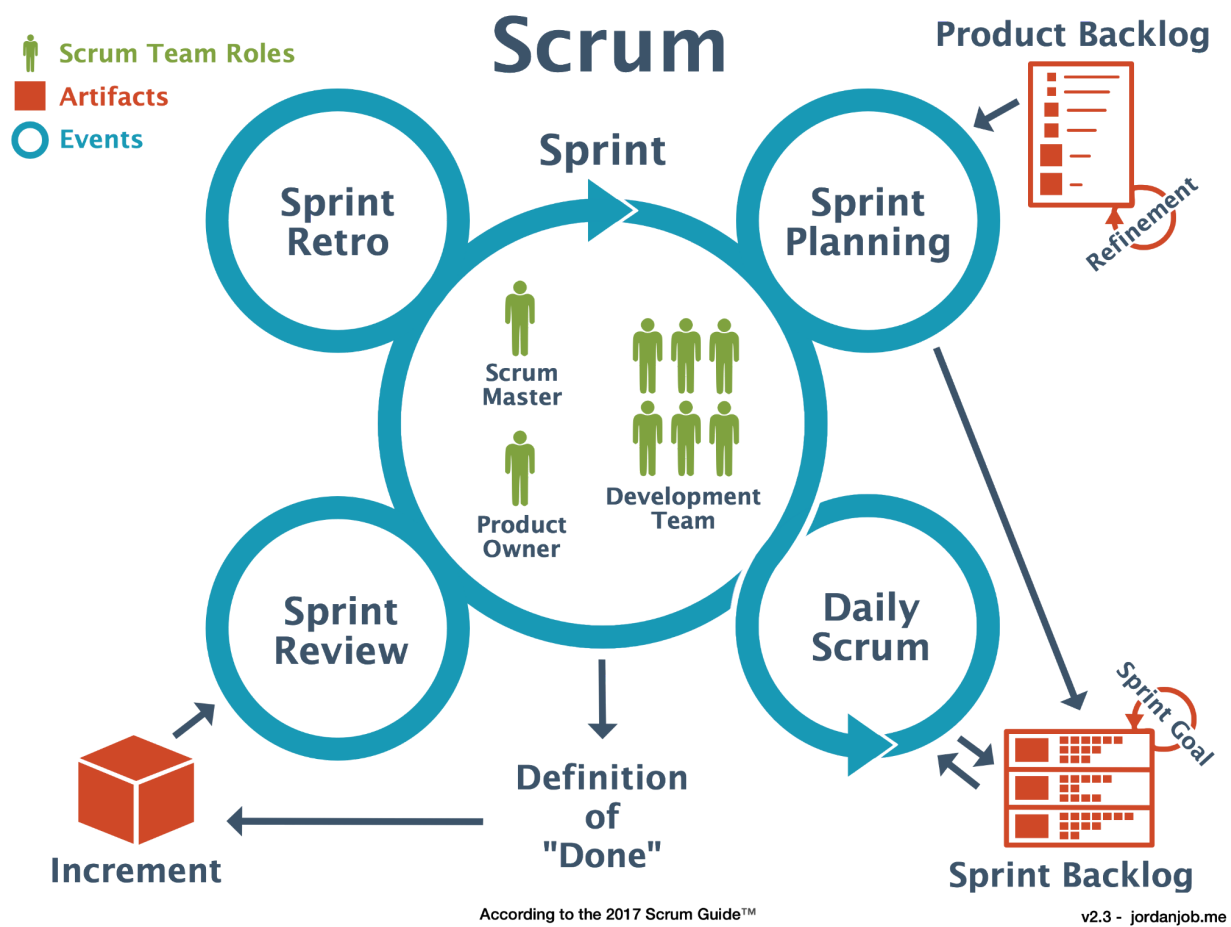


AGILE PROFESSIONAL  
ENG. KOUGANG ROSTAND

# SCRUM

## METHODOLOGY



**PRESENTATION BY GROUP 2**  
ABEA BRANDON ABEA  
ICTU20212415

ATAWAN BELTUS ASANJI  
ICTU20212260

KERNYUY FOMO NSAHLAI  
ICTU20212274

TIM RODNEY-DAVIS NCHAMCHAM  
ICTU20212489

NDEFO PAUL AUGUSTE  
ICTU20212527

BATIMEHIN YOUNG AYODEJI  
ICTU20222873

# OUTLINE

<b>OUTLINE</b>	<b>2</b>
<b>Scrum Introduction</b>	<b>3</b>
<b>Mindset Essentials</b>	<b>10</b>
<b>Application And Domain</b>	<b>11</b>
<b>Best Practice</b>	<b>11</b>
<b>Implementation</b>	<b>12</b>
<b>Case Study</b>	<b>12</b>
<b>Question And Answer</b>	<b>13</b>

## Scrum Introduction

### Scrum Definition

Scrum is a lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems. To better understand, many people ask **So, why is it called Scrum?** " People often ask, "Is

Scrum an acronym for something?” and the answer is no. It is actually inspired by a scrum in the sport of rugby. In rugby, the team comes together in what they call a scrum to work together to move the ball forward. In this context, Scrum is where the team comes together to move the product forward in order to meet their expected goals.

In a nutshell, Scrum requires a Scrum Master to foster an environment where:

1. A Product Owner orders the work for a complex problem into a Product Backlog.
2. The Scrum Team turns a selection of the work into an Increment of value during a Sprint.
3. The Scrum Team and its stakeholders inspect the results and adjust for the next Sprint.
4. Repeat

Scrum is simple. Try it as it is and determine if its philosophy, theory, and structure help to achieve goals and create value. The Scrum framework is purposefully incomplete, only defining the parts required to implement Scrum theory. Scrum is built upon by the collective intelligence of the people using it. Rather than provide people with detailed instructions, the rules of Scrum guide their relationships and interactions.

Various processes, techniques and methods can be employed within the framework. Scrum wraps around existing practices or renders them unnecessary. Scrum makes visible the relative efficacy of current management, environment, and work techniques, so that improvements can be made.

## **Scrum Values**

Successful use of Scrum depends on people becoming more proficient in living five values:

### **Commitment, Focus, Openness, Respect, and Courage**

The Scrum Team commits to achieving its goals and to supporting each other. Their primary focus is on the work of the Sprint to make the best possible progress toward these goals. The Scrum Team and its stakeholders are open about the work and the challenges. Scrum Team members respect each other to be capable, independent people, and are respected as such by the people with whom they work. The Scrum Team members have the courage to do the right thing, to work on tough problems.

These values give direction to the Scrum Team with regard to their work, actions, and behavior. The decisions that are made, the steps taken, and the way Scrum is used should reinforce these values, not diminish or undermine them. The Scrum Team members learn and explore the values as they work with the Scrum events and artifacts. When these values are embodied by the Scrum Team and the people they work with, the empirical Scrum pillars of transparency, inspection, and adaptation come to life building trust.

## **Scrum Team**

The fundamental unit of Scrum is a small team of people, a Scrum Team. The Scrum Team consists of one Scrum Master, one Product Owner, and Developers. Within a Scrum Team, there are no sub-teams or hierarchies. It is a cohesive unit of professionals focused on one objective at a time, the Product Goal.

Scrum Teams are cross-functional, meaning the members have all the skills necessary to create value each Sprint. They are also self-managing, meaning they internally decide who does what, when, and how.

The Scrum Team is small enough to remain nimble and large enough to complete significant work within a Sprint, typically 10 or fewer people. In general, we have found that smaller teams communicate better and are more productive. If Scrum

Teams become too large, they should consider reorganizing into multiple cohesive Scrum Teams, each focused on the same product. Therefore, they should share the same Product Goal, Product Backlog, and Product Owner.

The Scrum Team is responsible for all product-related activities from stakeholder collaboration, verification, maintenance, operation, experimentation, research and development, and anything else that might be required. They are structured and empowered by the organization to manage their own work. Working in Sprints at a sustainable pace improves the Scrum Team's focus and consistency.

The entire Scrum Team is accountable for creating a valuable, useful Increment every Sprint. Scrum defines three specific accountabilities within the Scrum Team: the Developers, the Product Owner, and the Scrum Master.

## **Developers**

Developers are the people in the Scrum Team that are committed to creating any aspect of a usable Increment each Sprint.

The specific skills needed by the Developers are often broad and will vary with the domain of work. However, the Developers are always accountable for:

- Creating a plan for the Sprint, the Sprint Backlog;
- Instilling quality by adhering to a Definition of Done;
- Adapting their plan each day toward the Sprint Goal; and,
- Holding each other accountable as professionals.

## **Product Owner**

The Product Owner is accountable for maximizing the value of the product resulting from the work of the Scrum Team. How this is done may vary widely across organizations, Scrum Teams, and individuals.

The Product Owner is also accountable for effective Product Backlog management, which includes:

- Developing and explicitly communicating the Product Goal;
- Creating and clearly communicating Product Backlog items;

- Ordering Product Backlog items; and,
- Ensuring that the Product Backlog is transparent, visible and understood.

The Product Owner may do the above work or may delegate the responsibility to others. Regardless, the Product Owner remains accountable.

For Product Owners to succeed, the entire organization must respect their decisions. These decisions are visible in the content and ordering of the Product Backlog, and through the inspectable Increment at the Sprint Review.

The Product Owner is one person, not a committee. The Product Owner may represent the needs of many stakeholders in the Product Backlog. Those wanting to change the Product Backlog can do so by trying to convince the Product Owner.

## **Scrum Master**

The Scrum Master is accountable for establishing Scrum as defined in the Scrum Guide. They do this by helping everyone understand Scrum theory and practice, both within the Scrum Team and the organization.

The Scrum Master is accountable for the Scrum Team's effectiveness. They do this by enabling the Scrum Team to improve its practices, within the Scrum framework.

Scrum Masters are true leaders who serve the Scrum Team and the larger organization.

The Scrum Master serves the Scrum Team in several ways, including:

- Coaching the team members in self-management and cross-functionality;
- Helping the Scrum Team focus on creating high-value Increments that meet the Definition of Done;
- Causing the removal of impediments to the Scrum Team's progress; and,
- Ensuring that all Scrum events take place and are positive, productive, and kept within the timebox.

The Scrum Master serves the Product Owner in several ways, including:

- Helping find techniques for effective Product Goal definition and Product Backlog management;
- Helping the Scrum Team understand the need for clear and concise Product Backlog items;
- Helping establish empirical product planning for a complex environment; and,
- Facilitating stakeholder collaboration as requested or needed.

The Scrum Master serves the organization in several ways, including:

- Leading, training, and coaching the organization in its Scrum adoption;
- Planning and advising Scrum implementations within the organization;
- Helping employees and stakeholders understand and enact an empirical approach for complex work; and,
- Removing barriers between stakeholders and Scrum Teams.



# MINDSET ESSENTIALS

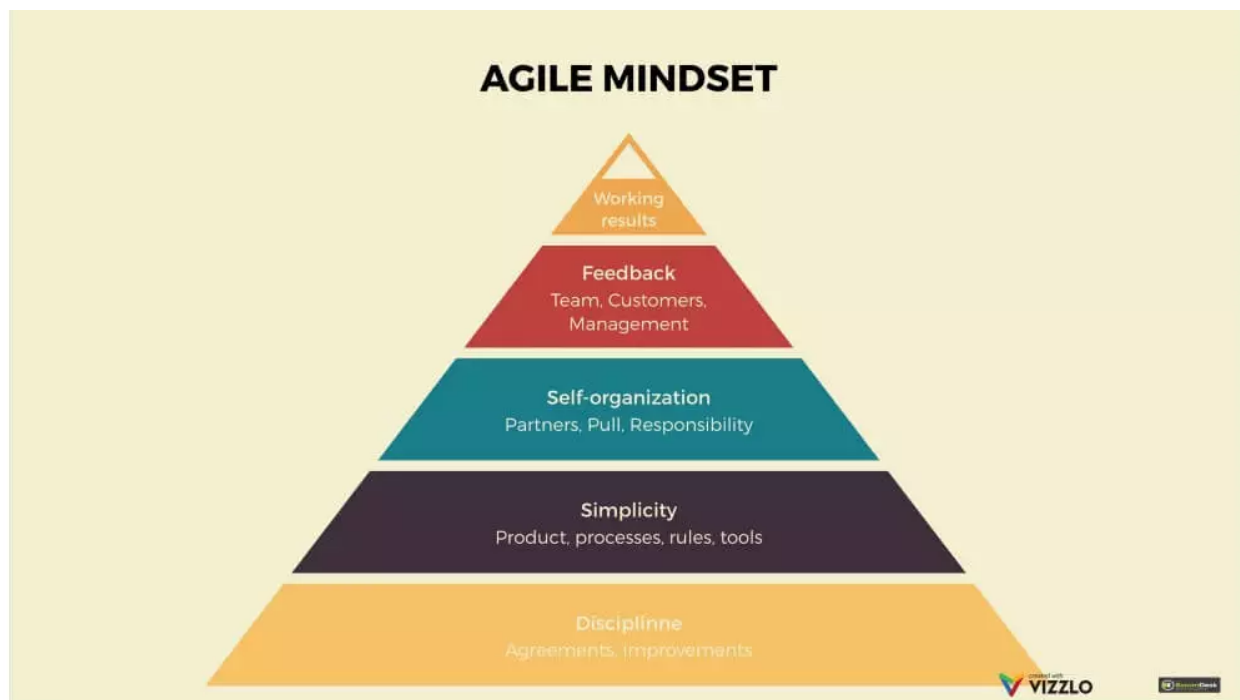
Embracing the Scrum mindset is crucial for successful implementation. The following essentials define this mindset:

**Empirical Process Control:** Scrum relies on transparency, inspection, and adaptation to optimize outcomes. Teams make decisions based on evidence and feedback.

**Self-Organization:** Development Teams self-organize to determine how to best accomplish their work. They are accountable for their commitments.

**Collaboration:** Cross-functional collaboration and open communication are vital for delivering value. Teams and stakeholders work together closely.

**Iterative and Incremental Delivery:** Scrum emphasizes delivering small, valuable increments of work in short cycles to enable fast feedback and continuous improvement.



# APPLICATION AND DOMAIN

Scrum is widely applicable across various industries and projects. Some common application domains include:

- 1) **Software Development:** Scrum originated in software development and is widely used to manage software projects. Its iterative approach suits the ever-evolving nature of software.
- 2) **Product Development:** Scrum can be applied to develop physical products, ensuring that prototypes are built, tested, and refined in iterative cycles.
- 3) **Marketing:** Scrum's flexibility is useful in marketing campaigns, allowing teams to adjust strategies based on real-time feedback and market changes.
- 4) **Education:** Scrum principles are adopted in educational settings to enhance collaboration, adaptability, and student engagement.

## BEST PRACTICE

### **Clearly Define Roles and Responsibilities:**

Ensure that each team member understands their role in the Scrum process. This includes the Product Owner, Scrum Master, and Development Team. Clear role definitions promote accountability and collaboration.

### **Create a Product Backlog:**

Maintain a prioritized list of all the features, enhancements, and bug fixes that need to be addressed. The Product Owner is responsible for maintaining this list and ensuring that it is well-defined and updated regularly.

### **Sprint Planning:**

Conduct well-structured sprint planning meetings where the team collaboratively decides which backlog items to work on in the upcoming sprint. These items should be chosen based on priority and feasibility.

**Time-Boxed Sprints:**

Sprints should have a fixed duration (typically 2-4 weeks) and remain consistent. This time-boxed approach encourages the team to focus on delivering a potentially shippable product increment within each sprint.

**Daily Standup Meetings:**

Hold short daily standup meetings where each team member provides updates on what they've accomplished since the last standup, what they plan to do next, and any impediments they're facing. This fosters transparency and quick issue resolution.

**Incremental Delivery:**

Break down the work into small, manageable units that can be completed within a single sprint. This approach enables the team to deliver incremental value at the end of each sprint.

**Definition of Done (DoD):**

Clearly define the criteria that a user story or task must meet to be considered "done." This ensures a common understanding of what constitutes a complete and shippable piece of work.

**Continuous Improvement:**

Hold regular sprint retrospectives to reflect on what went well, what could be improved, and how to adjust processes for the better. Encourage open communication and implement changes to enhance team productivity and satisfaction.

**Empowered Development Team:**

The Development Team should have the autonomy to self-organize and make decisions on how to best achieve sprint goals. The Scrum Master's role is to support and coach the team rather than dictate actions.

**Collaboration and Communication:**

Foster strong communication among team members, stakeholders, and customers. Regularly involve the Product Owner to ensure that the team is aligned with the product vision and priorities.

**Limit Work in Progress (WIP):**

Encourage the team to focus on a limited number of tasks at a time to prevent bottlenecks and improve overall flow. This boosts efficiency and reduces multitasking.

**Transparency:**

Maintain transparency regarding the project's progress, impediments, and any changes to the plan. This helps stakeholders understand the project's status and make informed decisions.

**Adaptability:**

Embrace change and be open to adapting the project's direction based on feedback, emerging requirements, or market shifts. Scrum's iterative approach allows for flexibility in responding to change.

**Use of Agile Tools:**

Utilize digital tools for backlog management, sprint planning, tracking progress, and facilitating communication. Tools like Jira, Trello, or Azure DevOps can enhance efficiency and collaboration.

**Training and Education:**

Invest in training for the Scrum Master, Product Owner, and team members. A solid understanding of Scrum principles and practices is essential for successful implementation.

Scrum is designed to be adaptable to different contexts, so for a team to be better, they need to tailor these best practices to suit your team's specific needs and project requirements. The team should Regularly review and refine their approach to continuously improve the companies of team Scrum implementation.

## IMPLEMENTATION



An organization must understand Scrum in practice to chart its path. Scrum is a framework, not a set of instructions. Scrum guides the work of product development into iterative, incremental sprints, consistent with Agile development.

Thus, the flow of work in Scrum typically fits into sprints that last between two and four weeks.

### **Step 1: Choose the Product Owner**

The Product Owner is a person who has a vision of the product and is perhaps championing the product in the company. This person will also have a vision of what the SCRUM Team needs to do, perform, or achieve. They must be able to consider risks and take note of possible threats. The Product Owner is the only person who decides what the priorities are and in what order the work will be done.

### **Step 2: Select the Development Team**

To build the right Team for the project, you need to know a great deal about both the pool of employees you have to choose from and the skills required to complete the tasks. Teams may be as small as 3 people for a simple project, while no project is large enough to need more than 9 people. The success of the project depends on the Team more than anything else, so make sure the members get along, communicate well, and are good at organizing themselves.

### **Step 3: Choose a SCRUM Master**

The SCRUM Master will guide the other Team members through the SCRUM framework and help remove any obstacles that might slow the Team down in its efforts.

### **Step 4: Create the Backlog and Prioritize**

The Product Backlog is a list of all the important elements (functionalities) that need to be built, produced, or completed. The Product Owner builds the Backlog from Epics, which in turn are aggregated user stories. This person then prioritizes each item on the backlog and uses it as a checklist during implementation.

### **Step 5: Improve the Product Backlog**

Next, the SCRUM Team needs to evaluate how much effort will be needed to complete the Product Backlog Items (PBI) and assess the following:

- how feasible each item will be with the resources provided (knowledge, skills, people, time, and equipment);
- if the item is small enough that the Team members' effort can be assessed;
- if appropriate milestones can be defined.

Each PBI should be defined so that, when complete, a Team member can present it to any stakeholder who asks. Interestingly, the workload needed to complete the PBI is assessed on the Fibonacci sequence: 1, 2, 3, 5, 8, 13, 21, and so on. The number 1 represents the smallest amount of time and effort, and each successive number represents a multiple of that amount. These numbers are then used to settle tasks and determine the Team's Velocity.

## **Step 6: Sprint Planning Meeting**

The Product Owner, the Scrum Master, and the Development Team work together to plan the Sprints. No Sprint can be longer than 4 weeks—the Definition of Done (the functional product of a Sprint) is adjusted so that every Sprint takes the same number of weeks.

First, the Team members review the Product Backlog, starting with the highest priority PBI. They then break each Backlog Item into tasks.

Next, they define the objective of the upcoming sprint and determine how much they can accomplish during the upcoming Sprint. To do this, they need a deep understanding of each PBI. With experience, the Team members will be able to learn from previous sprints in terms of what's generally possible, how the Team works and how that affects its pace, and what they themselves are capable of. They can then adjust their predictions accordingly.

One of the principles of Scrum is that once the Team agrees on what it thinks is achievable in a Sprint, it becomes mandatory for all Team members. This mutual accountability is an essential feature of SCRUM.

## **Step 7: Complete Tasks and Present Results**

Most commonly, Scrum Teams will make a Scrum board—a large piece of paper—with 3 or 4 columns, or more recently, use one of a number of digital versions. Each PBI is written on a self-stick note and placed in a column to show that it is planned for that Sprint, in progress, or completed (often termed “To Do,” “In Progress,” “Done”). During the Sprint, the Team moves the notes from one column to the next to indicate the current status of each item.

There is also a Burndown Chart, a graph that the Team can use to check how many Stories or points are left in the project versus how many days are left in the Sprint.

## **Step 8: Organize and conduct the Daily SCRUM**

The Daily SCRUM is a meeting that is held at the same time every day for a maximum of 15 minutes. The entire Development Team meets to answer three questions:

What have the Team members accomplished since the last Daily SCRUM in order to move closer to achieving the goal of the current Sprint?

What will the Team members do today to help the Team achieve the goal and overcome difficulties?

Do Team members see any obstacles blocking themselves or the Team from achieving the Goal?

This meeting keeps the goal of the Sprint at the top of everyone’s mind and ensures that all of the Team members are aware of where they are in the Sprint. If things are going well, the positive note is empowering; if things are a bit slow, it’s a chance to note who might need assistance, attention, or just more motivation.



Then, the Team decides together who will do what task and when on that day, taking into account each person's skills, the current situation, and the goal of the Sprint.

## **Step 9: The Sprint Review**

The purpose of this meeting is to present the achievements made during the Sprint. This is not only for Team members—you can and should invite and welcome others to the meeting, such as:

Product Owner

Management

Clients

Other stakeholders

The Scrum board is useful again at this step for presenting the tasks that have been completed. The Scrum Team should also demonstrate that the “Definition of Done” has been fulfilled and show what can be immediately delivered to the Client without further effort. (If the Definition of Done has not been met, you should not be having this meeting yet.) Remember, this is not necessarily the same as having a complete product. Each Definition of Done is something that is fully functional in terms of the tasks completed so far.

## **Step 10: The Retrospective**

Finally, the Scrum Team meets alone to discuss the last Sprint and, if applicable, to compare it to previous Sprints. The idea is to appreciate what went well and give due credit, and then to consider what could have gone better. This is where the Team members have the chance to learn from each other and find ways in which they can grow as professionals. You should especially consider the following:

What should be done better in the next Sprint?

What improvements can the Team implement right away in the next Sprint?

Why did the work go the way it did?

If something was forgotten, why did that happen?

Could the Team work faster? If so, what changes would help that happen?

It is important that each Team member take responsibility for his or her performance on the Sprint and for the effect on the results. The members must also have the courage to raise difficult issues and look for solutions. The Scrum Master and the Development Team can then agree on what to do to eliminate any problems in the next Sprint.

The product owner must first establish a clear vision for the work to be done. The product vision must include elements like the product name and description, the value the work will deliver back to the business and challenges that might occur during development.

Then, Scrum sprint iterations typically follow the same basic pattern:

- Begin a new sprint.
- Refine the product backlog.
- Determine the sprint's goal.
- Plan the work.
- Develop the sprint backlog.
- Conduct daily Scrum meetings.
- Conduct a sprint review.
- Conduct a sprint retrospective.

Before development work can begin, the organization should have the roadmap in place. Pre-development work entails backlog refinement, user story creation, defining done, establishing clear sprint goals and conducting the first planning meeting of the sprint. When the development work begins, it typically follows a quality-centric approach that is native to Scrum. Scrum is a framework that serves as a container for other engineering processes. Therefore, test-driven development, continuous integration, and automated testing all logically fit within the Scrum framework and help to ensure the quality of the software.

Daily Scrum meetings help keep the team informed. On the final day of the sprint, the Scrum team holds a sprint review to compare the work to the goals. Afterward,

a sprint retrospective helps determine what went well and what needs to be improved in the future. The goal is team-wide incremental improvement.

At the sprint's completion is either a potentially shippable product that agrees with the team's definition of done, or there is unfinished work that gets added to the backlog. The latter adds more work to the team's plate, which can increase technical debt that makes it more difficult to fix defects. Other challenges can also occur during Scrum sprints. As with any development methodology, Scrum requires organizational buy-in and structure to succeed. It takes a fundamental culture change to succeed with Scrum.

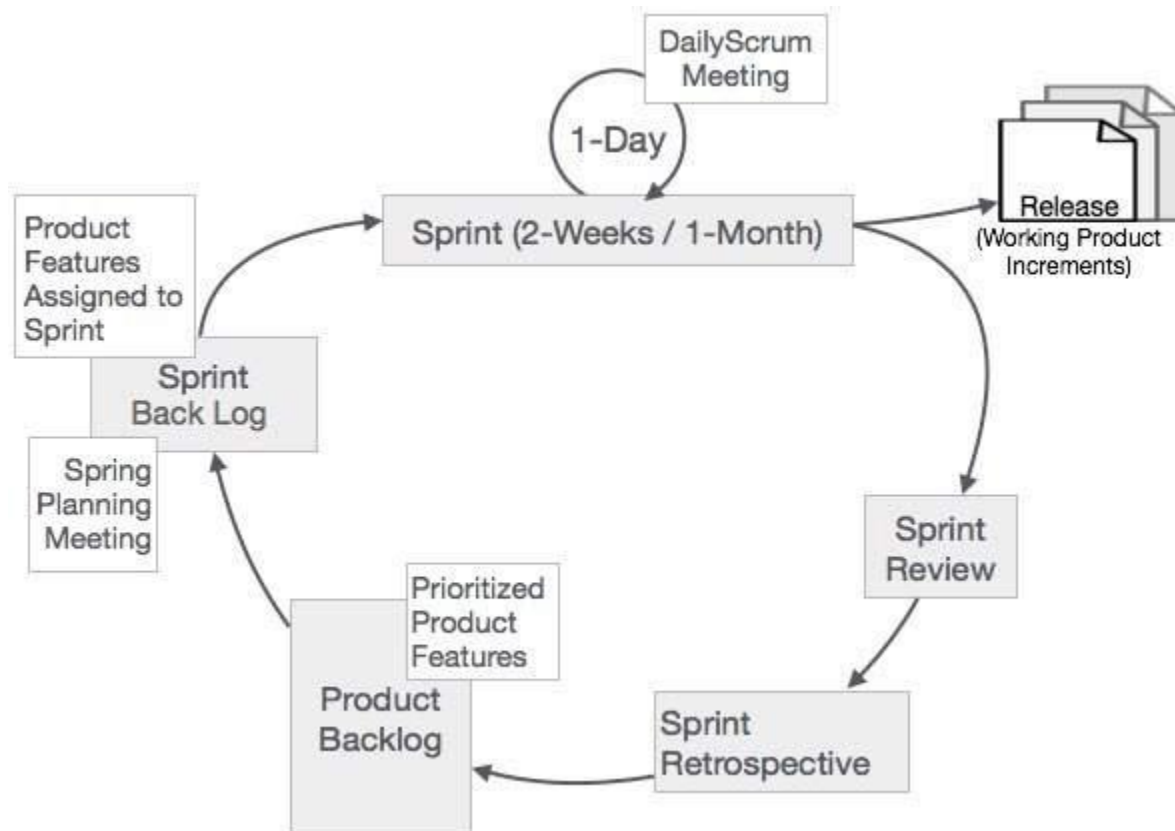
Scrum implementation killers are lack of executive sponsorship and buy-in to underinvestment, including in education and communication within the organization. And don't tackle Scrum without a strong Agile mindset. It's not just organizational points of failure to avoid. Scrum implementation challenges, such as technical debt, hurt the team's ability to consistently release quality software.

Without resources in place, such as a Scrum center of excellence, the team cannot address challenges that pop up. It won't be able to correct Scrum antipatterns that continue sprint after sprint. Some Scrum antipatterns include;

- Roles that are not part of Scrum;
- Focus on the wrong product backlog items;
- Inappropriate priorities;
- Directing instead of leading;
- Non-value-added activities;
- Team burnout;
- Failure to provide full transparency;
- Development done beyond economic value;
- Missed market segment opportunities;
- Pushing deliveries beyond capacities;
- Failure to work as a team; and
- Failure to evolve the product incrementally.

While hardly an overnight process, the effort to implement Scrum in all of your development projects will certainly pay off. Scrum methodology offers you not just an efficient way to complete projects but also a powerful motivational boost in terms of interpersonal accountability and individual growth. There are also the benefits from working in a cross-functional Team rather than entirely within a single department.

## CASE STUDY



### Scrum Case Study:

### Byte Technologies and Kfood App Development

#### Background:

Byte Technologies is a software development company specializing in mobile applications. One of their recent projects is the development of "Kfood," a food ordering application that aims to revolutionize the way people order and enjoy food from their favorite restaurants. The company decided to use the Scrum framework to manage the development process and deliver a high-quality app to their customers.

### **Team Structure:**

The development team consists of the following members:

#### **Product Owner (PO):**

Responsible for defining and prioritizing features, maintaining the product backlog, and ensuring alignment with business goals.

#### **Scrum Master (SM):**

Facilitates the Scrum process, removes impediments, and ensures that the team follows Scrum practices.

#### **Developers:**

Cross-functional team members responsible for designing, coding, testing, and delivering the app's features.

#### **Sprint Duration:**

Three Months sprints were chosen to strike a balance between regular progress updates and sustainable development pace. The sprints were divided into **6 Sprints**  
Each sprint was to last two weeks

#### **Week 1-2>> Sprint 1:**

Goal: Set up the project infrastructure and create basic user authentication.

Best Practices Applied:

Sprint Planning: The team conducted thorough sprint planning, breaking down high-priority features into actionable tasks and estimating their effort.

Daily Stand-ups: Daily stand-up meetings were held to discuss progress, blockers, and upcoming tasks. The focus was on transparency and quick problem-solving.

Sprint Review: The team presented the completed user authentication feature to the PO and stakeholders, gathering feedback for further improvements.

### **Week 2-3>> Sprint 2:**

Goal: Implement restaurant browsing and menu selection functionalities.

Best Practices Applied:

Refining the Backlog: The PO continuously refined the product backlog, adding new features and adjusting priorities based on user feedback and changing market trends.

Incremental Development: The team followed an incremental approach, delivering a basic version of restaurant browsing before adding more complex features.

Pair Programming: Developers engaged in pair programming, enhancing code quality and knowledge sharing within the team.

### **Week 4-5>> Sprint 3:**

Goal: Build the ordering and payment system.

Best Practices Applied:

Cross-functional Collaboration: The developers collaborated closely with the PO to ensure the ordering and payment system aligned with the user's needs and business requirements.

Continuous Integration: The team used continuous integration practices to integrate code changes frequently, reducing integration challenges and improving software stability.

### **Week 6-7>> Sprint 4:**

Goal: Testing, bug fixing, and final polish.

Best Practices Applied:

Quality Assurance: Dedicated testing cycles were allocated to identify and fix bugs, ensuring the app's reliability and functionality.

User Acceptance Testing (UAT): A UAT phase involves selected end-users testing the app to gather real-world feedback and identify any remaining issues.

### **Week 8-9>>Sprint 5:**

Goal: Deployment and launch preparation.

Best Practices Applied:

Release Planning: The team worked closely with the operations team to plan the app's deployment and launch activities, ensuring a smooth transition to production.

Documentation: Comprehensive documentation was created to facilitate the deployment process and provide support to the operations team.

### **Week 10-11>>Sprint 6:**

Goal: App launch and initial post-launch support.

Best Practices Applied:

Monitoring and Feedback Loop: The team established a feedback loop to monitor the app's performance and gather user feedback post-launch, enabling quick adjustments and improvements.

Results and Benefits:

The Kfood app was successfully launched within the planned time frame, meeting user expectations and business goals.

Scrum practices allowed for continuous feedback and adjustments, leading to a highly usable and well-received application.

Cross-functional collaboration, regular communication, and iterative development led to a robust and reliable app.

The Scrum framework enabled Byte Technologies to adapt to changing requirements and market dynamics efficiently.

**Conclusion:**

Through the adoption of Scrum best practices, Byte Technologies effectively developed and launched the Kfood food ordering application, demonstrating the benefits of iterative development, close collaboration, and continuous improvement. The success of the project laid the foundation for future endeavors, solidifying Scrum as the preferred methodology for delivering innovative and customer-focused solutions.

## QUESTION AND ANSWER

**Following are some FAQs regarding Scrum -**

**Question:** Why do they call this methodology scrum or people often ask, is scrum an acronym or something?

**Answer:** And the answer is no. It is actually inspired by a scrum in the sport of rugby. In rugby, the team comes together in what they call a scrum to work together to move the ball forward. In this context, Scrum is where the team comes together to move the product forward.

**Question:** What is the difference between Scrum and Agile Development?

**Answer :** Agile Development is a software methodology, whereas Scrum is one of the process frameworks that follows Agile.



**Question:** Are Sprints and Iterations the same?

**Answer :** Both Sprints of Scrum and Iterations of Iterative Incremental model deliver working product increments. The difference is that Life Cycles of Sprint and Iteration are different. Sprints are time-boxed, while Iterations are not. Duration of Sprints is much less compared to durations of Iterations.

**Question:** Is Scrum Master a job title or a role that someone with an existing job title fills?

**Answer :** Scrum Master is a role that someone with a job title fills. Normal practice is that the person playing the role of project manager plays the ScrumMaster's role as well.

**Question:** Can Product Owner and ScrumMaster's roles be played by the same person?

**Answer :** No, since the ownership differs. Product Owner takes care of the Product Backlog, Prioritization of User Stories, and Validation of the working product increment with the user stories allocated to the Sprint.

**Question:** Is it that Scrum Projects need not have any Documentation?

**Answer :** No. Scrum Projects, like any other Projects require documentation such as user stories, design, test cases, etc.

