

PP19

The Scrum Mindset

Flo Ratcliffe
flo.ratcliffe@instem.com

ABSTRACT

We've all heard of Agile and Scrum, perhaps as techniques for managing projects. But do any of us really know what these valuable principles and processes comprise? This poster provides an introduction to Agile and Scrum, providing the audience with a clear understanding of the meaning of familiar terms and how to apply the principles and processes to your project, and how to optimise your project's efficiency and outcomes.

Written by three Associate Project Managers at Instem, this poster highlights how to effectively utilise Scrum techniques in an easy-to-understand way, allowing anyone from any department to make the most of Scrum and Agile techniques.

THE AGILE PRINCIPLES

✔ Customer Satisfaction	⚙ Working Software
🔄 Welcome Change	🧠 Constant Pace
👉 Delivery Frequently	💡 Good Design
👥 Working Together	🔗 Simplicity
🏢 Motivated Team	📋 Self Organisation
🗣 Face-to-Face	🔄 Reflect & Adjust
INDIVIDUALS AND INTERACTIONS over processes and tools	
WORKING SOFTWARE over comprehensive documentation	
CUSTOMER COLLABORATION over contract negotiation	
RESPONDING TO CHANGE over following a plan	

WHAT'S IN IT FOR ME?

1. PREDICTABILITY:

 - Unknowns cannot be proactively managed
 - Being able to predict potential risks in a project is key to its success
 - A clear communication schedule increases communication, which leads to increased predictability.
2. FLEXIBILITY:

 - The final product is closely aligned with requirements and expectations
 - Real-time customer feedback
 - Frequent opportunities to adjust the product
 - The project plan offers estimated dates and durations
 - Adjust during the project lifecycle
 - Discussed with the client
 - Not a strict timeline
3. INFORMATION AVAILABILITY:

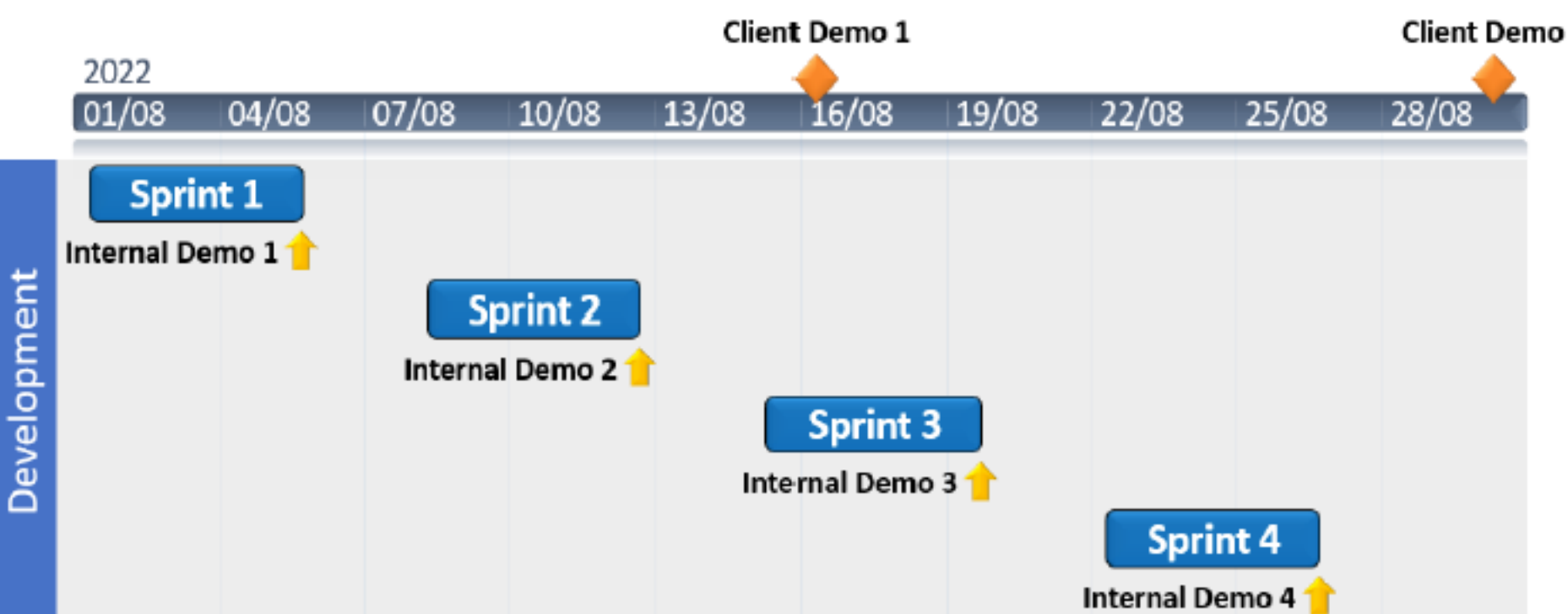
 - Agile management encourages a constant feedback loop
 - The whole team is kept up-to-date with project information
 - Well-organised documentation is paramount to project success
 - Ensures there are no grey areas or a singular points of failure
 - It is equally important to have clear, concise documentation to avoid ambiguity
4. QUALITY CODE:

 - Flexibility and collaboration are conducive to quality code
 - Ensures it is well-aligned with client expectations
 - Frequent internal reviews of deliverables
 - Code quality is the responsibility of the whole team, not just developers or testers
5. WELL-DEFINED PROCESS:

 - Ensure the whole team knows:
 - Their role
 - The plan
 - The project goal

SPRINTS

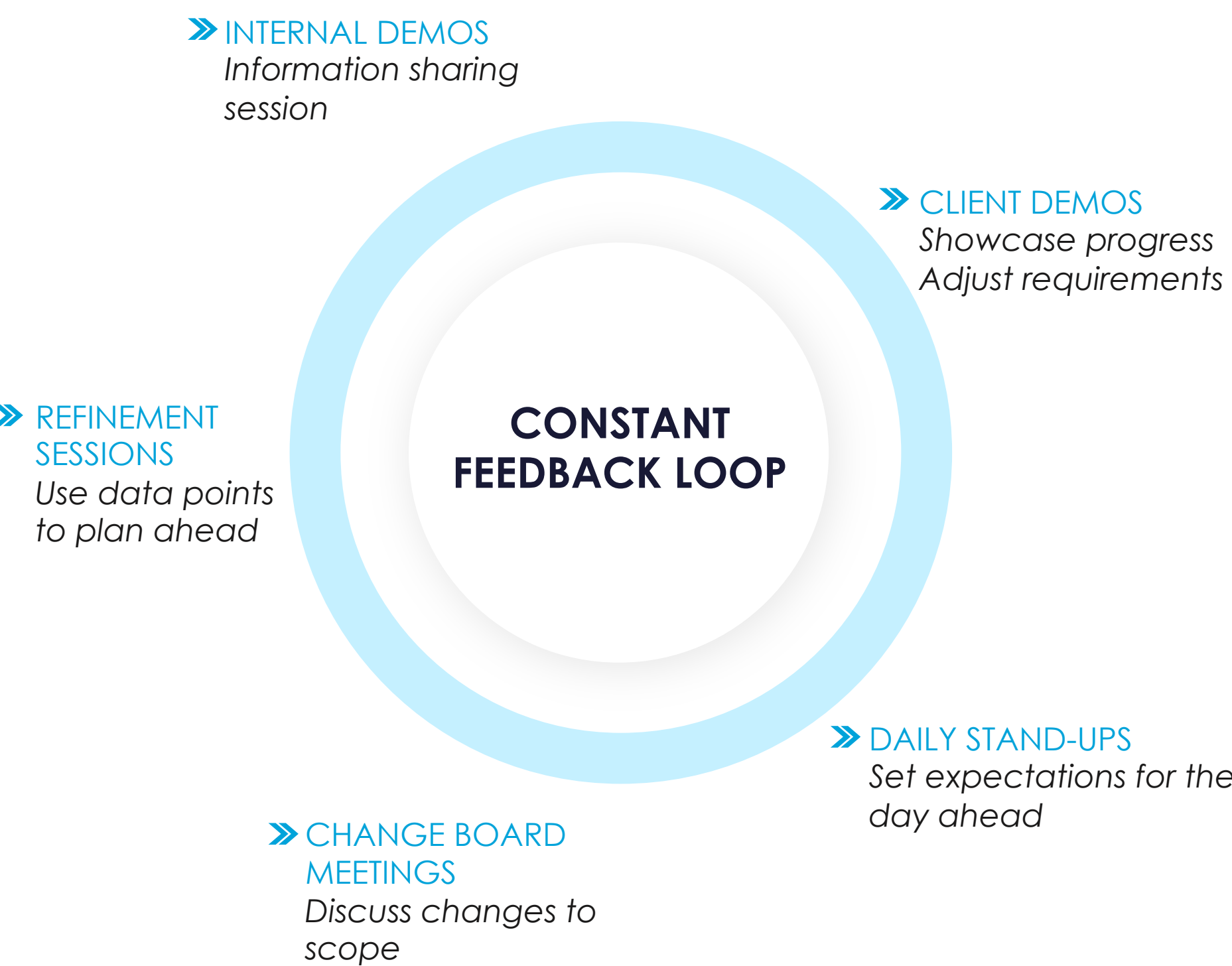
Sprints are delivery iterations, where the tasks making up the project have been broken down into sections. These sections are known as sprints and are usually scheduled to last one or two weeks—it's better to have a higher number of short sprints than fewer, longer sprints. This is because at the end of each sprint, the work is reviewed by the project team and the customer. A higher number of sprints means a higher number of touchpoints with the customer, and more allowances for feedback and adjustments. Internal demonstrations, or 'demos', should be performed at the end of each sprint, while client demos can be a little less frequent depending on availability and functionality development.



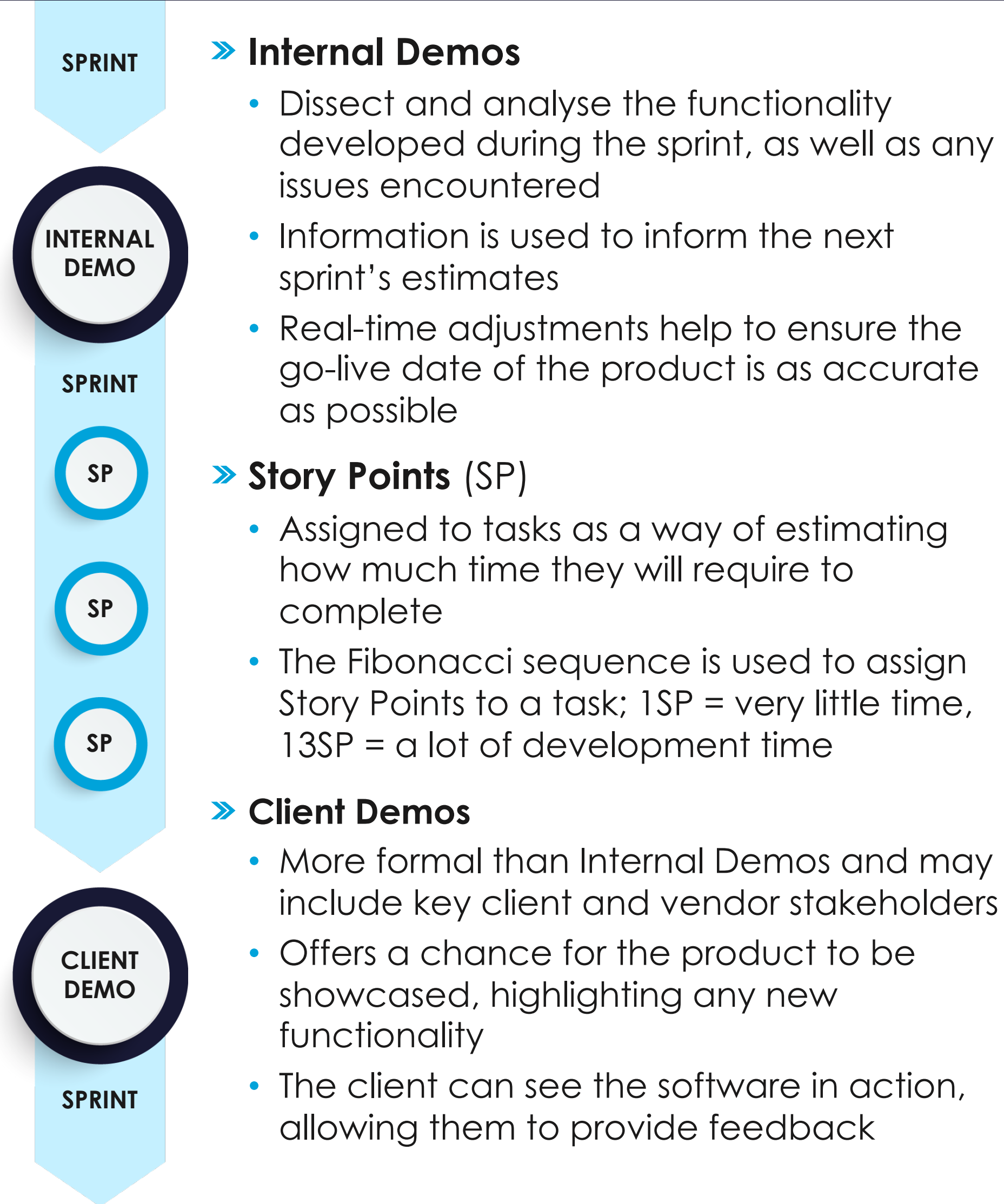
CONSTANT FEEDBACK LOOP

Agile methodologies facilitate a **Constant Feedback Loop** to ensure maximum efficiency and alignment with the requirements. This is done through a multitude of meetings and reviews of the software, both internal and external. In projects where the customer is not involved throughout, the written requirements are under high scrutiny and must be extremely well-written to ensure the end product matches the customer expectations.

The most common feedback opportunities are **Internal Demos, Client Demos, Daily Stand-Ups, Change Board Meetings, and Refinement Sessions.**



DEMOS & STORY POINTS



TICKETS, USER STORIES, & EPICS

Pieces of functionality and bugs are recorded as **Tickets** in a tracking system such as Atlassian's 'Jira' software. Tickets are assigned to a specific person, usually a developer, and given a unique ID, description, Story Point estimation, priority, and other information as deemed necessary for the success of the project.

Tickets also have to be assigned a status. This process is similar to a Kanban Board, where the ticket moves left to right sequentially through columns as its status changes.

Ticket workflow example:



Collectively, tickets make up User Stories. A **User Story** represents a piece of functionality that can be developed and tested in one sprint. Multiple User Stories may be intended to work in parallel to implement more complex functionality. These groups of User Stories are called **Epics**. While User Stories should not extend further than one Sprint, Epics can span over multiple sprints. Ticketing systems can easily track and display Epics, allowing for larger functionalities to be developed over multiple sprints.

THE FUTURE OF AGILE

While the Manifesto for Agile Software Development was published in 2001, fragments of Agile methodology, known alternatively as lightweight software development methods, have roots as far back as 1991. With so many branches of Agile (Scrum, Kanban, Extreme Programming, and Lean Development just to name a few) it's no wonder that lightweight development methods are constantly evolving. The Agile mindset particularly lends itself to futuristic thinking, with one of its main focuses being Continuous Improvement.

As with most industries, software development lifecycles are tending towards an increase in automation wherever possible. In areas where this is not yet possible, there is an emphasis on waste prevention, and efficiency.

An example of this is the frequent check-ins with clients that the Scrum methodology encourages. These check-ins result in far less potential for waste from occurrences like re-development due to misaligned or misunderstood requirements.

CONCLUSION

In a world with ever-increasing demand for faster, better, and more powerful technology, businesses must adopt delivery methodologies that allow them to meet and surpass this demand.

Ultimately, the most effective project delivery method will be that which the team has the most confidence in. This, however, may not be the processes most well-known to them. Businesses can, and should, frequently evaluate the efficiency of their methodologies and delivery processes to ensure they are employing approaches conducive to the best possible end product. Not only that, but they must assess if the practises they are employing are the most time-efficient, cost-effective path forward. In turn, this will allow for positive client -vendor relationships and ensure future business opportunities.

With continuous improvement at its core, Agile fosters an environment of efficiency with a customer-focused and forward-thinking outlook. While it may not be right for every project, this ever-evolving group of methodologies is sure to be around for a while, adapting to the demands of the modern world.

A massive thank you to Will Decker and Charlie Fitzpatrick for their contributions to this poster, and their contributions to my knowledge about Agile and Scrum!