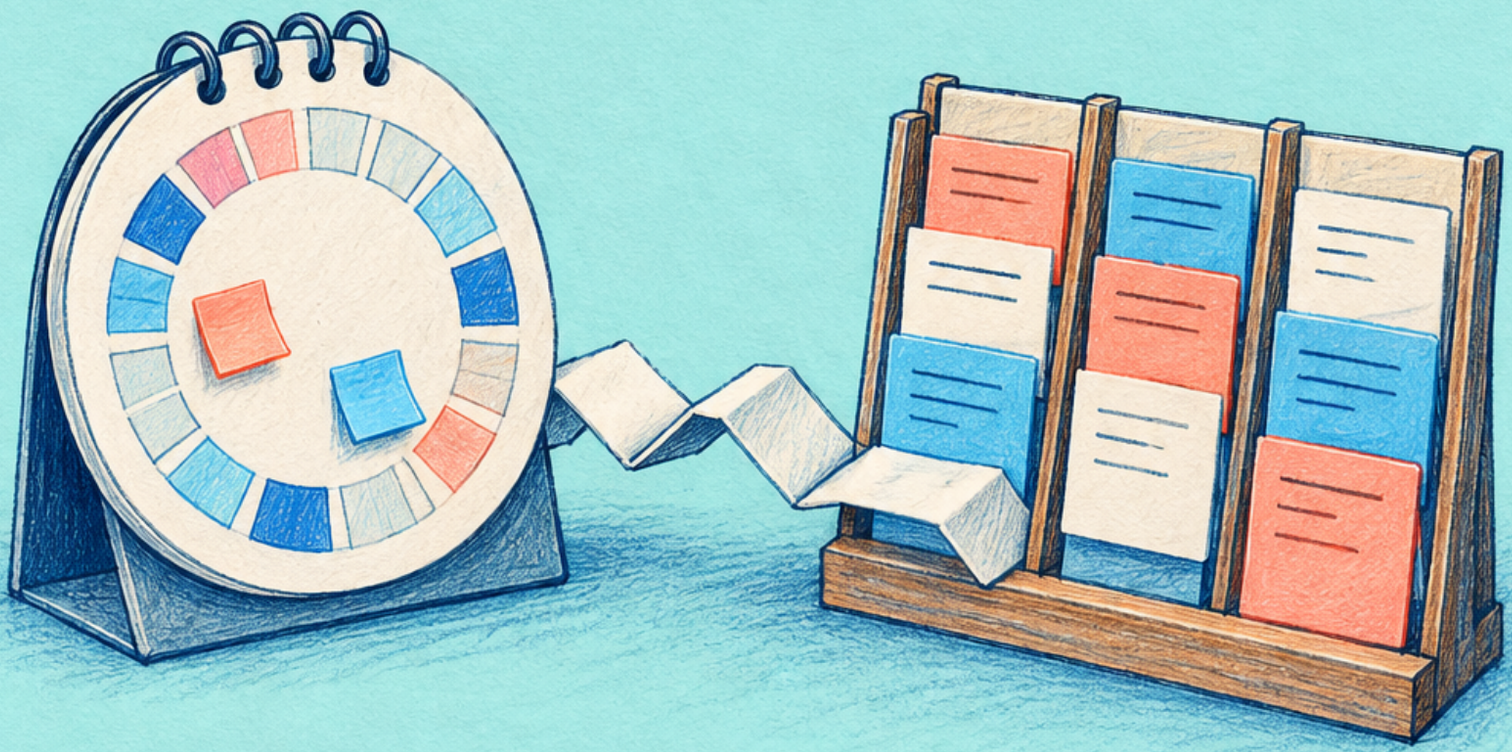
Scrum vs. Kanban: Key Differences Explained Through Examples



Compare the purpose first

Two teams can use cards while organising delivery in different ways.

Read the definitions

Scrum is a framework for complex product work, with specific accountabilities, events, artifacts and commitments. Kanban is a strategy for improving the flow of value through a defined process.

Support-product case

A team builds a help portal and handles urgent service fixes. The important question is how to learn about the product while managing incoming work and existing commitments.

Start with the problem

A familiar board layout does not establish whether either approach is helping. Identify the delivery difficulty and the decisions the team needs to improve.

TAKE IT TO YOUR PROJECT

Name the problem you want the approach to address.

Timeboxed learning and flow

A Sprint and a workflow boundary measure different parts of delivery.

Scrum lens

A Sprint is a fixed-length container, one month or less, in which the team works toward a Sprint Goal and creates a usable Increment.

Kanban lens

Define where items start and finish and manage their movement under explicit policies. Measure elapsed time and completion counts through those boundaries, with appropriate reviews.

Support-product case

The help-portal Sprint aims to make password recovery usable. Meanwhile the team can inspect the age of a carrier-login defect in its workflow to decide whether it needs attention.

TAKE IT TO YOUR PROJECT

Describe the goal and the flow boundary you want to inspect.

Accountability and existing roles

An operating approach needs clear decision rights even when roles differ.

Scrum structure

Scrum specifies a Product Owner, Scrum Master and Developers. Those accountabilities cover value, Scrum effectiveness and creating the usable result.

Kanban context

The Kanban Guide defines practices for a Kanban system without requiring the three Scrum accountabilities. A team still needs clarity about decisions, policies and improvement responsibilities.

Support-product case

The service manager may own priority rules in an existing workflow. If the team also uses Scrum, keep Product Owner accountability and the Sprint Goal clear rather than creating competing directions.

TAKE IT TO YOUR PROJECT

Identify who can change priority in your team's actual system.

Goals and work selection

Starting an item should serve a coherent objective and respect available capacity.

Scrum planning

During Sprint Planning the Scrum Team creates a Sprint Goal, selects work and develops an initial plan. Developers adapt the Sprint Backlog as they learn.

Kanban selection

Use explicit policies and WIP control to decide when work may enter or move. Replenishment can use a chosen cadence without being a required Sprint event.

Support-product case

A password reset improvement supports this Sprint's goal. An unrelated styling request should be evaluated against that goal and the team's agreed policies before it enters active work.

TAKE IT TO YOUR PROJECT

Write the rule for pulling the next item into your workflow.

Work in progress

Too many started items can hide waiting and slow completion.

Use the measure

WIP counts items that started but have not reached the defined finish. A WIP limit is an explicit control on how much work may occupy a state or set of states.

Support-product case

Build has 3 items and Review has 2. Total WIP is 5 under these boundaries. If Review's limit is 2, a newly coded item cannot simply join it without addressing the policy.

Use both ideas

A Scrum Team can apply Kanban practices to its Sprint workflow. A Sprint itself does not automatically create a useful WIP limit in every step.

TAKE IT TO YOUR PROJECT

Count all open items, including blocked work, against your defined boundary.

The All-in-One PM Templates Bundle

940+ Premium Templates • One Payment • Lifetime Access

$49

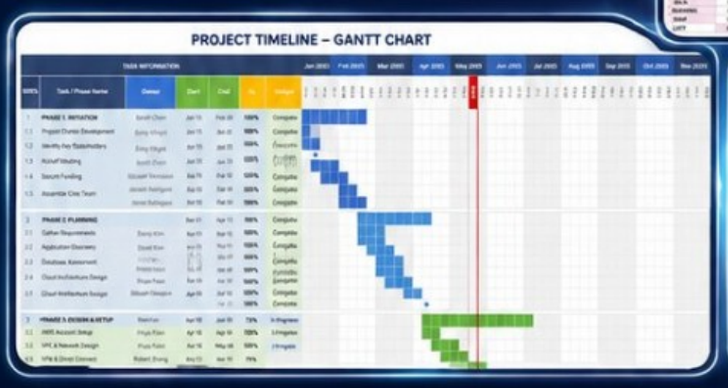
SAVE 94%

Worth $3,300+











EXCEL & GOOGLE SHEETS | PROFESSIONAL & RELIABLE | AUTOMATED DASHBOARDS | SAVE TIME & STAY ON TRACK | INSTANT ACCESS DOWNLOAD

GET LIFETIME ACCESS

Handling new information

A new request needs a decision, not an automatic rule based on a label.

Scrum response

Clarify the request with the Product Owner and Developers. Scope can be renegotiated as learning occurs, while protecting the Sprint Goal and maintaining quality.

Kanban response

Apply explicit selection, priority and exceptional-work policies while managing capacity. A continuously visible queue does not mean every new request must interrupt active work.

Support-product case

A security incident may need urgent attention. Make its effect on current work visible and use the agreed decision route rather than quietly abandoning existing commitments.

TAKE IT TO YOUR PROJECT

Define how an urgent item will be evaluated and authorised.

Events and chosen cadences

A review schedule should produce useful inspection and adaptation.

Scrum events

Sprint Planning, Daily Scrum, Sprint Review and Sprint Retrospective occur within the Sprint, each with a defined purpose. They support product progress and improvement.

Kanban cadences

Teams choose appropriate opportunities to inspect demand, flow and improvement. The Kanban Guide does not prescribe the complete Scrum event set.

Support-product case

A daily flow review can examine the oldest stuck ticket. A Sprint Review can examine the delivered password-recovery outcome with stakeholders. The questions can complement each other.

TAKE IT TO YOUR PROJECT

State the decision that makes one of your recurring meetings worthwhile.

Quality and completion

A finished-looking card still needs a clear quality boundary.

Scrum commitment

The Definition of Done describes the quality measures for the Increment. Work failing it cannot count as part of that Increment.

Workflow policy

A Kanban system defines its workflow and finish. The definition should support meaningful value and consistent measures. Make the required checks visible.

Support-product case

A reset button works in a demo but fails the agreed email-verification check. Record it as unfinished under the applicable quality definition rather than reporting completion.

TAKE IT TO YOUR PROJECT

Identify the evidence needed before an item crosses your finish boundary.

Compare the metrics

Choose measures that help answer the question in front of the team.

Scrum context

Scrum does not require story points or velocity. A team can use suitable evidence to plan and inspect progress toward goals, quality and value.

Kanban flow measures

The Kanban Guide identifies WIP, throughput, work item age and cycle time as core flow metrics. Keep their boundaries and units clear.

Support-product case

Three fixes finished this week, giving throughput of 3 items per week. An open fix started Monday and is still active Thursday; inspect its age before starting more work.

TAKE IT TO YOUR PROJECT

Explain what one flow measure can tell you and what evidence is still missing.

Estimate without false precision

A commitment should reflect the uncertainty and evidence behind the forecast.

Compare the evidence

A Scrum Team may use sizing and delivery experience when selecting Sprint work. Flow-based forecasting can use the observed completion times or counts of comparable work.

Support-product case

Seventeen of twenty comparable fixes finished within six days of starting. That supports a statement about the observed proportion, with assumptions about future comparability and conditions.

State the uncertainty

A time-and-probability expectation for one item is different from a guarantee for every item or a complete project deadline.

TAKE IT TO YOUR PROJECT

Write a forecast with its boundary, unit and assumption.

Release and value delivery

A calendar event and a customer's ability to use the result need not coincide.

Scrum delivery

A usable Increment may be delivered before the Sprint ends. The Sprint Review is a working inspection event, not a mandatory release gate.

Kanban delivery

The workflow can represent delivery, but policies determine what finish means. A deployment queue can still delay usable value after coding is complete.

Support-product case

A verified security fix is ready Wednesday. Check release policy, operating readiness and the applicable authority. Do not infer a release restriction solely from the word Sprint.

TAKE IT TO YOUR PROJECT

Find the actual policy governing release in your environment.

Deal with a bottleneck

A board becomes useful when the team changes a decision from its evidence.

Read the situation

Build finishes 6 fixes a week and Review clears 4. With comparable counts and no other flows, work accumulates before the slower step.

Support-product case

Help clear review work, improve review capability or reduce incoming starts. Moving cards into a new column without changing capacity can hide the same waiting.

Use the approaches together

The Scrum Team can inspect how the constraint threatens its goal while using flow data to choose a concrete action and review the result.

TAKE IT TO YOUR PROJECT

Choose one way to reduce waiting without lowering quality.

Scaling across teams

Shared outcomes need coordination beyond separate team boards.

Map dependencies

Identify interfaces, common quality expectations, shared specialists and timing assumptions. A local throughput improvement can create more waiting for another team.

Support-product case

The portal team needs identity-service changes from a second team. Coordinate the interface and integration evidence rather than assuming two healthy boards imply a ready service.

Keep ownership visible

When Scrum Teams work on one product, apply the shared product expectations in the framework. With Kanban, define the connected workflows and decision policies clearly.

TAKE IT TO YOUR PROJECT

Name one cross-team item whose true finish is currently unclear.

Choose a useful starting point

Select an approach from the work and organisational conditions.

Examine the need

Consider product learning, demand variability, interruption patterns, team capability and decision authority. Scrum may support a coherent Sprint learning cycle; Kanban may improve an existing workflow.

Support-product case

The team wants better product feedback and less hidden service waiting. It considers Scrum for the product work with Kanban practices to control and inspect the workflow.

Run a bounded change

Define the expected improvement, baseline and review point. Avoid assuming a renamed board or new meeting schedule establishes the desired behaviour.

TAKE IT TO YOUR PROJECT

State which problem your first change is expected to improve.

Work an interruption scenario

Apply the goals, policies and measures to one concrete situation.

Scenario

A team is developing password recovery toward a Sprint Goal. Review is at its limit. A carrier-login incident arrives and affects many current users.

Reason through it

Assess urgency and impact with the appropriate decision owners, inspect current work and use the agreed exceptional-work policy. Make the effect on the Sprint Goal visible and adapt the plan.

Check the result

Track whether the incident is resolved and existing work returns to flow. Review whether the policy handled the situation without hiding quality or unfinished work.

TAKE IT TO YOUR PROJECT

Write the first three questions you would ask before reallocating the team.



Learn from 350+ top universities and companies

ILLINOIS

Duke UNIVERSITY

Google

UNIVERSITY OF MICHIGAN

IBM

VANDERBILT UNIVERSITY

JOHNS HOPKINS UNIVERSITY

Invest in your career

Explore new skills

Access 10,000+ courses in AI, business, technology, and more.

Your comparison desk check

Use the comparison to explain the team's operating choices.

Check the system

Can people name the relevant goal, who orders work, when it may start, how WIP is controlled, what Done means and where feedback changes decisions?

Support-product case

The team retains Scrum's accountabilities and events, defines a visible review limit, tracks age and cycle time, and improves the workflow while inspecting progress toward product goals.

Review the evidence

Look for usable outcomes, clearer decisions and reduced harmful waiting. Explain which practices are framework requirements and which are choices made for this context.

TAKE IT TO YOUR PROJECT

Complete the six questions with your team and identify one unresolved policy.