Kanbanstudy
Targeting success

A Guide to the

KANBAN BODY OF KNOWLEDGE (KBOK™ GUIDE)

8. Execute

The Practical Implementation Guide for Managing Workflows using Kanban

(Includes Examples from popular digital Kanban tools, facilitates integration with other Agile frameworks, and recommends ways to use AI for increased productivity.)

8 EXECUTE

This chapter includes the process related to execution of a Kanban initiative: Get Work Done.

Execute, as defined in *the Kanban Body of Knowledge (KBOK™)*, is applicable to the following:

- Kanban initiatives in *any* industry
- Products, services, or any other results to be delivered to Stakeholders
- Kanban Initiatives of any size or complexity

Kanban can be applied effectively to any initiative in any industry—from small initiatives or teams with as few as two team members to large, complex initiatives with up to several thousand members in several teams. To facilitate the best application of the Kanban framework, this chapter identifies inputs, tools, and outputs for each process as either "mandatory" or "optional." Inputs, tools, and outputs denoted by asterisks (*) are mandatory, or considered critical to success, whereas those with no asterisks are optional. It is recommended that the inexperienced Kanban practitioners and those individuals being introduced to the Kanban framework and processes focus primarily on the mandatory inputs, tools, and outputs; while experienced Kanban professionals, including Sponsors and relevant Stakeholders strive to attain a more thorough knowledge of the information in this entire chapter.

This chapter is written from the perspective of a single Kanban initiative within the company or a specific department and follows:

- The *Setup* chapter, where the Kanban function is established for the entire organization or a specific department.
- The *Plan* chapter, where the Kanban Team is formed, stakeholders are identified and optimized Kanban Workflows are created

The outputs from this chapter will serve as valid inputs to *Enhance* (Chapter 9) which is discussed in the next chapter.

In the Execute phase, the Kanban team regularly reviews the Kanban Backlog to prioritize upcoming work and assesses the Kanban Board to track tasks in progress. The team ensures that work items are clear, manageable, and ready to be pulled into the relevant To Do column when there is capacity. As tasks move through the Workflow, from To Do to In Progress and finally to Done, the team focuses on completing each item efficiently. Regular reviews help identify bottlenecks, optimize flow, and maintain a steady pace of output. Completed Work Items are the primary output, driving the team's overall progress. The goal of the Execute phase is to get work done for the Kanban Initiative and create Completed Work Items.

It is also important to realize that although all phases and processes are defined uniquely in the Kanban Body of Knowledge, they are not necessarily performed sequentially or separately. At times, it may be more appropriate to combine some phases and/or processes, depending on the specific needs of each initiative.

Figure 8-1 provides an overview of the Execute phase process, which is as follows:

8.1 Get Work Done—In this process, the Kanban Team regularly reviews the Kanban Backlog to prioritize upcoming tasks and tracks progress on the Kanban Board. Work items are pulled into the To Do column when there's capacity. As tasks move through the Workflow—from To Do to In Progress to Done—the team focuses on efficient completion. Regular reviews help identify bottlenecks, optimize flow, and maintain a steady output. Completed Work Items drive the team's overall progress.

Figure 8-1 shows all the inputs, tools, and outputs for processes in the Execute phase.

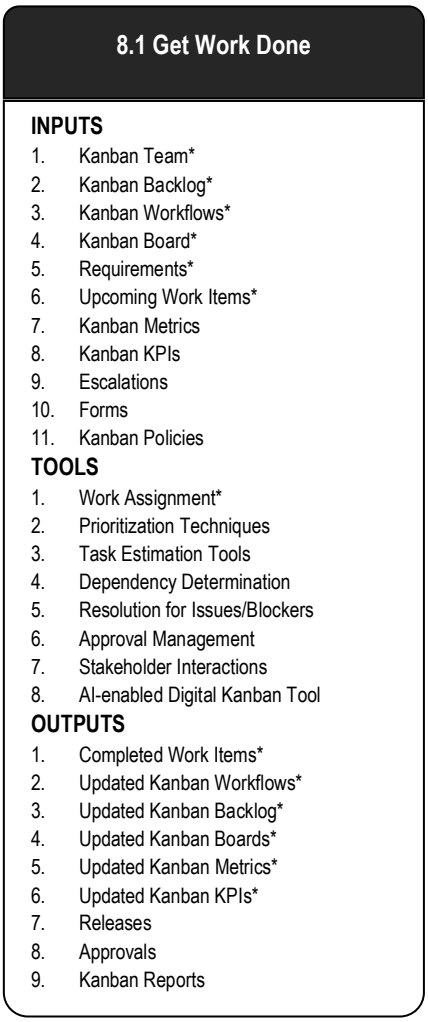

Figure 8-1: Overview of Execute Phase Processes

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

Figure 8-2 below shows the mandatory inputs, tools, and outputs for processes in Execute phase.

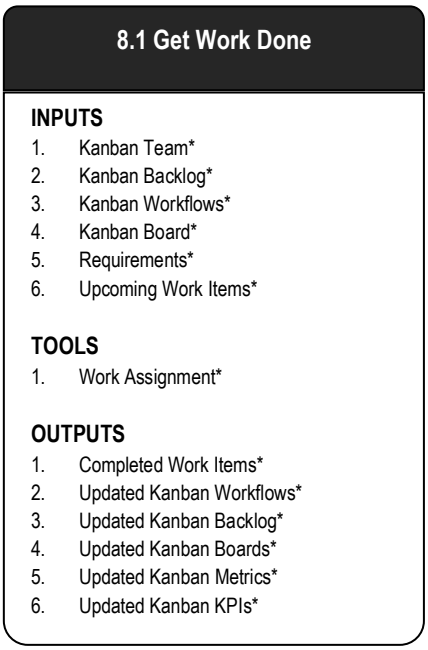

Figure 8-2: Overview of Execute Phase Processes (Essentials)

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

8.1 Get Work Done

In this process, the Kanban team regularly reviews the Kanban Backlog to prioritize upcoming work and assesses the Kanban Board to track tasks in progress. The team ensures that work items are clear, manageable, and ready to be pulled into the relevant To Do column when there is capacity. As tasks move through the Workflow, from To Do to In Progress and finally to Done, the team focuses on completing each item efficiently. Regular reviews help identify bottlenecks, optimize flow, and maintain a steady pace of output. Completed Work Items are the primary output, driving the team's overall progress.

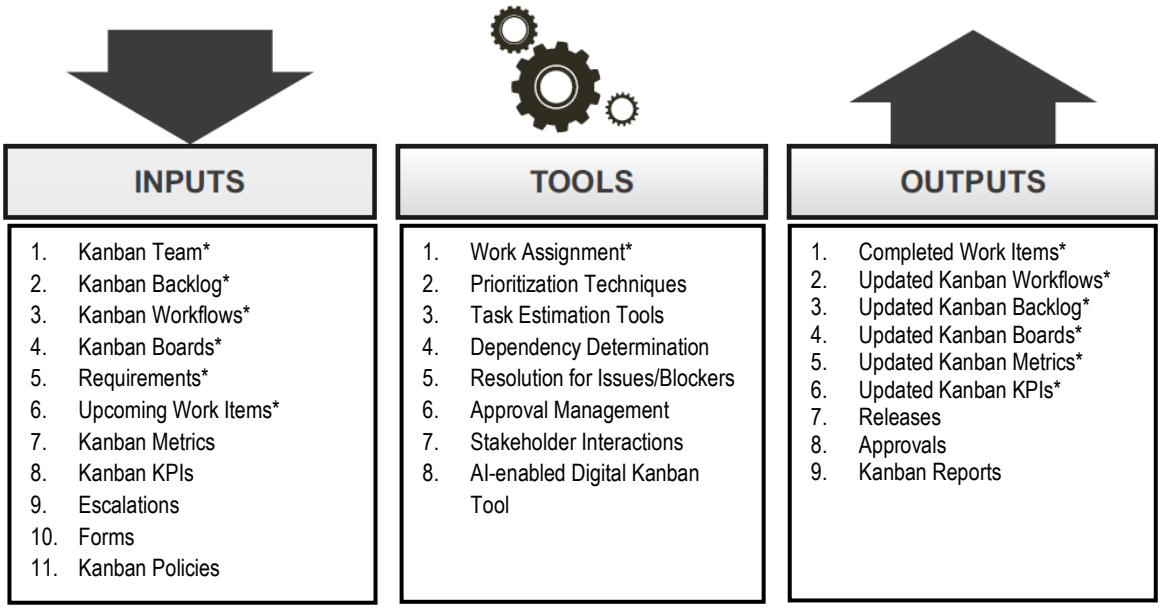

Figure 8-3: Optimize Workflows and Determine Stakeholders—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

8.1.1 Inputs

8.1.1.1 Kanban Team*

Kanban Team pulls prioritized work, collaborates continuously, manages flow, limits work in progress, and delivers value iteratively, ensuring efficient execution within the 'Get Work Done' process.

For more information, see section 3.1.

8.1.1.2 Kanban Backlog*

Kanban backlog provides a prioritized, visible queue of Work Items, enabling the team to pull Tasks, manage flow, and deliver value efficiently in the process.

For more information, see section 3.5.2.

8.1.1.3 Kanban Workflows*

Kanban Workflows defines stages, policies, and transitions for Work Items, guiding the team in managing flow, ensuring clarity, and promoting efficiency in getting work done.

For more information, see section 3.5.1.

8.1.1.4 Kanban Boards*

The Kanban Board visualizes work, stages, and flow, enabling the team to track progress, identify bottlenecks, and manage Tasks effectively, ensuring transparency and alignment in the 'Get Work Done' process.

For more information, see section 3.5.1.

8.1.1.5 Requirements*

Requirements, whether they come from customers, internal stakeholders, regulatory bodies, or other sources, serve as the foundation for creating Work Items. These requirements are analyzed and translated into specific Task Groups, Tasks or User Stories that can be added to the Kanban Backlog or Kanban Board. A clear understanding of requirements is crucial for ensuring that the team is working on the right things and delivering value to the customer.

8.1.1.6 Upcoming Work Items*

Upcoming Work Items in Kanban act as prioritized inputs to the Get Work Done process, enabling smooth Task selection and continuous Workflows based on team capacity. They are derived from Requirements, as mentioned in Section 8.1.1.5.

Managing Upcoming Work Items in Kanban is described in Section 3.5.4.

8.1.1.7 Kanban Metrics

Kanban Metrics, like cycle time, lead time, and throughput, provide real-time insights into Workflow efficiency. By analyzing these metrics, teams can identify bottlenecks, optimize processes, manage WIP limits, and ensure smoother, faster delivery of work items in Kanban.

For more information, see section 4.2.

8.1.1.8 Kanban KPIs

Kanban KPIs, such as delivery predictability, customer satisfaction, and time to market, guide prioritization and focus, ensuring that the team aligns work to business goals, improving value delivery and decision-making.

For more information, see section 7.2.5.7.

8.1.1.9 Escalations

Escalations, or issues that require immediate attention, can trigger the creation of new Work Items. When an escalation occurs, a new Task or bug report may be created to address the issue. Escalations help prioritize work and ensure that critical issues are resolved promptly.

For more information, see section 5.4.

8.1.1.10 Forms

Forms provide structured templates for capturing essential work details, ensuring consistency, clarity, and completeness of information. They support efficient Task initiation, tracking, and collaboration throughout the 'Get Work Done' process in Kanban.

For more information, see section 5.3.

8.1.1.11 Kanban Policies

Kanban policies provide clear rules and expectations for how work is handled. They guide team behavior, manage flow, limit work-in-progress, and support collaboration, making them essential for effectively getting work done in Kanban.

For more information, see section 7.2.5.6.

8.1.2 Tools

8.1.2.1 Work Assignment*

Assigning Work Items to specific team members is a crucial step in the Get Work Done process. By considering factors such as skill sets, workload, and dependencies, teams can effectively distribute work and minimize bottlenecks. Proper assignment of Work Items also helps identify potential collaborations and dependencies between team members.

Kanban follows a pull-based system, meaning that work is not assigned to individuals by a manager or lead. Instead, team members pull work from a prioritized backlog or input queue when they have the capacity to do so. In Kanban, work assignment is self-directed, visual, and flow-driven, based on capacity and system constraints. This empowers teams, increases accountability, reduces overload, and leads to more predictable delivery.

Here is a detailed breakdown of how work assignment functions in Kanban:

1. Pull System, Not Push

- In traditional systems, work is "pushed" onto team members, often leading to overload and context switching.
- In Kanban, team members pull work themselves, ensuring they only take on Tasks when they have the capacity and focus.

2. Visualized Workflows on the Kanban Board

- All Work Items are represented as cards on a Kanban Board.
- The board is divided into columns representing stages of the Workflows (e.g., To Do, In Progress, Review, Done).
- This visualization helps team members see where work is, what's available, and what needs attention.

3. Prioritized Backlog/Input Queue

- The Product Owner or customer representative maintains and prioritizes the upcoming work as described in Section 3.5.4.
- Team members typically pull from the top of this queue, ensuring they are working on the most valuable items first.

4. WIP (Work-In-Progress) Limits

- Each stage in the Workflows has a WIP limit, restricting how many items can be in progress at once.
- These limits prevent bottlenecks and ensure focus.
- Team members can only pull new work if it doesn't exceed the WIP limit.

5. Self-Assignment & Collaboration

- Team members choose Work Items based on:
  - Skills and expertise
  - Availability

    - Task priority and urgency
- Collaboration is encouraged, especially on complex or blocked items.

6. Explicit Policies

- Teams define clear policies for when and how items can move between stages.
- This ensures consistency in how work is pulled and completed.

7. Continuous Flow & Improvement

- The goal is to maintain a steady flow of work, reducing idle time and delays.
- Regular stand-ups and retrospectives help adjust the process for better flow and team dynamics.

8.1.2.2 Prioritization Techniques

In Kanban, prioritization is a bit different from other project management methods like Scrum or Waterfall. Instead of relying on fixed-length iterations or detailed upfront planning, Kanban emphasizes continuous flow and visual management. Prioritization in Kanban is dynamic and fluid, allowing teams to respond to changing needs quickly.

Here is a detailed breakdown of prioritization techniques in Kanban:

1. Work Item Types and Classes of Service

Kanban encourages the use of Work Item Types and Classes of Service to manage and prioritize Tasks.

Work Item Types
These are categories of work such as:

- Features
- Bugs
- Technical debt
- Support tickets

Each type may have different priorities or handling procedures.

Classes of Service
Classes of Service define how work is treated. Common examples:

Class of Service	Description	Priority Implication
Expedite	Urgent, critical items that need immediate attention. Only 1 or 2 should exist at a time.	Top priority; jumps the queue
Fixed Date	Must be delivered by a specific date (e.g., compliance Tasks).	Scheduled accordingly to meet the date
Standard	Normal priority Tasks with no fixed date.	Handled in regular flow

Class of Service	Description	Priority Implication
Intangible	Important but not urgent (e.g., tech debt).	Often lower priority but should not be neglected

Table 8-1: Example of Classes of Service

2. Explicit Policies and Pull Criteria

Kanban systems rely on explicit policies for moving Tasks from one column to another. Prioritization occurs at the point of pull (when a team member pulls the next item from the backlog or "Ready" column).

Techniques:

- Define Pull Criteria: Rules that specify which item should be pulled next (e.g., "pull the highest-value item that fits the available skill set").
- Limit WIP (Work in Progress): By limiting WIP, you force prioritization decisions since not everything can be worked on at once.

3. Cost of Delay (CoD) and Weighted Shortest Job First (WSJF)

- **Cost of Delay (CoD)**

  CoD measures the impact of not delivering a feature over time. It's used to decide which Work Item to do first based on value lost if it's delayed.

  Formula:

  Cost of Delay = Business Value + Time Criticality + Risk Reduction/Opportunity Enablement

- **WSJF (Weighted Shortest Job First)**

  Popular in SAFe, WSJF builds on CoD by considering job size:

$$\text{WSJF} = \frac{\text{Cost of Delay}}{\text{Job Duration}}$$

  Items with the highest WSJF score should be prioritized. This helps optimize economic outcomes.

4. Replenishment Meetings / Commitment Point

In Kanban, replenishment meetings are used to decide what goes into the "Ready" column. This is where most prioritization discussions occur.

How it works:

- Team meets regularly (e.g., once per week or as needed).
- Stakeholders and team members review backlog items.
- Items are prioritized based on business value, urgency, risk, etc.
- Selected items are moved into "Ready" for the team to pull when they have capacity.

5. Visual Prioritization in the Backlog

The backlog in Kanban is often visually ordered, with the top item being the highest priority. Some other prioritization techniques used:

- Color codes (e.g., red for urgent)
- Tags or swimlanes to separate work types
- Priority labels (High, Medium, Low)

Visual cues make prioritization transparent and easy to update.

6. Cumulative Flow Diagrams (CFD) and Lead Time Analysis

Kanban encourages using metrics to inform prioritization. For example:

- A growing queue of bugs may signal the need to prioritize defect resolution.
- Long lead times in a certain work type may indicate a bottleneck that needs attention.

7. Stakeholder Input and Feedback Loops

In Kanban, stakeholder engagement is ongoing. Frequent feedback and collaboration help keep priorities aligned with real-world needs. Some tools used are:

- Customer interviews
- Data from deployed features
- Service Level Expectations (SLEs) to ensure timely delivery

Summary

Technique	Use Case
Classes of Service	Categorize and prioritize different types of work
WSJF / Cost of Delay	Quantify value and urgency
Pull Criteria	Set rules for choosing the next item
Replenishment Meeting	Collaborative prioritization
Visual Indicators	Transparency and flexibility
Metrics (Lead time, CFD)	Data-driven prioritization
Stakeholder Feedback	Keep priorities relevant

Table 8-2: Summary of Prioritization Techniques

8.1.2.3 Task Estimation Tools

In Kanban, Task estimation is typically approached differently from traditional project management or even other Agile methods like Scrum. Instead of emphasizing time-based estimations (e.g., story points or hours), Kanban focuses on flow, cycle time, and throughput to manage and forecast work.

However, there are still tools and techniques within the Kanban methodology that help teams estimate, prioritize, and deliver work effectively.

Description of Task estimation tools and methods used in Kanban:

1. Cycle Time Tracking

- Definition: Cycle time is the amount of time it takes for a Task to move from the start of the Workflows to completion.
- Tool: Cycle Time Charts
- Usage:
    - Visualize how long Tasks typically take.
    - Identify bottlenecks or variability in the Workflows.
- Estimation Role: Historical cycle time data is used to predict how long new Tasks might take, without needing to estimate each individually.

2. Cumulative Flow Diagram (CFD)

- Definition: A chart that shows the number of Tasks in each state (To Do, In Progress, Done) over time.
- Tool: CFD Graphs (available in tools like Jira, Azure DevOps, Trello with Power-ups)
- Usage:
    - Monitor work in progress (WIP).
    - Predict future delivery based on current flow.
- Estimation Role: Helps in forecasting completion of features or releases by analyzing the rate of work done.

3. Work Item Types and Class of Service

- Definition: Tasks are categorized by their type (e.g., feature, bug, maintenance) and by urgency (standard, expedite, fixed date, intangible).
- Tool: Swimlanes or color-coding on Kanban Boards
- Usage:
    - Classify and prioritize work visually.
    - Understand historical flow per category.
- Estimation Role: Historical performance per class/type allows estimation based on category, not individual Tasks.

4. Monte Carlo Simulation

- Definition: A statistical technique used to simulate many possible outcomes and forecast delivery dates.
- Tool: Specialized software/plugins
- Usage:

    - Run simulations using historical data.
    - Predict with a confidence interval when a batch of Tasks will be completed.
- Estimation Role: Provides probabilistic forecasting rather than deterministic estimates.

5. Lead Time vs. Cycle Time Analysis

- Definition:
- Lead Time: Time from Task creation to completion.
- Cycle Time: Time from when work begins to when it's finished.
- Tool: Lead/Cycle Time Reports
- Usage:
    - Compare how long Tasks wait before starting vs. time taken to complete once started.
    - Identify delays and optimize flow.
- Estimation Role: More accurate estimations based on when work is likely to start and finish.

6. Service Level Expectations (SLEs)

- Definition: An agreement on how long a Work Item should take to complete.
- Tool: Kanban policies (often shown visually on boards)
- Usage:
    - Define expected cycle time for different types of work.
    - Track how often the team meets expectations.
- Estimation Role: Tasks are planned with the SLE in mind, creating reliable forecasts for stakeholders.

7. Little's Law

- Definition: A formula used in queuing theory:

$$\text{Average Lead Time} = \frac{\text{Work in Progress (WIP)}}{\text{Throughput}}$$

- Tool: Kanban Metrics dashboard (many tools calculate it)
- Usage:
    - Estimate average lead time based on real-time WIP and throughput.
- Estimation Role: Enables high-level forecasting without Task-level estimation.

8. Planning Poker

A collaborative tool used to estimate the effort required to complete a Task. It is a consensus-based estimation technique used by teams to estimate the effort, complexity, or time required to complete a Task. It involves team members assigning estimates using a set of cards with values representing effort (e.g., Fibonacci sequence: 1, 2, 3, 5, 8, 13, etc.). This technique fosters team collaboration and helps avoid bias in estimates. While estimation is less emphasized in traditional Kanban, Planning Poker can still be valuable in Kanban initiatives for forecasting work completion and improving flow efficiency.

Summary: Kanban's Estimation Philosophy

Traditional Agile	Kanban Approach
Estimates in story points or hours	Uses flow-based metrics
Focus on individual Task estimation	Focus on system performance and throughput
Commit to sprint workload	Pull-based, continuous delivery
Planning in fixed iterations	Forecast using historical data trends

Table 8-3: Summary of Kanban's Estimation Philosophy

8.1.2.4 Dependency Determination

In Kanban, dependencies are constraints that prevent a Work Item from progressing independently through the Workflows. They often create delays, blockages, or increased lead time and need to be handled deliberately to maintain the system's flow efficiency.

In Kanban, teams should not avoid dependencies — they should visualize and manage them as part of the flow. Rather than hiding dependencies within sprints or backlogs, teams should expose them early, map them clearly, and use data to reduce their impact.

Goals of Dependency Determination in Kanban

- Improve flow: Avoid bottlenecks caused by waiting on other Tasks or teams.
- Increase visibility: Clearly see which Tasks rely on others.
- Enable better planning: Understand sequencing needs and resource requirements.
- Reduce risk: Identify points of potential delay before they happen.

Types of Dependencies

- Task-to-Task: Task B can't start until Task A finishes.
- Team-to-team: One team's work depends on another team's deliverable.
- Resource-based: A shared resource or person is needed for multiple Tasks.
- External: Waiting on third parties (e.g., vendors, regulatory bodies).
- Temporal: A Task must happen at or after a specific time/date.

How Are Dependencies Determined in Kanban?

1. Visual Workflows Analysis

- Tool: Kanban Board (digital or physical)
- Method:
    - Review blocked items, slow-moving Tasks, or work piling in certain columns.
    - Look for patterns in flow disruption.
- Clue: Cards in "In Progress" or "Waiting" columns for a long time often signal a hidden dependency.

2. Explicit Relationship Mapping Between Cards

- Tools:
  - Vabro (linked Task Groups, Tasks)
  - Jira (linked issues: “blocks”, “is blocked by”)
  - Kanbanize (“Related to”, “Precedes”, “Follows”)
  - Azure DevOps, Trello (with Power-Ups)
- Method: Use card linking to define:
  - “Depends on” (card B depends on card A)
  - “Blocks” (card A blocks card B)
- This allows real-time tracking of which items can’t proceed without others.

3. Tagging & Labeling for Dependency Identification

- Method:
  - Use labels, stickers, or tags (e.g., “External Dependency”, “Waiting on Backend”)
  - Assign colors to different dependency types
- Benefit: Enables quick filtering and tracking of all dependent items.

4. Use of Swimlanes or Grouping

- Visual Aid: Horizontal lanes can represent:
  - Functional areas (e.g., design, development)
  - Work packages (e.g., features, epics)
- Team ownership Insight: If a Task must cross lanes, it may have inter-team dependencies.

5. Workflows Policies to Expose Dependencies

- Kanban Policy: Define when a Task may move forward (e.g., “All upstream Tasks must be complete”).
- Checklist or Definition of Ready (DoR):
  - Use checklists to ensure that prerequisites are met.
    Example: “UI Mockups Approved” may be a precondition before development starts.

Indicators of Undiscovered Dependencies

- Repeated context switching or Task rework
- Tasks stuck in blocked/waiting states
- Missed deadlines or unstable flow
- Frequent handoffs between teams

Risks of Poor Dependency Management in Kanban

Risk	Consequence
Delayed Work	Longer lead times, missed delivery goals
Bottlenecks	Flow halts, Task queues build up
Poor Predictability	Forecasting becomes unreliable

Risk	Consequence
Decreased Throughput	Work Items pile up in queues

Table 8-4: Risks of Poor Dependency Management in Kanban

Tools and Techniques to Manage Dependencies

- Advanced Approaches
  - Monte Carlo Simulation with Dependencies
  - Predicts how dependencies affect flow and delivery date probabilities.
- Dependency Matrices
  - Grid-based mapping of interrelated Tasks or teams to spot and resolve conflict areas.

Best Practices

- Make dependencies visible – hide nothing; surface them on the board.
- Prioritize breaking dependencies – reduce cross-team or handoff delays.
- Apply WIP limits – encourages faster handling of blockers and dependencies.
- Foster cross-functional teams – reduce the need for handoffs.
- Track blockers and delays – treat them as metrics to improve the system.

8.1.2.5 Resolution for Issues and Blockers

In Kanban initiatives, resolving issues and unblocking stalled work is critical to maintaining a smooth Workflows and ensuring steady progress toward goals. It is essential to enable teams to quickly identify, assess, and remove impediments that disrupt the flow of work. Steps in issue and blocker resolution can include:

- Issue or blocker identification
- Issue categorization and assessment
- Resolution planning, prioritization, and escalation
- Resolution implementation and monitoring
- Feedback and improvement

By addressing issues and blockers effectively, Kanban initiatives can enhance their ability to overcome obstacles, maintain Workflows momentum, and deliver value consistently. This approach not only resolves immediate challenges but also empowers teams to identify and mitigate long-term inefficiencies, driving continuous improvement.

8.1.2.6 Approval Management

Kanban is a visual Workflow management method that emphasizes continuous delivery, limiting work in progress (WIP), and optimizing flow. While it is often associated with flexibility and simplicity, Kanban is highly effective in managing structured processes, including approval Workflows. Approval management within Kanban ensures that specific work items meet necessary review, validation, or compliance requirements before progressing to the next stage of the process or being considered complete.

Approval management in Kanban is a robust and adaptable process that enhances quality control, accountability, and compliance without disrupting flow. By visualizing approval stages, defining explicit policies, leveraging modern tooling, and integrating automation, Kanban teams can effectively manage approvals while maintaining agility and efficiency.

1. Visualizing Approval Stages on the Kanban Board

- In Kanban, all Workflow steps are represented as columns on a board. To incorporate approval management, dedicated columns can be introduced to reflect approval stages clearly.
  For example:

  To Do → In Progress → Review → Awaiting Approval → Approved → Done

  In this model:

  - "Awaiting Approval" indicates that the work item has been completed and is pending approval.
  - "Approved" signifies that approval has been granted and the item is ready for final delivery or deployment.

- This visual approach ensures transparency in the approval process, allowing team members and stakeholders to easily identify work items awaiting review or authorization.

2. Utilizing Explicit Process Policies

- A core principle of Kanban is the use of explicit policies—clearly defined rules that govern how work is performed and how transitions occur between Workflow stages. For approval management, explicit policies might include:
- Who is authorized to provide approval (e.g., a product owner, QA lead, or compliance officer).
- Criteria for approval, such as completed documentation, test results, stakeholder feedback, or regulatory sign-offs.
- Procedures to follow if approval is denied, such as returning the item to a previous stage with specified revisions.
- By establishing and communicating these policies, teams reduce ambiguity and ensure consistent handling of approvals.

3. Leveraging Labels, Tags, and Custom Fields

- In scenarios where board space is limited or the approval process is not linear, approval states can be tracked using:
  - Having “Approvals” as a specific Task Property while defining Tasks – described in Section 3.5.1.
  - Specific Labels or tags, such as Pending Approval, Approved, or Rejected.
  - Custom fields that reflect the current approval status or the name of the approver.
  - Card checklists to outline specific approval steps and track their completion.
- These methods provide flexibility while preserving clarity and traceability.

4. Work-In-Progress (WIP) Limits and Flow Efficiency

- Approval stages should be included within the Kanban system's WIP limits. This design choice encourages teams to monitor and address bottlenecks that may arise from delayed approvals. By limiting the number of work items in the approval stage, teams are incentivized to resolve pending approvals promptly, thereby maintaining flow efficiency and reducing lead times.

5. Automation and Notifications

- Modern Kanban tools (e.g., Vabro, Jira, Trello, Azure DevOps) support automation features that streamline approval processes. Examples include:
- Automatic notifications to alert approvers when an item enters an approval stage.
- Trigger-based transitions, where a card is automatically moved to the next stage upon receiving approval via a comment, checklist item, or status update.
- Integration with external systems, such as document management or CI/CD tools, to facilitate compliance and release approvals.
- Automation reduces manual overhead, shortens cycle times, and minimizes the risk of oversight.

6. Audit Trails and Compliance

- In many industries, maintaining a verifiable record of approvals is essential for compliance and governance. Kanban systems support this need through:
    - Activity logs that track who moved a card, when it was moved, and any associated comments or file attachments.
    - Card histories that provide a chronological view of each item's lifecycle, including approvals and rejections.
    - Version control integration to associate work items with specific changes or releases.
- This ensures full traceability and satisfies internal or external audit requirements.

8.1.2.7 Stakeholder Interactions

Stakeholder interactions become more transparent, timely, and data-informed—driving better alignment, faster feedback loops, and stronger collaboration across all levels.

For more information about Stakeholders, see section 3.3.

8.1.2.8 AI-enabled Digital Kanban Tool

In the context of Kanban initiatives, AI-enabled Kanban tools can provide recommendations that enhance Task management by streamlining Workflows, reducing decision-making time, and improving prioritization. By leveraging AI, teams can identify and organize Task Groups and Tasks that align with organizational goals, resource availability, and workload distribution. AI analyzes historical data, ongoing work, team performance metrics, and Workflows objectives to suggest Task Groupings and specific Tasks based on the following elements:

- Work Item Characteristics: Task dependencies, estimated effort, and Task type (e.g., development, testing, review).
- Historical Trends: Past throughput, lead times, and success rates for similar Tasks.

- Resource Availability: Current team capacity and skillset alignment.
- Priority and Impact: Business goals, deadlines, and the potential value of Tasks.

AI-enabled digital Kanban tools can also suggest Tasks for inclusion in specific Kanban Boards, addressing the needs of:

- Backlog Refinement
- Prioritization of Work
- Work In Progress (WIP) Management
- Risk Mitigation
- Cycle Time Reduction
- Innovation or Stretch Goals
- Resource Utilization
- Continuous Improvement

By incorporating the Task Groups and Tasks into Kanban initiatives, teams can optimize Workflows, enhance productivity, and better align efforts with organizational objectives.

Figure 8-4 shows Notion's Task management interface. It displays Task details, comments, and AI-assisted editing suggestions such as "Change tone," "Fix typos," "Shorten," "Summarize," and "Translate."

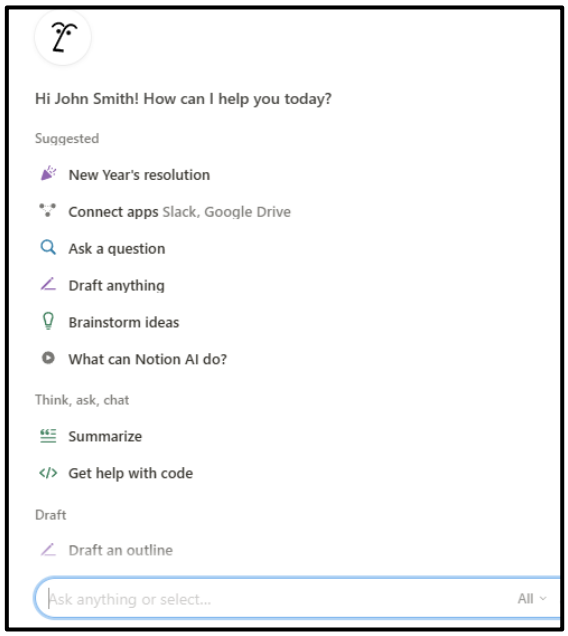

Figure 8-4: Use of AI in Kanban (Source: Notion)

Figure 8-5 shows a digital workspace for Task management in Wrike, featuring a Task titled "Communicate the details of the event," assigned to John Smith with a deadline of January 31. It includes AI-assisted options to modify the Task, such as "Change tone," "Fix typos," "Shorten," "Summarize," and "Translate."

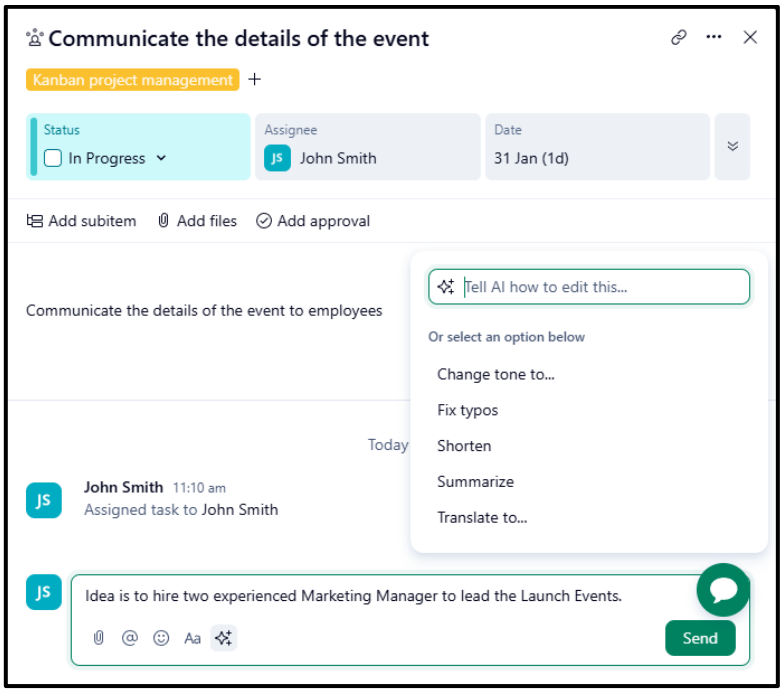

Figure 8-5: Use of AI in Kanban (Source: Wrike)

Figure 8-6 depicts a product management roadmap interface in Airtable, showcasing a feature titled "More control over notifications" in the backlog stage, with an option to create an AI-assisted field for further analysis. Users can interact with the feature details, provide feedback, and manage its progress within the product roadmap.

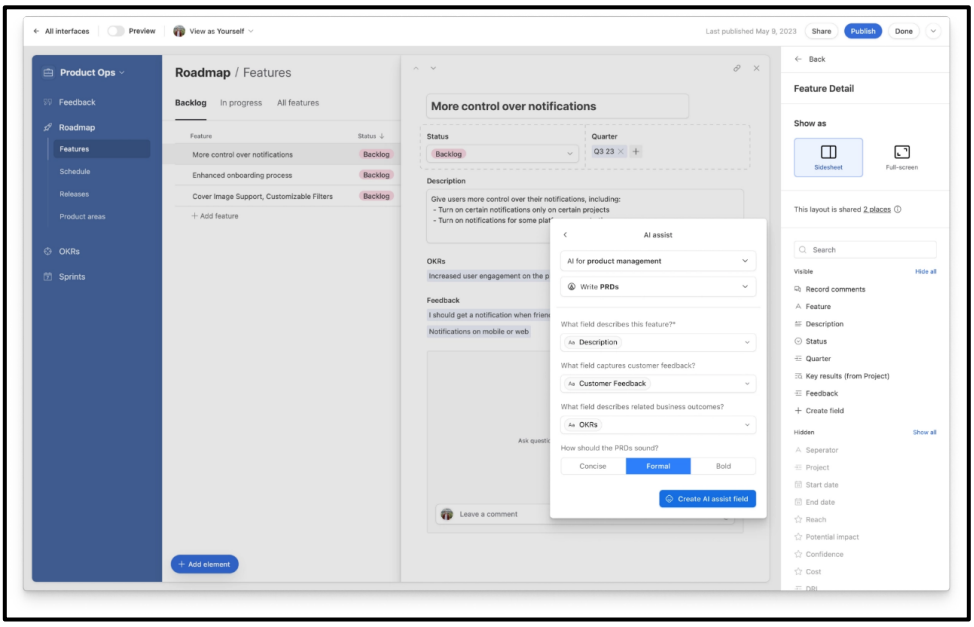

Figure 8-6: Use of AI in Kanban (Source: Airtable)

Figure 8-7 displays a Workflows management interface in Nifty, focusing on Task scheduling and recurrence options for a "Weekly Blog" article within the "Content Marketing" dashboard. It shows the recurrence pattern setup, highlighting an example of "Every first day of the month," along with options to customize the schedule, including start/end dates, reminders, and specific days of the week.

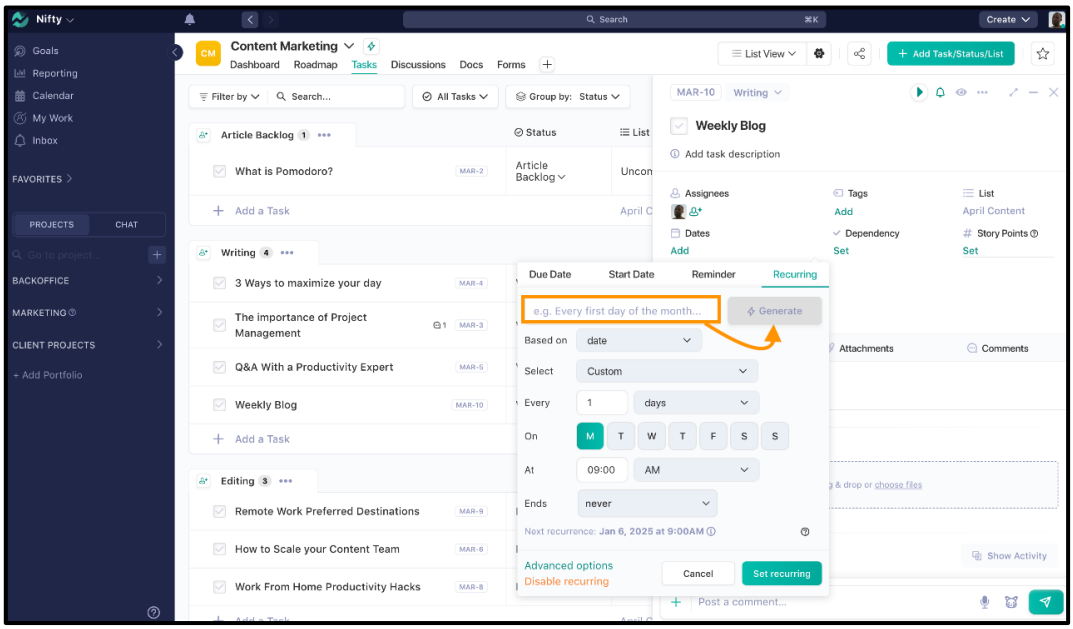

Figure 8-7: Use of AI in Kanban (Source: Nifty)

8.1.3 Outputs

8.1.3.1 Completed Work Items*

Completed Work Items in Kanban are Tasks or Task Groups that:

- Have moved through all the stages of the Kanban Workflows.
- Are no longer being worked on or reviewed.
- Are ready to be delivered, deployed, or archived.
- If defined, have satisfied the team's Definition of Done (DoD). The Definition of Done (DoD) for Work Items in Kanban defines when Tasks or Task Groups are considered complete, meeting all criteria such as functionality, quality, testing, and approval before delivery.

These items are usually placed in the final column of the board — often labeled "Done", "Completed", or "Ready for Delivery".

Characteristics of Completed Work Items

- Finalized: No further work, changes, or review are needed.
- Validated: Quality checks, testing, or approvals are complete.
- Tracked: They are counted in metrics like throughput and lead time.
- Archived or Delivered: Depending on the process, they may be handed off to a customer or stored for record-keeping.
- Approved or Rejected: If necessary, some Completed Work Items may require approval and must go through the approval management process outlined in Section 6.1.2.6. Based on the decision of the approvers, certain work items may be marked as "Approved," while others may be marked as "Rejected." If needed, some rejected work items may require rework before they can be approved.

Why Track Completed Work Items?

Tracking Completed Work Items allows teams to:

- Measure performance (e.g., using throughput or cycle time).
- Improve predictability of delivery.
- Identify bottlenecks by analyzing which items take too long.
- Celebrate progress and maintain morale.

Important Related Kanban Metrics

Metric	Description
Throughput	Number of Completed Work Items over a set period of time
Lead Time	Time from request to completion of a work item
Cycle Time	Time from when work starts to when it finishes

Metric	Description
Cumulative Flow Diagram (CFD)	Shows the status of all work items, including completed ones

Table 8-5: Related Kanban Metrics

Example Workflows Showing Completed Work Items

Here's a simplified Kanban Board with a completed item flow:

[Backlog] → [To Do] → [In Progress] → [Testing] → [Done]

A card representing a Task (e.g., "Develop Login Feature") starts in Backlog, gets pulled through each stage, and when it reaches Done, it's considered a Completed Work Item.

Tips for Managing Completed Work Items in Kanban

- Keep the Done column clean: Periodically archive completed items to avoid clutter.
- Use Automation: In digital tools like Jira, Trello, or Azure DevOps, automate movement to Done when checklists or Workflows are finished.
- Analyze Patterns: Review completed items during retrospectives to understand trends and blockers.
- Maintain DoD (optional): Will ensure that only work that truly meets the “Definition of Done” goes into the Done column.

8.1.3.2 Updated Kanban Workflows*

The updated Kanban Workflows, as an output of "Get Work Done," delivers improved visibility, streamlined Task flow, intelligent dependency management, and data-driven insights for higher team efficiency and agility.

For more information, see sections 3.4.2 and 7.2.1.2.

8.1.3.3 Updated Kanban Backlog*

The updated Kanban backlog reflects prioritized, actionable Tasks with improved clarity, reduced waste, and better alignment to team capacity and Workflows efficiency.

For more information, see section 3.4.3.

8.1.3.4 Updated Kanban Board*

The updated Kanban Board, as an output of "Get Work Done," showcases optimized Workflows, clear Task ownership, real-time status tracking, and enhanced visibility across teams and priorities.

For more information, see section 3.4.4.

8.1.3.5 Updated Kanban Metrics*

Kanban Metrics keep getting updated as work is done in the Kanban process, and these metrics help teams measure efficiency, identify bottlenecks, and continuously improve Workflow performance and delivery speed. Use of an AI-powered Digital Kanban Tool will greatly help in automatically updating and reviewing all important Kanban Metrics.

Kanban Metrics are discussed in section 4.1.

8.1.3.6 Updated Kanban KPIs*

After the "Get Work Done" process in Kanban, KPIs may be updated, reflecting how effectively the team met goals, delivered value, and optimized Workflow efficiency.

Kanban KPIs are discussed in section 7.2.3.7.

8.1.3.7 Releases

While Kanban does not have built-in release cycles like Scrum (e.g., sprints), it can still fully support the concept of releases—often more flexibly, if required.

Here is how Releases work in Kanban:

1. **Continuous Flow Instead of Time-Boxed Cycles**
   - In Kanban, work moves through the system continuously—there are no fixed sprints or release windows.
   - Once a work item reaches the final column (e.g., "Ready for Release" or "Done"), it's potentially shippable.
   - This makes Kanban ideal for continuous delivery or frequent releases.

2. **Visualizing Release States on the Board**
   - Kanban Teams can add release-specific columns or swimlanes to your Kanban board:
   - For example: To Do → In Progress → QA → Ready for Release → Released
   - Kanban Teams could also use labels, tags, or card types (e.g., "Release 1.2.0") to track which cards belong to which release.

3. **Release Policies**
   - Kanban encourages explicit policies to manage work. Kanban Teams can define policies like:
     - "Only release items from the 'Ready for Release' column after successful regression testing."
     - "Bundle all 'Ready' items for release every Thursday at 4 PM."
     - "Release any high-priority bug fix immediately after QA passes it."
   - Policies help the team know when and how things are released.

4. **Triggering Releases**
   - Kanban Teams can release in Kanban in different ways, depending on needs:
     - Continuous deployment: Every card that hits "Done" gets deployed right away.
     - Batch releases: Group several completed cards together and release them all at once (e.g., weekly).

- Manual or automated: Use a script, button, or DevOps pipeline to push code live once it's ready.

5. Tracking What's in a Release

- Kanban Teams can manage release content using:
  - Tags or versions on individual cards (e.g., "Release 3.5").
  - A release checklist card that links to all cards going into that release.
  - A dashboard or report showing completed items by release tag.
- This helps with transparency and communication with stakeholders.

6. Integration with DevOps Tools

- Modern Kanban tools (like Vabro DevOps, Jira, Trello + Power-Ups, Azure DevOps, or Kanbanize) often integrate with:
  - Git / GitHub / GitLab (track commits, branches, merges per card)
  - CI/CD pipelines (trigger automated builds, tests, and releases)
  - Test automation tools (link test results to cards)
- This means a card can move from "In Progress" to "Released" automatically as code progresses through the toolchain.

7. Post-Release Feedback Loop

- Kanban promotes continuous improvement, so after a release:
  - Review metrics (lead time, cycle time, deployment frequency).
  - Hold retros or feedback sessions (even though retros are not required in Kanban, they are useful).
  - Refine your policies or WIP limits based on what Kanban Teams have learned.

Summary: How Kanban Supports Releases

Feature	Description
Release Timing	Flexible—continuous or scheduled
Release Triggers	When work hits "Done" or based on policy
Board Visualization	Use columns/tags to show release status
Tooling	Integrates with CI/CD and DevOps pipelines
Release Policies	Explicit rules guide what/when/how to release
Metrics	Tracks throughput, lead time, and frequency
Post-Release Learning	Encourage feedback and improvement

Table 8-6: How Kanban Supports Releases

Figure 8-8 shows a software development Workflows management board, specifically the "Release Plan" view in ClickUp, with Tasks categorized by status (Overdue, Today, Done). It displays Task details such as title, assigned team, start and due dates, priority, and development type, providing an overview of the progress and schedule of ongoing software releases.

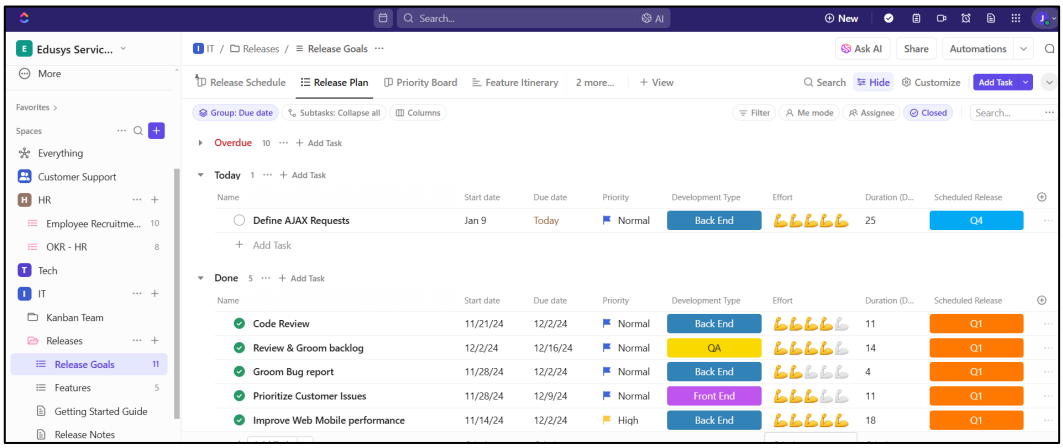

Figure 8-8: Sample Release Plan (Source: ClickUp)

Figure 8-9 shows the Vabro interface for a software release named "Sample Release," which is currently ongoing and scheduled for completion in 9 days, with a release date of January 31, 2025. It lists two Tasks in the "Release Backlog" section: "Sample Task - Setup New Employee Workstation" and "Sample Task - Customer Feedback on Resolved Issue."

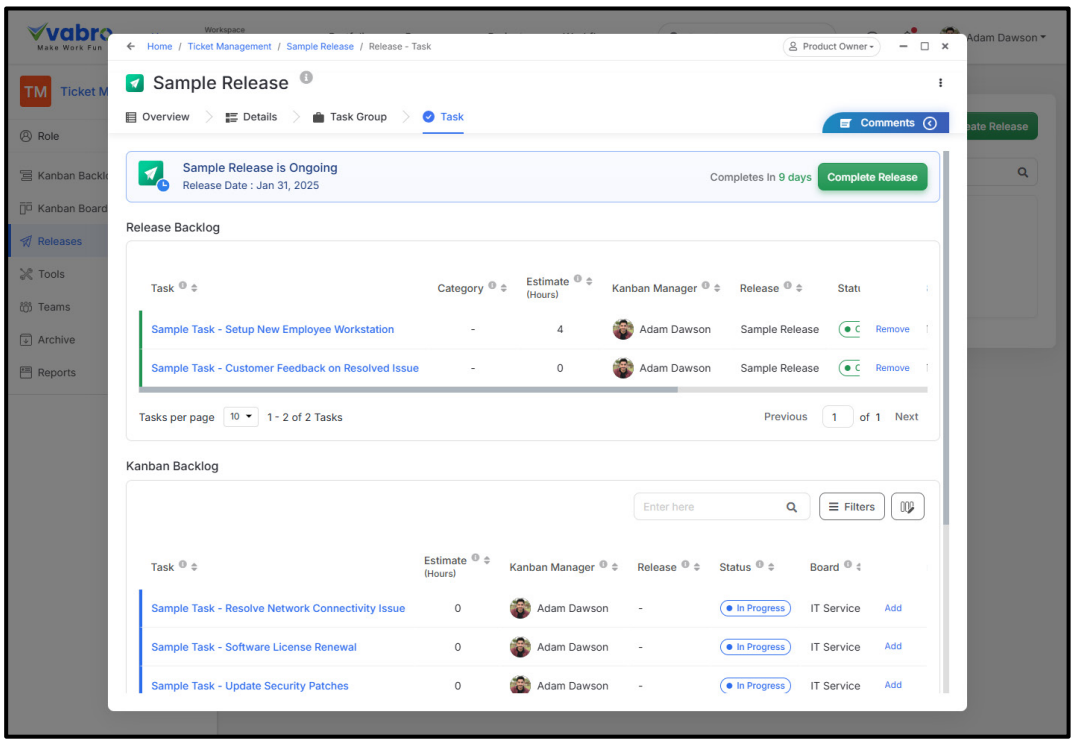

Figure 8-9: Sample Ongoing Release (Source: Vabro)

8.1.3.8 Approvals

Some Tasks or Task Groups may require approval before they are considered complete. In such cases, the assignee sends the Tasks or Task Groups to the Kanban Manager or Product Owner for approval, as required.

For more information about Tasks and Task Groups, please refer to Section 3.4.1.

8.1.3.9 Kanban Reports

Kanban reports provide insights into Workflow efficiency by tracking metrics such as cycle time, throughput, and WIP. These reports help teams identify bottlenecks and improve delivery performance. It is highly recommended to use an AI-powered digital Kanban tool, which can automatically update and review all key Kanban reports. These reports may be reviewed by Kanban Team Members, the Kanban Manager, the Kanban Product Owner, stakeholders, and senior management.

Kanban Reports are discussed in section 4.2.

The Practical Implementation Guide for Managing Workflows using Kanban

The Kanban Body of Knowledge (*KBOK™ Guide*) offers guidelines for successfully implementing Kanban, a widely used Agile methodology for managing business workflows. Originally developed in manufacturing, Kanban is now applied across various industries and sectors, including software development, healthcare, education, human resource management, retail, sales and marketing, finance, and more. It works for organizations of all sizes, from small businesses to large enterprises.

The *KBOK™ Guide* is built on insights from thousands of workflows across industries, with significant input from the global Kanban community and the VMEdu® Global Authorized Training Partner Network, comprising over 2,000 companies in more than 50 countries. Its development was a collaborative effort involving experts and practitioners from diverse fields.

The *KBOK™ Guide* is a comprehensive yet easily accessible framework for managing workflows with Kanban. It includes practical examples of Kanban implementation using popular IT tools, helping readers apply the methodology in their organizations. The guide also covers how Kanban integrates with other Agile frameworks such as Scrum, DevOps, OKRs, and Lean. Recommendations about how Artificial Intelligence can be used to increase productivity in Kanban workflows are also included in the *KBOK™ Guide*.

The *KBOK™ Guide* serves as a resource for both experienced Kanban practitioners and professionals new to workflow management. It's also suitable for those with no prior Kanban experience. The widespread adoption of the *KBOK™ Guide* framework standardizes how Kanban is applied to workflows globally and significantly helps organizations improve their overall productivity and return on investment.

Kanbanstudy
Targeting success

ISBN 979-8-9898916-1-0
9 798989 891610