NN/g

Lean UX & Agile
Glossary

Acceptance Criteria

Specific standards and functional requirements that a task or user story in the product backlog must meet before it is considered complete. Criteria are written in plain language from the user's perspective. They ensure that everyone involved in development, including UX, understands what to design or build and how it should work for users. Clear acceptance criteria prevent misunderstandings and delayed releases.

Agile

A method for developing software and user experiences that emphasizes collaboration and embracing change instead of following a linear, sequential plan, which is more common in Waterfall development. UX working directly with engineers and product managers, and a mindset focused on small slices of high-value research and design work can help make Agile more manageable.

Backlog *(also Product Backlog)*

A repository of high-level requirements written as epics, user stories, and tasks. The development team references, estimates, and prioritizes items in the product backlog as it completes work over time. The product backlog should include UX work, either as shared stories with engineering or as separate stories and tasks. Including UX work in the backlog helps prioritize it and enables the team to better anticipate collaboration points.

Backlog Refinement *(also Backlog Grooming)*

A meeting where the team discusses tasks and user stories in the product backlog to prioritize and prepare for the next sprint planning. It helps keep the development team, including UX, informed and the backlog organized. Refinement usually happens halfway through a sprint. UX should attend and actively participate.

Burn-down Chart

A line chart representing remaining work over time and capturing whether the team is on track for completion. UX can use burn-down charts to reprioritize, align, or adjust work with engineers accordingly.

Daily Standup *(also Daily Scrum)*

A team meeting that happens at the same time every day and that usually lasts 15 minutes or less. Team members share yesterday's progress, what they plan to do today to make progress, and any blockers inhibiting progress toward the sprint goal. UX should attend daily standups to stay aligned with the development team's progress, communicate any design or research-related blockers, and provide updates on user research and design decisions.

Development Team
(also Product Development Team)

The team that collaboratively delivers work on a product. It comprises 5–9 people with crossfunctional skills such as UX research and design, content strategy and UX writing, product management, engineering, and others. It's often best if UX roles are embedded on product-development teams rather than supporting multiple teams.

Epic

A large user story that the development team divides into smaller stories and tasks to complete over time and across multiple sprints. Epics help UX prioritize discovery and coordinate design work.

Estimation

The process of agreeing on a size measurement for user stories and tasks in a product backlog. Like engineers, UX should also estimate their work — accurate estimation helps with planning and forecasting research and design efforts. Estimation also allows UX to manage their workload and provide visibility to stakeholders. UX can use story points or t-shirt sizes (small, medium, large) to estimate work.

Kanban

A framework for developing products in which development-team members pull work from a Kanban board with columns representing work phases (e.g., to do, doing, done). The process, from the definition of a task to its delivery to the user, is displayed for all to see. Its visual and flexible nature makes it a suitable method for teams struggling to manage a UX research and design process as it eases prioritization and task tracking. Kanban also provides a transparent view of the status of UX work to the development team and stakeholders.

Lean UX

A collection of methods and frameworks emphasizing rapid prototyping and experimentation, collaboration, and iterative design. It aims to minimize waste and maximize learning by focusing on delivering value to users as quickly as possible. The Lean UX process involves testing hypotheses directly with users and refining designs based on early user feedback. By testing early and often, development teams create products that are more likely to succeed.

MVP *(Minimum Viable Product)*

A product or feature that works from start to finish but is limited in scope to only include what users need to complete a core task. For example, for a checkout flow, the MVP might include only guest checkout and credit-card payment; authenticated checkout and integration with third-party payment platforms can be deferred for later. Minimum viable products can help UX prioritize and design essential features and get early user feedback to reduce the risk of wasting time on the wrong things.

Product Manager *(also Product Owner)*

The role on the development team responsible for unifying members around a common vision and driving a product strategy to deliver value to the business and users. UX and product management are closely related roles and should partner in product development to understand and address user and business needs. Product managers are usually focused on viability and communicating with the team and stakeholders across the organization. UX practitioners working with both product owners and product managers in the Agile SAFe methodology should partner with both roles to ensure that product strategy and execution align.

Retrospective *(also Retro)*

A meeting that happens at the end of every sprint and where the team reflects on how things went and how the team members worked together. The team considers ways to improve process, based on lessons from recently completed work. Retro is a great time for UX to talk about process changes; because the topic of the meeting is to reflect on the team's processes, people are more receptive.

SAFe *(Scaled Agile Framework)*

A framework to help scale Agile development across many teams and projects. SAFe is typically used in large enterprise organizations where several groups contribute to a single product. There are specified planning and reflection points for all levels to ensure that leadership, management, and delivery teams align throughout projects.

Scrum

A framework that helps Agile teams structure and manage work using a set of meetings and artifacts. Like other Agile frameworks, Scrum emphasizes team collaboration, process flexibility, and continuous improvement through user and stakeholder feedback. In Scrum, UX is part of the crossfunctional development team working collaboratively with engineers, product owners, and stakeholders to integrate design and research into the development process. Work is done in a series of sprints, which allows for learning and adjustments to the product backlog along the way. Communication is frequent, which enables UX to include others in research and design.

Scrum Master

The role on the development team accountable for removing blockers and distractions so the team can deliver the sprint goal and associated work. Scrum masters are helpful partners for UX as they can clarify how the development process works and ensure that UX is involved. UX can also help scrum masters facilitate team discussions and influence process improvements.

Spike

Dedicated time during a sprint for user-focused, technical, business, and organizational research. It is typically added to the sprint backlog to account for unpredictable factors — issues that will require a yet unknown amount of time or effort to address. UX should use spikes to account for discovery work in Agile and to balance the time needed for discovery with other delivery work in the sprint.

Sprint

A fixed duration of time during which the development team, including UX, works on user stories and tasks comprising a product increment in the development process. Sprints usually last between 2-6 weeks.

Sprint Backlog

The prioritized scope of work that the development team anticipates completing by the end of a sprint. It should include UX research, content, and design work, with realistic effort estimates. Complete the most critical user stories and tasks first in each sprint.

Sprint Demo

A meeting in which the development team shares functioning code and interfaces with stakeholders and other teams. It usually occurs on the day before the end of the sprint and is an opportunity for UX to share how the work will affect the user experience. It's also a forum to reiterate the UX evidence that informed the current release and subsequent iterations.

Sprint Goal

A short, 1-2 sentence description of what the team plans to achieve in a sprint. It communicates the sprint's intended outcome rather than a list of features that the team plans to build. During sprint planning, UX can help define a user-centered sprint goal that reminds the team and stakeholders what the user will be able to do once the work in the sprint is done.

Sprint Planning

A meeting in which the development team reviews prioritized product backlog items, adds acceptance criteria, and estimates the level of effort for each item. It occurs on the first day of the sprint, resulting in a sprint backlog that the team will work on in the coming weeks. UX involvement in sprint planning is important because, like development work, UX work should also be prioritized and accounted for in the sprint backlog.

Sprint Review

A meeting where the team reviews what was completed in the previous sprint to prepare for sharing the work with others in sprint demo. It's important for UX to attend sprint review to confirm what was completed, plan for anything outstanding, and identify pertinent details to communicate at sprint demo.

Story Points

A unit of estimation measuring the complexity and size of the work to be done in a user story or task. If UX work is included in the team's velocity, UX should use the same method to estimate story points as the engineers in the team. Many teams use the Fibonacci sequence (0, 1, 2, 3, 5, 8, 13, 21) to assign points to stories, with more complex stories being assigned more points.

T-shirt Sizing

A simplified way to estimate the complexity and size of the work in a user story or task. T-shirt sizes are often used in addition to story points to avoid estimating different types of work with a single-story point value (e.g., usability testing and backend coding). UX often uses t-shirt sizes (extra small, small, medium, large, extra-large) alongside engineering story points to estimate research or design work — more complex stories get a larger t-shirt size.

Task

A user story can be broken down in to one or more specific tasks. When UX shares user stories with other roles such as product managers and quality assurance, research, content, and design work is often organized into subtasks of the parent user story.

User Story

A deliberately brief description of work or functionality from a user's point of view. It communicates what the user wants to do and the benefit they will receive. The typical format of a user story is a single sentence: "As a [type of user], I want to [goal], so that [benefit]" — for example, "As a checking account holder, I want to deposit checks with my mobile device so that I don't have to go to the bank." UX practitioners should represent their research, design, and content work in the backlog using user stories. We recommend that UX roles write these stories.

Velocity

The amount of work, expressed as story points, that the development team is comfortable committing to complete in a given sprint. Velocity is used to plan and forecast the future workloads of stable team members, including UX professionals, not just engineers.

UX Debt

Shortcut solutions that help the development team hit release dates, but, over time, leave mounting experience issues that adversely impact users and the organization. UX practitioners are well-suited to identify and track UX debt and to advocate and prioritize resolving it.

Waterfall

A traditional method of software development that follows a linear and sequential approach. Work happens in distinct phases; team members complete each phase before starting the next. Waterfall typically involves a separate design phase before development; in that phase, UX creates detailed designs that are then handed off to engineers to implement. Unlike with Agile, in the Waterfall framework there is little opportunity for feedback and iteration. Projects with well-defined requirements, clear scopes, and legitimate deadlines may be a good fit for Waterfall, whereas Agile is better suited for projects with changing needs and flexible requirements.

WIP *(Work In Progress)*

The number of stories, tasks, or items that have been started but not yet completed. Some teams will have a WIP limit, which limits the amount of work in progress. WIP limits help all roles improve focus and collaboration while reducing context shifting.