

Competencies Model for Non-Linear Training

Overview [↗](#)

The Competencies Model supports modular, non-linear neuro-literacy training by shifting the focus from linear content delivery to dynamic learner development. It enables personalized learning experiences across [all modes of interaction](#) with the system, including AI-powered micro-learning conversations, modular learning experiences, and formal programs.

Table of Contents

Overview
Competencies vs. Learning Goals
Learning Goals
Competencies
Key Features of the Competencies Model
Why Competencies Are Essential
Additional Considerations
Operationalizing the Competencies Model via xAPI

Competencies vs. Learning Goals [↗](#)

Learning Goals [↗](#)

Learning Goals define the content's *intended outcome* (e.g., knowledge or skills to be acquired). They are focused on what each module or session delivers (e.g., "Each learner understands how to map brain patterns").

- *Static, one-size-fits-all*: Each learner follows the same path, whether or not it meets their needs or matches their starting point.
- *Locks learners into linear progress*: Cannot adapt to individual pacing or prior knowledge.

Competencies [↗](#)

Competencies define what skills a learner has *demonstrated*, independent of specific training contexts. They reflect the learner's ability and progress across all interactions with the system, enabling personalization and non-linear delivery (e.g., "Can the learner map a brain pattern?").

- *Dynamic and adaptive*: Competencies follow the learner, allowing them to move dynamically between content.
- *Focuses on mastery*: Allows learners to skip steps they've mastered, accelerating their progress.

Key Features of the Competencies Model [↗](#)

Competency Model Design [↗](#)

- Competencies are structured hierarchically, with defined levels (e.g., novice to intermediate to advanced).
- Each competency level ties to specific skills and abilities, allowing learners to move through content modules via multiple pathways.
 - Early interactions help identify learner interests and competency baselines.

Granular and Adaptive Delivery [↗](#)

- **Assessment**: The system assesses the learner's competency in neuro-literacy skills through interactions.
- **Adaptive Learning**: Based on the learner's demonstrated competency, the system dynamically serves the most relevant content, conversation, or lesson to help them progress within their *zone of proximal development*.

Modes of Interaction

- The Competencies Model tracks learner performance across all interactions with the system, including:
 - **Conversational Micro-learning Experiences:** Short, interactive experiences to engage learners and assess competencies.
 - **Modular Learning Paths:** Rich multimedia lessons with opportunities for exercises and deeper engagement.
 - **Formal Structured Programs:** Interactive, AI-enhanced versions of existing guided programs, offering conversational debriefs, facilitated by Live Guides.

Personalization Over Time

- Initial interactions are generic due to limited learner data.
- As the system gathers insights from ongoing interactions, it tailors the learning path, offering progressively personalized experiences.

Iterative Learning Experience

- The model ensures flexibility by offering varied paths to achieve the same competencies.
- Learners progress through meaningful, relevant content aligned with their current abilities and goals.

This approach enables NEXT to create a dynamic, learner-centered training ecosystem, meeting users where they are and guiding them toward their goals in a personalized, engaging manner.

Why Competencies Are Essential

Competencies aren't just a better version of learning goals—they're a foundational shift that enables the personalization, scalability, and innovation that our V2 Platform promises. Without a Competencies Model, we're stuck with a rigid, linear system that can't adapt to diverse learners or AI-driven delivery. Competencies future-proof the V2 Platform, let us scale efficiently, and position NEXT as a leader in cutting-edge learning technology.

Additional Considerations

Personalization Isn't Optional in a Modular, AI-Driven World

- Learning goals are rigid, designed for linear, one-size-fits-all experiences.
- Our vision for V2 is *non-linear, modular, personalized*, and AI-supported.
- A competency model is the *only way* to track and adapt to individual learner needs at scale.

Competencies Align with NEXT's Mission and Scalability


- NEXT's mission is to empower learners to master their brain and behavior in *their own way*. Competencies allow us to:
 - Meet learners where they are (novices, intermediates, or advanced users).
 - Give users agency in their journey, reflecting our belief in self-directed neuroplasticity.
 - Scale effectively by letting AI handle personalization, reducing the dependency on human Guides.
- Learning goals limit us to small-scale, highly guided delivery, which isn't feasible for expansion or competitive positioning.

Competencies Future-Proof the Platform

- The market for AI-driven personalized learning is accelerating. Platforms like Duolingo, Khan Academy, and Coursera are moving beyond content delivery to *skill mastery and adaptive learning*. Without competencies, we risk falling behind.

- A competency model lets us:
 - Implement advanced AI-driven recommendations sooner.
 - Generate compelling learner insights (e.g., “90% of users gained competency in X within 2 weeks”).
 - Empower ecosystems for user-driven innovation, opening opportunities for third-party content and partnerships.
-

Operationalizing the Competencies Model via xAPI

 For more detail see the [Operationalizing the Competencies Model with xAPI](#) page.

The **Competencies Model** serves as the foundation for tracking and advancing learner progress across the neuro-literacy platform. It defines the key competencies learners need to develop (e.g., self-awareness, self-regulation, and self-management) and outlines how these skills are demonstrated, measured, and refined.

To bring the Competencies Model to life, the platform will leverage **xAPI (Experience API)** as its operational engine. xAPI allows us to capture, analyze, and respond to learner interactions in real time, ensuring a personalized and adaptive learning experience. Here’s how xAPI operationalizes the model.

Granular Tracking of Competency Development

Every learner action—whether engaging with a conversational micro-experience, completing a modular lesson, or participating in a guided support call—is logged as an xAPI statement. These statements document demonstrated competencies (e.g., “identified a counterproductive brain pattern”) and provide detailed insights into learner progress.

Adaptive Learning Pathways

By tracking interactions across all learning modalities, xAPI enables dynamic adaptation of learning paths. For example:

- If a learner struggles with emotional regulation, xAPI triggers targeted interventions like emotional state transitioning exercises or additional micro-experiences.
- If a learner demonstrates mastery, xAPI unlocks advanced modules or formal programs.

Centralized Data and Feedback Loop

xAPI consolidates all learner activity into a centralized **Learning Record Store (LRS)**. This data is used to:

- Monitor progress through the Competencies Model hierarchy.
- Identify patterns of success or challenge to refine both the learning content and AI-driven personalization.

Real-Time Competency Insights

The platform continuously analyzes xAPI data to provide actionable insights, ensuring learners receive immediate feedback, encouragement, or guidance to stay engaged and on track.

Iterative Model Refinement

Over time, xAPI data informs updates to the Competencies Model itself, ensuring it evolves based on real-world learner outcomes and remains aligned with the platform’s goals.

Making Competencies Actionable

By integrating xAPI as the backbone of the Competencies Model, the platform transforms abstract learning goals into measurable, actionable, and adaptive experiences, driving meaningful progress for every learner.