ispring

From Code TO CO-DESIGN

A Developer's Guide to Collaborative eLearning Design

Table of Contents

You're More Than a Coder

YOU'RE A DESIGN PARTNER

Have you ever received a storyboard with a vague instruction like, “Make this engaging!” and wondered what that actually means? Or perhaps you've been handed a set of instructions so rigid and detailed that there was no room for your technical creativity?

For too long, the eLearning world has run on a “handoff” model. An instructional designer creates a storyboard, throws it over a metaphorical wall, and you, the developer, are expected to execute it perfectly, often with little context.

This model is broken. The best, most effective, and most innovative eLearning isn't born from a handoff. It's forged in a partnership between instructional design and technical development.

This guide is designed to help you bridge that gap. It’s not about adding more work to your plate. It’s about transforming your work from a series of tasks into a strategic mission.

So, what's in it for you?

- **Drastically Reduce Revisions:**
  By understanding the intent behind a request, you’ll get it right the first time, saving you from frustrating and repetitive rework.
- **Increase Your Professional Value:**
  You'll evolve from a task-taker to a strategic problem-solver – the person that teams rely on for technical and creative leadership.
- **Boost Your Job Satisfaction:**
  You'll engage your creativity, provide meaningful input, and see your expertise directly impact the quality of the final product.
- **Build Faster and Smarter:**
  You’ll learn to anticipate problems and suggest elegant solutions before they can derail a project timeline.

This guide will give you the framework to architect
truly effective learning experiences.

CHAPTER 1

Thinking Like a Designer: The “Why” Behind the “What”

To become a design partner, you first need to understand the fundamental principles that guide an Instructional Designer's decisions. Think of it as getting access to the core logic behind the design document.

1.1 The North Star: Understanding Learning Objectives

Before a single slide is created, an ID establishes the ***Learning Objectives***. These are clear, measurable statements that define what the learner should be able to do after completing the course.

Example of a poor objective

“The learner will understand the new safety protocol.”

Vague

Example of a good objective

“After this module, the learner will be able to correctly identify the five most common workplace hazards and list the appropriate safety response for each.”

Specific, measurable, action-oriented

Why this matters to you?

Learning Objectives are your ultimate success metric. Every technical choice you make, every interaction, animation, and layout, should serve these objectives. If a feature doesn't help the learner achieve the objective, it's noise.

Developer Analogy

Think of Learning Objectives as the project's ***API documentation***.
They define the exact expected output for the end-user (the learner).
Your code is the implementation that guarantees that output.

1.2 The Learner's Brain is a CPU: Managing Cognitive Load

Cognitive Load is the amount of mental energy a person has to process new information. A learner's brain has limited processing power, just like a computer's CPU and RAM. Bad design overloads the system, causing confusion and inhibiting effective learning.

How your development work directly impacts cognitive load?

- **Clean, intuitive UI:**
  When a learner doesn't have to think about ***how*** to use the course, they can focus all their mental energy on the content itself.
- **Well-paced animations:**
  A subtle animation that draws attention to a key concept is helpful. A flashy, distracting animation overloads the senses.
- **Consistency:**
  Using consistent layouts and button styles means the learner only has to learn your system once.

You are the gatekeeper of the learner's attention. Your clean, thoughtful implementation is one of the most powerful tools for reducing cognitive load.

1.3 From Slides to Stories: The Learner's Journey

An eLearning course isn't just a collection of screens; it's a structured journey designed to take a learner from a state of not knowing to a state of knowing and doing. IDs use concepts like scaffolding – building up knowledge piece by piece, from simple to complex – to make this journey successful.

A course's structure often looks like this:

1. Introduce a concept
2. Show an example
3. Let the learner practice in a safe environment
4. Test their understanding with a quiz

When you see a quiz in a storyboard, don't just consider it a technical component. See it as a crucial checkpoint in the learner's journey, allowing them to confirm their understanding before they move on to the next, more complex topic.

The Learner's Journey

CHAPTER 2

Decoding the Storyboard: Reading Between the Lines

A storyboard is a blueprint, but sometimes the most important information isn't explicitly written down. Learning to read between the lines will empower you to make smarter decisions.

2.1 Function vs. Intent: The “Tabs Interaction” Example

Let’s look at a common scenario.

The Function described in the storyboard

“Create a 4-tab interaction using the provided text for each tab.”

The Intent of the designer

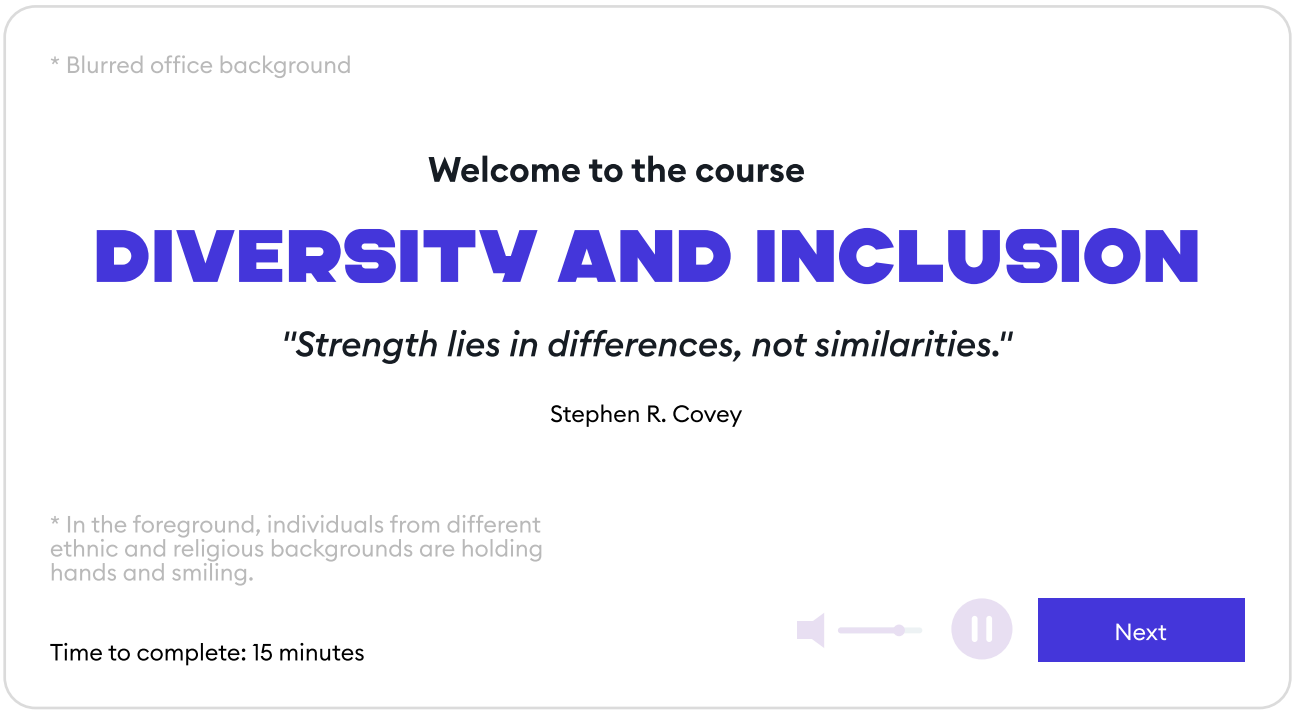
To chunk related information, prevent a wall of text, and reduce cognitive load for the learner.

A developer who only sees the function will build the tab interaction and move on. But a ***strategic develope***r understands the intent and might ask a powerful question:

“I see we're trying to chunk this information to make it digestible. That's a great idea. Since this is a linear process, would a simple click-and-reveal sequence be more intuitive for the learner than asking them to jump between tabs? It would guide them through the steps in order.”

This single question shows you're thinking about the learner experience and not just checking off a task. This is the essence of a design partner.

Storyboard Template Example



Course name: Diversity and Inclusion in the Workplace

SLIDE 1 – Intro slide. Welcome to the course!

Voiceover

As more companies focus on diversity and inclusion, we must ask ourselves, "How can we create a safe and inclusive work environment where diverse groups of people can work together effectively?"

This course will explore possible answers to this question and give you a better understanding of what diversity and inclusion are all about.

Besides the narration, there's a relaxed background sound – Slide1_sound.mp3

Programming notes

When a learner opens the course, background music will automatically start playing, and after three seconds, the narration will begin. When the narration has ended, the slide changes. However, the learner has the option to press the "Pause" button if they want to stay on that slide longer.

2.2 Spotting Gaps and Red Flags

Train yourself to scan a storyboard for common issues before you start building. This can save you hours or even days of rework.

- **Vague Instructions:**

  Phrases like "Make it pop," "Add some flair," or "Make it interactive" are red flags. They require clarification.

- **Asset Mismatches:**

  An image is provided, but it doesn't quite match the on-screen text or the cultural context of the learners.

- **Accessibility Gaps:**

  Key accessibility elements are missing, such as alternative text for images, a clear reading order for screen readers, or notes on keyboard navigation.

- **Conflicting Notes:**

  The ID's note in the storyboard conflicts with a client's comment in the margin.

Quick Tip

To think more like a designer, it helps to see the tools they use. A useful reference is the iSpring PowerPoint eLearning storyboard template. Examining its standard structure, with dedicated fields for on-screen text, audio script, and developer notes, gives you a clearer framework for interpreting the storyboards you receive. For more resources to decode design logic, you can explore An Instructional Designer's Notebook, a collection of design guides and checklists.

CHAPTER 3

The Art of Collaboration: Proposing, Not Opposing

The goal here is not to challenge the designer but to strengthen the design with your unique technical expertise. It's all in how you frame your feedback.

3.1 How to Give Feedback That Gets Heard

Your feedback is incredibly valuable, but it can be dismissed if it sounds like a complaint. A simple way to structure your comments is to use the ***P.A.C.T. Framework*** (see Chapter 4 for a full breakdown).

The Problem Statement

"This animation is impossible to build in the time we have."

(This sounds like a roadblock)

The Partnership Solution

"I see you're trying to illustrate this complex process, which is a great idea for the learners. Given our tight timeline, that specific animation style could be risky. What if we created a simple motion graphic video instead? I can use [Tool X] to get it done efficiently, and it will be just as clear for the learner."

(This presents a shared understanding, identifies a risk, and offers a concrete solution)

3.2 Become an Accessibility Champion

Accessibility (a11y) is not just a checkbox; it's a core component of good design. As the developer, you are the last line of defense and the ultimate advocate for an inclusive learning experience.

- Proactively build for keyboard navigation
- Test color contrast ratios
- Ensure that your code generates a logical reading order for screen readers

By embedding accessibility in your workflow, you don't just make the course compliant; you demonstrate senior-level ownership of the product's quality.

3.3 Suggesting Better Tools for the Job

The ID knows learning theory; you know technology. An ID might not be aware of a brand-new feature in the authoring tool or a powerful JavaScript library that could solve a problem more elegantly.

Example

"I know the storyboard calls for a complex, custom-built drag-and-drop quiz. Just so you know, the latest update to our authoring tool has a new matching feature that could meet this need perfectly and save us about 10 hours of development time. Would you be open to looking at it with me?"

CHAPTER 4

The Strategic Developer's Toolkit

This chapter contains “grab-and-go” tools you can use right away to put the principles in this guide into action.

TOOL 1

The “Storyboard Sanity Check” Flowchart

Use this mental process the moment you receive a new storyboard. It helps you spot issues ***before*** you start building.

START: New Storyboard Received

Step 1. Check the North Star

QUESTION

"Do I clearly understand the learning objective(s) for this module?"

IF NO

PAUSE. Ask the ID for clarification

IF YES

Proceed to Step 2

Step 2. Conduct an Asset Audit

QUESTION

"Are all assets (images, videos, etc.) provided and clearly labeled?"

IF NO

PAUSE. Send a list of missing assets to the ID

IF YES

Proceed to Step 3

Step 3. Hunt for Red Flags

QUESTION

"Do I see any potential technical, accessibility, or UX issues?"

IF NO

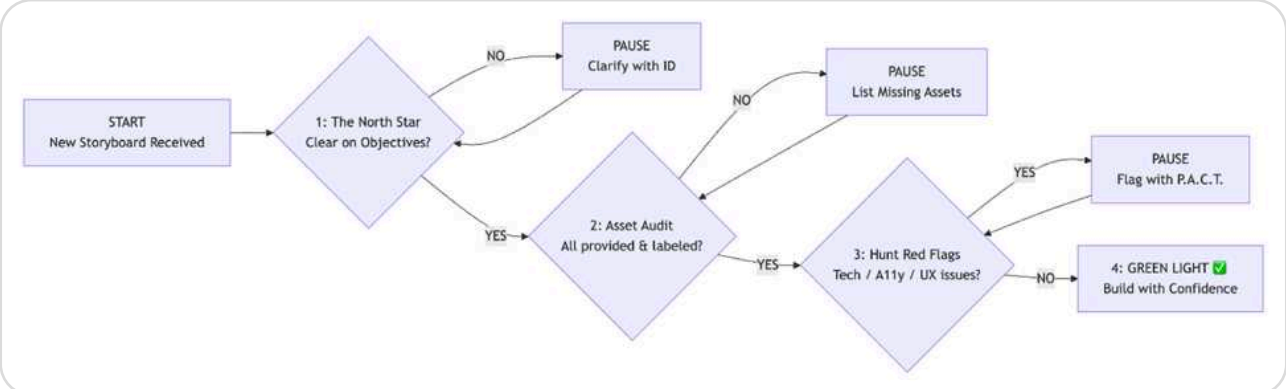
Proceed to Step 4

IF YES

PAUSE. Flag these items and use the P.A.C.T. framework to discuss them

Step 4. Green Light for Development

You are now ready to build with confidence



TOOL 2

The P.A.C.T. Feedback Framework

Use this four-step framework to structure your feedback constructively, focusing on partnership and shared goals. P.A.C.T. stands for ***Praise, Ask, Concern, Try***.

Step	Action	Example
Praise	Start by acknowledging the good intent behind the design choice.	*"I really like the idea of visualizing this complex process. Making it interactive is a great way to engage the learner."*
Ask	Ask a clarifying question to ensure that you fully understand the learning goal.	*"Could you tell me more about the single most important thing the learner should understand after this interaction?"*
Concern	State your technical or usability concern from a neutral, problem-solving perspective.	*"My only concern with building a custom 7-step drag-and-drop is the development time and the risk that it might not be fully accessible or mobile-friendly."*
Try	Propose a concrete alternative solution collaboratively.	*"What if we tried a different approach? We could create a simple animated GIF or a series of 'click-to-reveal' hotspots. It would deliver the same core message and be fully accessible."*

The Start of Your Partnership Journey

Moving from a task-executor to a design partner requires a shift in mindset. It's about seeing yourself as a co-creator of the learner's journey, not just the technical assembler of the files. You're moving from simply following a recipe to co-creating the entire menu.

The benefits are immense: you'll work more efficiently, your professional value will skyrocket, and you'll find deep satisfaction in knowing that you had a direct hand in creating something that truly works for the learner.

You don't have to change everything overnight. Start small. On your very next project, pick one screen and ask the Instructional Designer this simple question:

"What is the core learning intent behind this slide, and how can my technical skills best serve that goal?"

You'll be amazed where the conversation takes you.

Appendix: Quick-Reference Sheets

The "ID Lingo to Dev Speak" Translator

ID Lingo	Dev Speak (The Translation)
Engagement	An interaction requiring meaningful user input (more than just clicking 'Next') that directly serves the learning objective.
Scaffolding	A multi-level learning path. ***Level 1*** is the tutorial, ***Level 2*** is the first mission, ***Level 3*** is the boss battle.
Chunking	Breaking down a large block of data into smaller, manageable packets to avoid overloading the user's working memory (RAM).
Formative Assessment	A low-stakes ***unit test*** (like a knowledge check) to identify and fix bugs in understanding before the final deployment.
Summative Assessment	The ***final release build*** or ***UAT***. The final graded quiz that certifies that the user has met the objectives.

BONUS

The Pre-Flight Kickoff Questions

Ask these questions at the start of any new project to ensure clarity and alignment.

- What is the #1 thing the learner should be able to do after this course?
- Who is our target audience (tech-savvy, novice, etc.)?
- What are our primary technical constraints (browser, LMS, bandwidth)?
- Who is the final sign-off for technical and visual decisions?

Now what?

Now that you understand how to move beyond the storyboard and think like a true design partner, it's time to apply this approach to your real projects. The shift doesn't happen overnight. It happens one screen, one conversation, and one smarter decision at a time.

Tools matter here. Working collaboratively is much easier when your authoring environment lets you prototype ideas quickly, test alternatives, and adjust without friction. iSpring Suite supports this kind of workflow, giving you the flexibility to focus on learning intent instead of technical hurdles.

If you're looking for a practical starting point, the iSpring guide to building interactive courses is a useful next step. It shows how to translate learning objectives into meaningful interactions without overloading the learner. You may also find the accessibility best practices guide helpful as you begin taking stronger ownership of the learner experience.

Start small. Open your next storyboard and ask:

What is the learning intent here, and how can I implement it in the clearest, most effective way?

That question, paired with the right tools, marks the beginning of your shift from course builder to design partner.

About the Author

Harish Bhagavathula

@harishbhagavathula

Harish is an instructional designer and eLearning developer with over 14 years of experience. Throughout his career, he has observed developers often being seen as “the people who copy content into tools.” He has also seen the impact when developers step up questioning, contributing, and co-creating learning experiences.

Over the years, Harish has worked on various projects where small developer decisions made a significant difference in learner experience. Through this guide, he shares the mindset shifts, insights, and strategies that help developers position themselves as more than just builders; as creators who shape meaningful learning.

Welcome to the iSpring Community!

Blog →

Explore the exciting world of eLearning together with our helpful articles and how-tos.

Webinars →

Get eLearning tips and tricks from top-notch industry experts.

Guides →

Get step-by-step directions on how to create and launch eLearning.

YouTube →

Enjoy weekly eLearning videos and don’t forget to subscribe.