

Foundations of Instructional Design: Methodology and Learning Theories



Introduction to Instructional Design

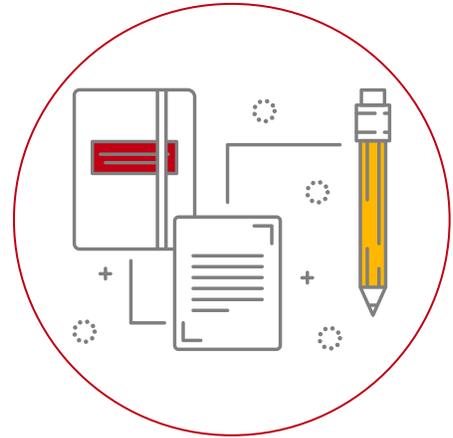
When an organization needs training solutions, the instructional designer must understand the business and individual needs that underlie the training initiative. This requires defining the business drivers for training program development and the organizational results needed or desired.

Once the designer has taken that critical first step, instructional design models and learning theories enter the picture to provide a systematic approach (or plan) for crafting effective and efficient training solutions that meet organizational and individual needs.

These plans are referred to as instructional systems design (ISD) models. ISD models are based on the systems approach; the output from one model phase provides the input for the next phase.

Instructional designers are then responsible for creating the course design and developing all instructional materials, including presentation materials, participant guides, handouts, and job aids or other materials. Instructional designers are commonly also responsible for evaluating training, including assessing what was learned and whether the learning solution led to measurable behavior change.

Theoretical models in the field of instructional systems development have been in place for many years, with ADDIE probably being the first and most recognized example. In recent research conducted by ATD (2015), more than 1,300 respondents listed the ADDIE model as the one they use most often. It was followed by Bloom's taxonomy and Kirkpatrick's four levels of evaluation. Clearly, while there are many models of ISD, the ADDIE model reigns supreme in terms of what professional instructional designers see as the model to reference and use as a foundation for their work.



What Are the Common Types of Instructional Design?

ADDIE has become the gold standard in ISD models because at its very core it reflects the most common elements of the ISD process—analysis, design, development, implementation, and evaluation. Almost all other ISD models contain these five core elements, but they're usually called something else, combined in some way, or have added elements of complexity connected to the ADDIE core. There are also more streamlined models now appearing that essentially amalgamate the elements into fewer individual elements while still retaining the ADDIE core as a foundation.

Some practitioners misinterpret the ADDIE model to suggest that the five elements are to be performed in a linear pattern, starting with analysis and ending with evaluation. However, strictly conforming to that rendering of the model will usually net a very simplistic design process with little in the way of dynamic responsiveness to the inevitable complexities and unique aspects of each specific set of variables in a project.

ADDIE at a Glance

Analysis

Analysis is theoretically the first, or entry, stage of the ADDIE model and is considered the data-gathering element. Here, instructional designers assemble all the information they can possibly gather about content, populations, delivery systems and methods, and budgets before they consider anything else. In this model, it is assumed that analysis is the first step and that it actually takes place in each instructional design process, which unfortunately is not always true.

Design

Design is the blueprinting stage of instructional systems, during which instructional designers create the blueprint for a project with all the specifications necessary to complete it. During this stage, instructional designers write the objectives, construct course content, and complete a design plan.

Development

Materials production and pilot testing are the hallmarks of development. At this stage, most non-designers, not working directly on a project, begin to see progress. Everything from lecture notes to virtual reality is brought from design to deliverable.

Implementation

The most familiar of the elements is implementation. At implementation, the design plan meets the learner and the content is delivered. The evaluation process that most designers and learners are familiar with takes place in this element. Evaluation is used to gauge the degree to which learners meet objectives and facilitators or technologies deliver the project.

Evaluation

Evaluation should not be listed last in the ADDIE model because it takes place in every element and surrounds the instructional design process. Evaluation is the process of determining both the learners' mastery of the material and the quality of a course design. It is generally seen as encompassing four basic types of evaluation, sometimes called levels, with many more layers of evaluation possible in any given scenario. Kirkpatrick (1959) named the four types of evaluation as reaction, learning, behavior, and results. Others have added a fifth type, named, among other things, return on expectation or investment.



SAM (Success Approximation Model)

SAM is another design model that emerged as an alternative to ADDIE, especially when creating e-learning solutions to drive performance improvement. This approach not only challenges the accountability of roles and ownership of deliverables, but also flies in the face of current multitasking project work. The primary difference in this approach is that smaller chunks are completely finished through locked-down teams in a fixed period of time (sprints). This is not prototyping—the deliverables are complete, useable components. The steps in SAM are to evaluate first, then design and develop the course, and then repeat the process by evaluating the first iteration of the course.

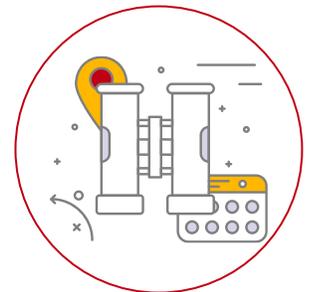
Iteration 1

The process repeats at least three times, starting and ending with evaluation. The initial evaluation looks at the situation, need, and alternative solutions. After each successive cycle of design and development, the emerging solution is evaluated to determine appropriateness and effectiveness. Information gathered in the initial evaluation is also re-examined for accuracy and sufficiency.

In the first iteration, design is kept to just listing objectives, sketching representative instructional treatments, and proposing methods to measure progress. Development is kept to preparing only representative content for each proposed delivery medium and instructional paradigm. Subsequent iterations will refine the work in terms of breadth, depth, and polish.

1. Evaluate: Begin with a quick evaluation (analysis) of the situation, need, and goals.

- Who are the learners and what needs to change about their performance?
- What can they do now? Are we sure they can't already do what we want?
- What is unsatisfactory about current instructional programs, if any exist?
- Where do learners go for help?
- What forms of delivery are available to us?
- How will we know whether the new program is successful?
- What is the budget and schedule for project completion?
- What resources are available, human and otherwise?
- Who is the key decision maker and who will approve deliverables?



2. Design: Quickly, but with thought, prepare a rough design for discussion.



- List and organize obvious goals.
- List behavioral objectives for each.
- List ways learner performance can be appraised.
- Select practical and appropriate delivery media.
- Sketch a few sample designs that appear to fit the situation and could reasonably be expected to achieve the goals. Be as visual as possible. (Sketching is an important activity we will return to in some detail later.)

3. Develop: Prepare prototypes using whatever tools can quickly provide a sense of the design idea in application.

- Select representative content to flesh out some of the sketches—just enough for understanding. Filler text can be used.
- Stay in sketch mode—nothing fancy here. Prepare bullet points rather than paragraphs; use rough art, snapshots, and homemade video rather than illustrations, professional photos, and commercial video.
- Assemble props that instructors or learners might use to perform activities.
- Focus on prototyping learner activities instead of presentation content.



Iteration 2

The iterative process returns to evaluation, one notch up. On the next go around, it's time to:

1. Evaluate: Determine the success of the first iteration.



- Was enough known about the situation, need, and goals? If not, it's time for some additional information gathering and analysis.
- What would and wouldn't work? Get some learners involved to help you decide. If instructors will deliver the instruction, conduct a mock class. Don't think it's too early.
- Where do alternatives need to be explored? Perhaps in this iteration, the team needs to sketch two or more alternative designs addressing the same content to compare.

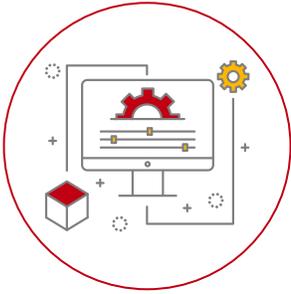
2. Design: Sketch new alternatives or refine previous ideas.

If evaluation determines that the previous cycle should be repeated, it's important to do it. A cycle that clarifies needs or discards initial ideas is a success and an important step forward, but only if taken advantage of. Repeat Iteration 1 if needed. When it's time to move on:

- Force a new design. Try creating a design that does not incorporate the design from the first iteration. It may be hard and frustrating at first, but people nearly always find that by imposing the restriction of doing something different, they actually create something better—and something they wouldn't otherwise have created.
- Identify content that previous designs didn't accommodate well and create initial designs for them.
- Flesh out more thoroughly those ideas retained from previous iterations to be sure the designs can serve as a solid foundation. Representative content should be used, but it's not yet time to work with the full bulk of it. If there are different types of content, however, representative chunks should be chosen and the design expanded as necessary.



3. Develop: Prototypes need to become more thoroughly representative of the final product.

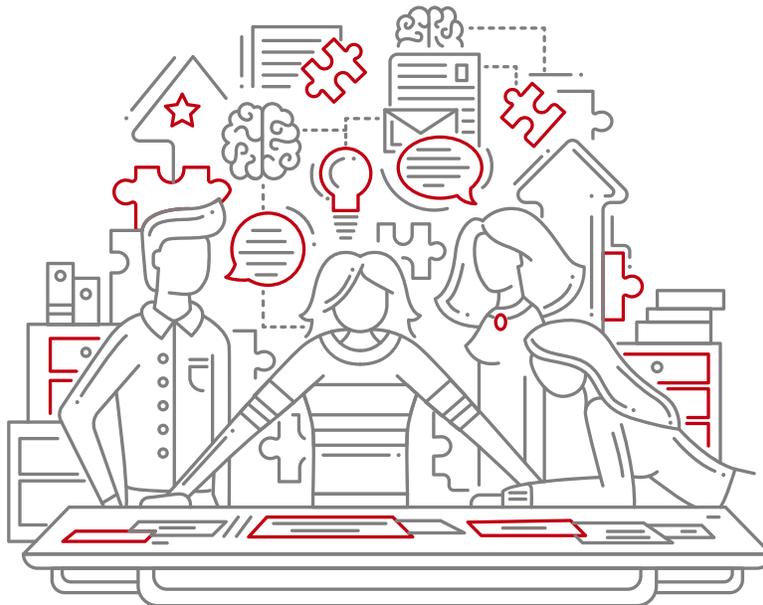


- Prepare learner materials. Prepare a set of learner materials using a format close to the format under consideration for the final materials.
- Test delivery. The means of delivery need to be tested soon, in either the next evaluation or the one in the cycle after that. The type of delivery determines the specific tasks.
- If instructor-led delivery is planned, prepare instructor notes and support materials to a sufficient level that an instructor who has not been involved in the project could deliver a segment of instruction.
- If using e-learning, prepare some interactive segments to test user interface designs and instructional approach.
- If using distance learning, test compatibility of presentation materials and learner responses with the communications system.

Iteration 3

Iteration 3 is similar to Iteration 2, although as confidence builds that issues have been properly handled, issues must cease to be re-examined. The iterations become much more focused on development than design.

Additional iterations often seem attractive if not compelling, but if all content areas have been included in the first three, it's rare that additional iterations would return results worthy of the time and effort. It's usually much better to put the product in use, get experience with it, and then consider another round of improvements.



LLAMA and Other Agile Methods

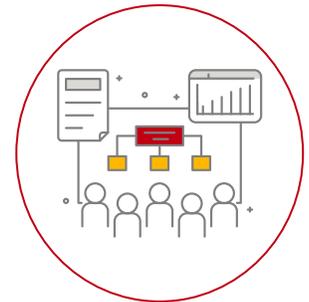
LLAMA (Lot Like Agile Methods Approach) is another Agile design methodology that many instructional designers have begun to embrace. Agile project management is an iterative, incremental process and approach for guiding the design and build of projects in a highly flexible and interactive manner. In addition, Agile focuses on maximizing customer value and fostering high team engagement. What's more, LLAMA presents a framework of values that enables teams of programmers to develop software in ways that accommodate changes to underlying needs and a continuous discovery of requirements throughout the project effort.

LLAMA makes some adjustments to the traditional Agile methodology by taking into consideration the syntax and structure of learning objectives, as well as the idea that instructional designers are likely to be working on multiple projects at one time.

Applying Agile to ID

With the success of the Agile project management in the software industry, it comes as no surprise that L&D practitioners have sought to adopt it. Indeed, many of its guidelines probably sound very familiar to instructional designers, such as:

- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity—the art of maximizing the amount of work not done—is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



As you can see, in many respects, the design-build aspect of software design and development is akin to that of instructional design and development, and this similarity holds even stronger when we're developing e-learning or other digital learning experiences.

There are several key differences between the two types of work, though. Some distinctions include:



- Instructional designers need to focus on learning objectives and performance outcomes in addition to functions and features.
- Most instructional designers work on several projects at once, while software developers usually are dedicated to a single team.
- Instructional designers often need to wait for content or input from subject matter experts, and have to account for that downtime in their project plans.

Instructional Design Project Management: A Checklist for Getting Started

● Identify a business goal.

Define the business goal in as much detail as is useful for the project. Think along the lines of business strategies, such as increasing revenue or income, decreasing costs, improving quality, or expanding the capacity of the organization. Connecting the project and its goals with the bigger picture and vision of the organization helps everyone stay motivated when challenges arise.

● Start with the learner in mind.

Identifying a primary learner persona (PLP) will help the ISD project management team make sometimes tough decisions about the direction of the project as it progresses, such as how regularly the PLP would use the training and in what context.

● Define the scope of the project with learner stories.

“Using traditional Agile story mapping for training projects tends to result in very information-driven courses,” notes Torrance, and this can be frustrating for the project team. LLAMA, by contrast, incorporates Cathy Moore’s action mapping, which uses proposed user actions to generate stories for project planning.

● Chunk the work effort.

Break the project into workable chunks. You can do this by taking each of the user stories— independent units of scope that can be prioritized, assigned, developed, and tested along the project’s path to completion—and breaking it into the tasks required to complete the story, such as identifying a location for the photo shoot or selecting the SME for the shoot.

● Plan the work and work the plan.

Include frequent reviews of the project, and avoid scheduling release dates for multiple projects all at once, suggests Torrance. “Remember that the project schedule is an estimate. The further out it goes, the less likely it is to be completely accurate, so plan accordingly.”

make text in this box flush left.

Source: Torrance, M. 2014. "Agile and LLAMA for ISD Project Management." *TD at Work*, November. Alexandria, VA: ATD Press.

Needs Assessment Checklist

This checklist will help you ensure that you've thoroughly explored everything within your needs assessment. Refer to this checklist periodically to help guide you through your assessment.

Identify

- Identify gaps
- Consider data collection sources
- Identify stakeholders and decision makers
- Identify required resources

Analyze

- Identify target audience, prior knowledge, and demographics
- Identify, sequence, and prioritize tasks involved
- Gather data
- Analyze data
- Summarize findings

Decide

- Identify potential solutions
- Prioritize solutions
- Propose solutions
- Generate report

Ready to Bring Your Design Skills Up a Notch?

Join 300 of your industry colleagues at ATD's Core 4 conference, March 14-15, 2019, in Anaheim, California. Gain the knowledge you need in the four critical areas of talent development—instructional design, training delivery, measurement and evaluation, and learning technologies.



Attendees love Core 4 because it delivers essential professional development in a more intimate setting. You'll leave with actionable takeaways to design better learning programs, engage learners, change behaviors, and improve performance.

Learn more and register at <https://core4.td.org/About/Anaheim>.

