

# HOW TO DESIGN SOFTWARE IMPLEMENTATION TRAINING

by Jasmine Salazar and Laura Page

New software implementation comes with a host of challenges. Without a solid change management strategy, organizations can fail to fully realize the benefits of large-scale software initiatives, and a lack of training can be a major setback in recovering the cost of implementation.

Our checklist below outlines what your implementation strategy needs to be successful:



Needed	Consideration
<input type="checkbox"/>	<b>1. Involve training and communication through the project lifecycle.</b> Have training and/or communication activities been aligned through the phases of the implementation lifecycle? What <b>messaging and activities</b> should we identify to support this?
<input type="checkbox"/>	<b>2.</b> Have you taken the <b>culture of your company</b> into consideration? Personalize your approach based on the different working styles of your team.
<input type="checkbox"/>	<b>3. Leverage vendor provided training</b> where you can. Fill in the gaps in the vendor training, such as company policies and procedures as well as any customization you plan to do.
<input type="checkbox"/>	<b>4.</b> Define <b>learning objectives by audience</b> . Make training <b>applicable by</b> mapping desired outcomes to behaviors.
<input type="checkbox"/>	<b>5.</b> Review your implementation objectives and determine if there are any existing KPIs or metrics to track against. <b>Determine the ROI of your implementation</b> based on new or existing performance metrics (KPIs) to <b>model the path to success</b> .



Remember: Training initiatives are far more sustainable when they connect to the culture and communication styles of individual teams. Determine the behavioral objectives for each audience, and set KPIs to measure the success of your personalized programs.