TOP 10 LIMS IMPLEMENTATION MISTAKES

Lessons from 30+ Years in the Lab, on the Bench, and in the Boardroom

I have worked both sides of the LIMS table, as a bench chemist, an implementation project manager, and now as an independent, vendor-neutral consultant. Over dozens of projects, regardless of the vendor, the same handful of mistakes show up repeatedly. Here are the ten I see most often, and what I recommend instead.

1. Treating Vendor Selection as a Checkbox Exercise

A feature matrix and a slick demo tell you what a system can do in theory, not how much configuration or workaround it will take to do what your lab does in practice. I have seen labs pick a platform on paper, then spend the first month of implementation discovering the gap.

Instead: *Bring real workflows, sample types, and edge cases into the evaluation. Ask vendors to demo your process, not their canned one.*

2. Underestimating Data Migration

Legacy data is messier than anyone remembers, historical records rarely map cleanly to a new data model, and the effort to cleanse and validate it can rival the effort of the implementation itself.

Instead: *Start data assessment early, well before go-live planning, and budget real time for cleansing, not just extraction and load.*

3. Skipping the SOP Rewrite

Bolting a new system onto old paper-era SOPs with minimal edits leaves a validated system running alongside procedures that no longer reflect reality, which creates both compliance risk and daily confusion for analysts.

Instead: *Treat SOP revision as a core deliverable of the project, not a cleanup task for after go-live.*

4. Letting IT Drive Requirements Without Lab Input

LIMS projects are technology projects, but they are lab projects first. Requirements gathered mainly through IT tend to optimize for infrastructure while missing the daily realities of sample handling and analyst usability.

Instead: *Give bench scientists and lab supervisors real weight in requirements sessions, not just a review pass at the end.*

5. Treating Validation as an Afterthought

In regulated environments, validation under GAMP 5 and 21 CFR Part 11 is not a phase you bolt on at the end. When planning starts late, teams retrofit documentation to match what was built, which weakens the audit trail.

Instead: *Build your validation approach alongside your requirements, not after configuration is already done.*

6. No Dedicated Project Owner on the Lab Side

Without a lab-side owner who has real authority and bandwidth, decisions get made by whoever happens to be in the room, and that is not always the right person for the decision at hand.

Instead: *Identify a lab-side project owner early, and protect their time for the duration of the project.*

7. Skipping Workflow Discovery, Then Insisting the New System Match the Old One

Skip real discovery, and the default requirement becomes I want the system to work like we do now. That is one of the most damaging things I hear on a project, because if every workflow must stay identical, there was no real reason to implement new software in the first place.

Instead: *Document current-state workflows first, then challenge each one and ask if it is genuinely necessary or just familiar.*

8. Underestimating Instrument Integration Complexity

Every instrument, driver version, and data format is its own small project. Labs with a mixed fleet of older and newer instruments frequently find integration takes longer than the core LIMS configuration.

Instead: *Inventory every instrument and interface early, and treat integration as its own workstream with its own timeline.*

9. Insufficient End User Training Before Go Live

A single training session a week before go-live rarely builds real proficiency. When training is rushed, adoption suffers, workarounds multiply, and the help desk gets flooded in the first weeks after go-live.

Instead: *Plan for hands-on training with real samples and real scenarios, not just a walkthrough of the screens.*

10. No Plan for Post Go Live Support

Momentum and budget tend to disappear right after go-live, just as analysts are hitting edge cases the training did not cover. Nobody delays a go-live on purpose, but plenty of teams delay planning for the weeks that follow it.

Instead: *Budget for a defined hypercare period with dedicated support, not just a ticket queue.*

None of these are exotic. They are ordinary, predictable pitfalls that show up on nearly every LIMS project, which is exactly why they are worth naming out loud. Most are avoidable with the right planning, the right voices in the room, and someone who has seen the pattern before and can call it early. If you are heading into a LIMS project, I would be glad to talk it through.

About Cogsult

Cogsult is an independent, vendor-neutral LIMS and lab informatics consulting practice built on more than 30 years of bench-to-boardroom experience. I work across multiple LIMS platforms with clients in pharma, environmental, food and beverage, chemicals, manufacturing, and contract labs, helping teams avoid exactly these kinds of mistakes before they become expensive ones.

Matt Citardi | Principal Consultant, Cogsult

cogsult.com | matt.citardi@cogsult.com