

# A practical guide to choosing, implementing, and improving legal systems.

A free field guide for law firms and legal teams navigating software, integrations, data, security, and change.

## How to use this guide

Use it as a planning companion before a new software purchase, during a troubled implementation, or when your current systems have become harder to manage than the work they support.

TIM SEEDORF CONSULTATION SERVICES

## Always a free consultation & conversation

This guide offers operational and technology guidance, not legal advice. Every team should review its own ethical, privacy, security, records, and regulatory requirements with appropriate counsel.

# A software project is a business change project.

The most expensive mistake is selecting a product before defining the problem. Start with the work: what takes too long, creates risk, gets duplicated, or depends on one person knowing the workaround?

## A useful project statement

For [team], improve [workflow or outcome] by changing [process, system, or behavior], while protecting [risk, service level, budget, or timeline].

## Before you look at demos, answer these six questions:

- What specific outcome would be noticeably better six months from now?
- Who performs the work today, and who is affected when it breaks?
- What information is created, received, changed, or retained?
- What existing systems, vendors, or integrations constrain the decision?
- What is the cost of doing nothing for another year?
- Who has authority to make decisions, approve spend, and resolve tradeoffs?

# Move from uncertainty to a system your team can sustain.

| PHASE        | PRIMARY OUTPUT                          | QUESTION TO ANSWER                        |
|--------------|-----------------------------------------|-------------------------------------------|
| 1. Discover  | Current-state map and problem statement | What is happening now, and why?           |
| 2. Define    | Requirements and success measures       | What must the future system accomplish?   |
| 3. Evaluate  | Shortlist and decision record           | Which option best fits the work and risk? |
| 4. Design    | Future-state workflows and data plan    | How will the new system work in practice? |
| 5. Deliver   | Configured system, migration, training  | Is the team ready to use it?              |
| 6. Stabilize | Support plan and health review          | Is adoption holding and risk controlled?  |

## Keep a decision record

Capture major decisions, owner, date, alternatives considered, and the reason for the choice. It prevents the project from relitigating old questions and makes transitions far easier.

# Separate must-haves from wishes.

A good requirement describes an outcome, not a feature name. 'The system should produce a conflict-aware matter access report' is more useful than 'needs better security.'

| CHECK    | WHAT GOOD LOOKS LIKE                                                                                       |
|----------|------------------------------------------------------------------------------------------------------------|
| Yes / No | Business: it supports the matters, practice groups, clients, and service model you actually have.          |
| Yes / No | Workflow: common tasks can be completed with a clear owner, status, and handoff.                           |
| Yes / No | Data: essential fields, documents, and history can be migrated or connected without losing meaning.        |
| Yes / No | Security: roles, access, audit history, encryption, and vendor controls match the sensitivity of the work. |
| Yes / No | Integration: important systems can exchange the right data at the right time, with an owner for failures.  |
| Yes / No | Operations: support, reporting, training, and change ownership exist after the launch team leaves.         |

Prioritize every item as one of four labels: **must have**, **should have**, **nice to have**, or **not in this phase**. This is where scope becomes manageable.

# Look beyond the demo.

A compelling demonstration proves that a product can do something. It does not prove it will work with your data, your people, your permissions, or your deadlines.

## Use the same scenarios with every finalist

- Open a new matter, apply the correct template and permissions, and assign work.
- Find the current approved document and its prior versions from a realistic search query.
- Route an item for review, capture a decision, and show its audit history.
- Create a billing or reporting record using your real data fields and exceptions.
- Demonstrate how an ethical wall or restricted matter is administered and tested.
- Show a failed integration, its alert, and the process to correct and reconcile it.

### Ask vendors directly

What implementation work is required from us? What is excluded? Which capabilities require an add-on? How do customers export their data? What happens during an outage? Who owns the integration when it fails?

# An integration is a product, not a checkbox.

The real work begins after two vendors say they integrate. Confirm exactly what moves, when it moves, how duplicates are handled, and who is responsible when something goes wrong.

| CHECK    | WHAT GOOD LOOKS LIKE                                                                              |
|----------|---------------------------------------------------------------------------------------------------|
| Yes / No | Name the source system, destination system, data owner, technical owner, and business owner.      |
| Yes / No | Define the record key used to match data across systems.                                          |
| Yes / No | Document direction and timing: one-way, two-way, real-time, scheduled, or manual.                 |
| Yes / No | Identify field mappings, required fields, transformation rules, and data that should not move.    |
| Yes / No | Plan for errors: alert recipients, retry behavior, correction process, and reconciliation report. |
| Yes / No | Test permissions, restricted matters, deleted records, and unusual but realistic edge cases.      |

**Data migration rule:** migrate the information that has a continuing business, legal, or operational purpose. Archive or retain the rest in a documented, searchable, defensible way. Do not move decades of inconsistent data just because it exists.

# Build risk controls into the project, not after it.

Legal teams work with confidential information, restricted matters, client obligations, and records that may be subject to hold, retention, or production requirements. Technology decisions should make those obligations easier to fulfill.

## Focus conversations on:

- Access model: who can see, edit, export, share, or administer information?
- Ethical walls: how are restricted matters identified, applied, monitored, and removed?
- Auditability: what actions are logged, how long are logs retained, and who can review them?
- Information lifecycle: where do records live, how are they retained, and what happens on legal hold?
- Vendor assurance: what security, privacy, incident, subcontractor, and location commitments are contractually supported?
- AI and automation: what inputs are used, whether data is retained or used for training, and how output is reviewed?

### Important

Technology guidance is not a substitute for legal advice. Involve the appropriate internal stakeholders, including information security, records, privacy, conflicts, and counsel, before relying on a system design for compliance.

# Configuration is only half the launch.

The best design fails if people do not understand what is changing, why it matters, and where to get help. Plan implementation as a sequence of decisions, tests, communications, and reinforcement.

| CHECK    | WHAT GOOD LOOKS LIKE                                                                                               |
|----------|--------------------------------------------------------------------------------------------------------------------|
| Yes / No | Pilot with representative users and real scenarios before the organization-wide launch.                            |
| Yes / No | Define acceptance criteria: the launch is not complete until critical workflows and reports pass testing.          |
| Yes / No | Create role-based training: attorneys, paralegals, finance, administrators, and leadership need different support. |
| Yes / No | Publish simple job aids for the tasks users perform most often.                                                    |
| Yes / No | Prepare a go-live support plan with a visible owner, response path, and daily issue review.                        |
| Yes / No | Measure adoption using observable behavior, not just attendance at training.                                       |

A launch communication should answer four things: what is changing, when it changes, what each person needs to do, and where help is available.

# Stabilize, learn, and improve.

Go-live is the start of operating a new system. Make space for a short stabilization period, then schedule regular health reviews so small issues do not become permanent workarounds.

## A 30 / 60 / 90-day review

| WHEN    | FOCUS     | EXAMPLES                                                                     |
|---------|-----------|------------------------------------------------------------------------------|
| 30 days | Stabilize | Fix critical defects, answer repeat questions, confirm support ownership.    |
| 60 days | Adopt     | Review usage patterns, training gaps, reports, and unhelpful workarounds.    |
| 90 days | Improve   | Prioritize enhancements, retire duplicate tools, and confirm value measures. |

### Support does not need to be all or nothing

Support contracts for one off incidents can help when a workflow, vendor, integration, or implementation needs focused attention. Ongoing advisory support can provide continuity for teams managing multiple systems and changes.

# Bring a little context. Leave with a clearer path.

You do not need a complete project brief to begin. A useful first conversation can start with a system that feels difficult, a vendor decision, an upcoming change, or a concern that your tools are not supporting the work as well as they should.

## Bring these five items if you have them:

- The problem you are trying to solve, in plain language.
- The people and teams affected by the change.
- The systems, vendors, and deadlines already in play.
- Examples of the workflow, documents, reports, or data involved.
- The outcome that would make the effort worthwhile.

### Always a free consultation & conversation

Tim Seedorf Consultation Services provides independent guidance for legal teams navigating software, systems, integrations, security, and operational change. Start with the question you have today.

**[timseedorf.com](https://timseedorf.com) | [consulting@timseedorf.com](mailto:consulting@timseedorf.com)**

Please do not include confidential, privileged, or time-sensitive information in a general contact request.