

Before you start building, make sure you are building the right system.

A focused engagement for organizations that need purpose-built software but want clarity on requirements, architecture, scope, and implementation before committing to a larger development project.

1-2 weeks

Typical engagement

From \$7,500

Fixed fee, scope confirmed up front

One application

Buildable plan + implementation roadmap

A good fit when...

- A business process has outgrown spreadsheets, email, shared drives, or disconnected SaaS tools.
- An aging internal application needs replacement or substantial rework.
- The team has requirements or a backlog but no clear technical owner or architecture.
- Off-the-shelf software does not fit the workflow well enough.
- Several systems need to work together behind a simpler application or interface.
- A previous development effort stalled because scope or architecture remained unclear.

More features are not automatically better.

Discovery identifies the smallest useful version of the application that solves the real business problem, separating what must ship first from what can wait.

What we examine

- Business problem, users, current workflow, and desired outcome
- Existing applications, tools, requirements, and priorities
- Integrations, APIs, data sources, and systems of record
- Authentication, security, privacy, compliance, and cloud constraints
- Expected usage, operational needs, build-vs-buy options, and technical risks

Typical input: one accountable owner, 3-5 stakeholders, existing notes or requirements, access to current tools/workflows, and relevant system/integration information.

What you receive

Problem & Workflow Definition

Business problem, users, current process, pain points, intended outcome, and important constraints.

Prioritized Requirements

Must-have, should-have, later-phase, and explicitly out-of-scope requirements to keep the first release focused.

Solution Architecture

Recommended components, application boundaries, hosting/runtime approach, authentication, integrations, and key infrastructure decisions.

Data & Integration Design

Major entities, data flows, systems of record, API/integration requirements, and ownership boundaries.

MVP / Phased Delivery Plan

A practical sequence for delivering useful software in manageable stages rather than attempting everything at once.

Implementation Estimate & Roadmap

Major workstreams, sequence, dependencies, delivery risks, approximate timeline, and estimated implementation investment.

Optional technical validation

Small architecture spikes, API tests, wireframes, or focused proof-of-concept code may be used when they reduce uncertainty around an important assumption.

Not a production build

The base sprint does not include full implementation, complete UX/UI design, production data migration, production integrations, or production support unless separately scoped.

How it works

1 Understand the problem

Kickoff, workflow review, stakeholder interviews, existing-system review, constraints, and desired business outcome.

2 Simplify & design

Prioritize requirements, remove unnecessary scope, evaluate integrations/data, surface risks, and define the architecture.

3 Plan implementation

Create the phased roadmap, implementation estimate, key dependencies, and recommended next step.

CLARIFY

Turn an incomplete business need into a concrete problem definition, prioritized scope, and buildable plan.

DE-RISK

Surface integration, data, security, architecture, and delivery risks before major implementation spending begins.

Senior engineering context

20+ years enterprise software · 12+ years Microsoft Consulting Services · AWS Solutions Architect · AWS AI Practitioner

Have an application you need to define before you build?

Start with a focused conversation. No obligation to continue into implementation.