Build the service you wish existed.

AI expands what we can attempt.
Good engineering turns it into a service people can trust.

For citizens

Clarity about what comes next.

For builders

More useful work within reach.

For leaders

A clearer path from intent to delivery.

The best outcome is a citizen who knows what happens next.

A clear status. A useful explanation.
A next step they can actually take.

Less uncertainty / More agency / Time returned

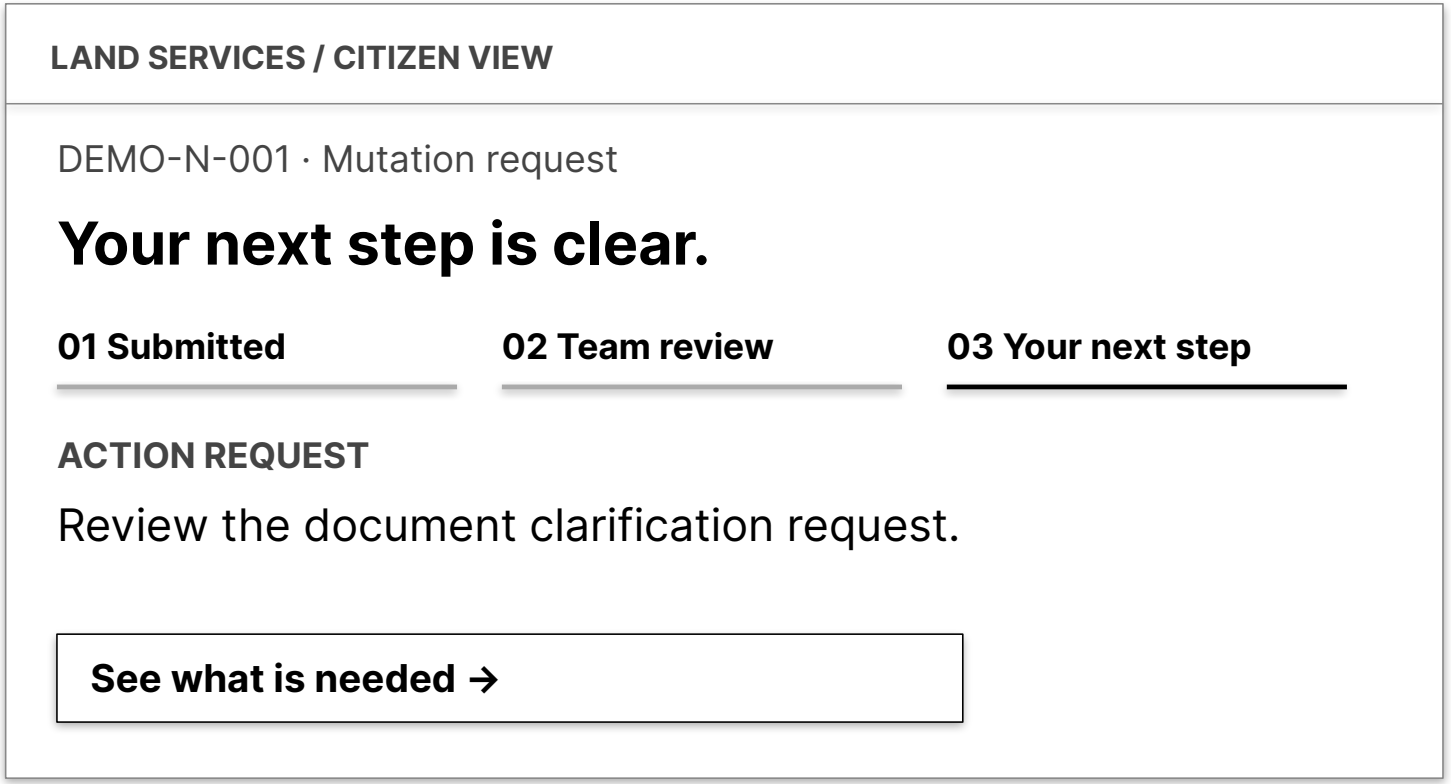
Make the next step unmistakable.

IMAGINE THIS SCREEN

Less guessing.
More clarity.

A status people understand.
A reason they can read.
An action they can take.

Illustrative concept · fictional data



Give the team a view it can act on.

IMAGINE THE TEAM'S VIEW

A shared view. A named next step.

See what needs attention.
Bring the right people together.
Keep responsibility visible.

Illustrative concept · fictional data

LAND SERVICES / OPERATIONS VIEW

Turn a queue into useful questions.

Application	Next action	Owner
DEMO-N-001	Document clarification	Circle team
DEMO-N-002	Payment reconciliation	Revenue team
DEMO-N-003	Specialist review	Named reviewer

Put a small team of agents to work.

One shared brief. Focused drafts. One integrated review.

01 / EXPLORE

Make the code legible.

Map the relevant workflow and the questions it leaves open.

02 / SPECIFY

Make the rule testable.

Draft examples and checks from agreed requirements.

03 / PREPARE

Make the change reviewable.

Prepare a small patch and explain what it changes.

Illustrative workflow · approved tools · synthetic inputs

Turn the ambition into a change the team can stand behind.

01

Choose the outcome

One citizen or team need.

02

Map the boundary

Policy, code, identity and data.

03

Build the evidence

Small changes, explicit tests.

04

Prepare the release

Review, owner and recovery.

The useful action works.
The boundary holds.

LIVE / SYNTHETIC DATA

Preview the intended mutation action.

Check that a clerk cannot preview ownership update.

Run the 43 workflow assertions.

Watch for

Useful behaviour.
Explicit authority.
A result we can inspect.

Let the evidence carry the promise.

Concrete starting points for the team’s own standards.

43

Workflow assertions

Roles, districts and allowed states.

83

Model-field checks

A note edit preserves ownership fields.

65

Deed-intake checks

Validation and quarantine planning.

Executed on supplied synthetic fixtures; each result is limited to its stated contract.

Pick the service you want to make possible.

Write one outcome from the backlog.
Then name the hardest question we must answer.

A citizen outcome / An engineering question / A team that owns it

Keep the ambition.
Engineer the confidence.

AI increases what we can attempt.
Evidence decides what we can release.

Data boundaries / Review capacity / Release control

Code volume can grow faster than review capacity.

More code makes understanding, tests and accountable review more valuable.

Small diffs / Named maker and reviewer / Tests and DevSecOps evidence

Give new joiners a map they can verify.

Find the code

Exact search, symbols and a workflow map.

Add retrieval

Approved source, decisions and commit-pinned citations.

Check the answer

A maintainer verifies the path and its tests.

A helpful agent can still cross a boundary.

01 Citizen records enter prompts, logs or attachments.

02 A credential lands in source or an artifact.

03 Untrusted text becomes an instruction to a tool.

04 A package or plugin arrives without review.

Treat repository text as input to evaluate.

Useful context

Reviewed conventions, source and task requirements.

Untrusted content

Issue comments, retrieved text, tool output and new plugins.

The agent can plan.
Your application keeps the keys.

01

Plan

Memory and retrieved text provide context.

02

Ask a narrow tool

Request one permitted status result.

03

Verify and return

Application checks identity, scope and minimum fields.

An AI service still depends on ordinary software.

Unexpected visibility

Some users could see another active user’s chat titles.

A familiar dependency

The reported defect was in the Redis Python client.

Assam decision

Minimize prompt data and test identity boundaries in shared infrastructure.

Make the approved route the normal route.

01

Developer

Approved client and identity.

02

Data boundary

Minimize; validate; reject unknown fields.

03

Gateway

Approved model routes and budgets.

04

Evidence

Metadata audit and reviewed changes.

LiteLLM can centralize policy. Someone must own it.

Control access

Authenticated callers, virtual keys, approved routes.

Control operations

Budgets, retention, updates and an emergency stop.

Close bypasses

Govern direct-provider access and metadata logging.

A filter can miss what a person can identify.

LIVE / SYNTHETIC DATA

Inspect a minimized synthetic request.

Reject an unapproved full-record field.

Observe a fictional name surviving the filter.

Watch for

Minimize first.
Test misses.
No automatic forwarding.

Local inference changes one part of the data path.

Pilot an exact artifact

Model revision, license, hardware and PHP task quality.

Review the surrounding system

Agent tools, plugins, telemetry, logs and network destinations.

Assamese voice is a pipeline, not one model.

Declared Assamese support. Local quality and latency still need measurement.

Hear

IndicConformer: Assamese speech recognition.

Translate

IndicTrans2: a checkpoint for each direction.

Speak

Indic Parler-TTS; IndicF5 as a comparison.

Protect the voice before it becomes text.

01

Minimize capture

Audio itself can identify a person.

02

Process deliberately

Local ASR, approved routing and short retention.

03

Preserve meaning

Check names, land references, numbers and negation.

04

Offer a fallback

Confirm critical fields; allow text or human help.

Keep credentials out of the agent’s working set.

Before commit

Local checks and safe configuration examples.

At the repository

Central checks that cannot depend on local hooks.

At runtime

Scoped credentials outside source and build output.

Deleting a secret does not revoke it.

01

Contain

Contact the credential owner.

02

Invalidate

Revoke or rotate through the approved process.

03

Investigate

Review use and affected systems.

04

Repair

Remediate source, artifacts and workflow.

The security scanner became the dependency risk.

19 March / Trivy

A compromised release and action tags entered trusted build workflows.

24 March / LiteLLM

Two PyPI packages were compromised; the vendor reported a likely Trivy link.

Assam decision

Separate scanning from publishing authority. Record the exact artifacts used.

Your supply chain is all the code you run.

01

Application

PHP, Java and JavaScript plus their dependencies.

02

Build

CI actions, install hooks and security tools.

03

Artifact

Container layers, configuration and release records.

04

Service

Deployment authority and operational recovery.

Review what enters. Keep updates deliberate.

Lock application dependencies

Review manifest and lockfile changes; inventory copied PHP libraries.

Review install-time execution

Composer plugins/scripts and npm hooks run inside a restricted job.

Pin reviewed build inputs

CI commit SHAs and image digests, with an update owner.

Check known advisories

Record inventory and feed age; missing coverage stays visible.

One missed server. A system-wide loss.

1 of 8 servers

A code deployment missed one server, leaving old behavior active.

About 45 minutes

Erroneous trading led to a loss exceeding $460 million.

Assam decision

Verify each deployed instance and reconcile outcomes before expanding rollout.

A build job should not hold the keys to everything.

Contributor checks

Untrusted code, isolated runner, synthetic data and minimal permissions.

Protected release

Reviewed workflow, constrained short-lived credentials and explicit approval.

A failed scanner is not a clean scan.

Completed with findings

Triage evidence, impact, owner and retest.

Completed with no findings

State the rules, inputs and limits of coverage.

Failed or incomplete

Hold the required check; fix execution or record an approved decision.

Inventory, provenance and signatures answer different questions.

SBOM

Which recorded components are in this artifact?

Provenance

Which source and build process produced it?

Signature

Does it match the expected identity and verification policy?

Build once. Promote the same digest.

01

Build

Reviewed inputs become an identified artifact.

02

Verify

Tests and reports name that exact artifact.

03

Promote

Protected approval deploys the same digest.

04

Observe

Check workflows; retain a tested recovery path.

Deploy the first defensive toolchain.

SAST / contracts

Semgrep CE + PHPUnit; CodeQL where its language coverage fits.

SBOM / SCA

cdxgen → CycloneDX; OWASP dep-scan → VDR, reachability and license review.

Secrets

TruffleHog for history; axi for the pre-push workflow and central enforcement.

DAST / API

ZAP Automation against synthetic staging and reviewed OpenAPI.

AI assist

ZAP LLM/MCP can explain evidence; it cannot authorize scans or releases.

Later: ScanCode.io

License, copyright, source and richer SBOM pipelines after the gate is owned.

docs.semgrep.dev · github.com · zaproxy.org · scancodeio.readthedocs.io · docs.github.com

An API needs more than a URL.

Inventory

Owner, operation, audience and data classification.

Contract

Reviewed OpenAPI, examples and authorization rules.

Publication

Explicit APISIX route and deployment evidence.

Internal does not mean trusted.

Authority

Workload identity plus user, role and record checks.

Failure

One authoritative writer; timeouts, idempotency and reconciliation.

Follow the journey across services.

01

Gateway

APISIX route and identity decision.

02

Services + queues

Trace calls, retries and RabbitMQ work.

03

Collector

OpenTelemetry with minimal attributes.

04

Owned dashboard

Grafana stack or SigNoz; an alert owner.

“We are already on PHP 8” is not enough.

PHP 7.4 and 8.1

Both beyond upstream security support.

Laravel 11

Security support ended in March 2026.

Other legacy branches

Review Spring Boot 2.7 and AngularJS exposure.

php.net · laravel.com · spring.io · docs.angularjs.org

PHP needs its own security clinic.

Language behavior

Implicit conversions and native object deserialization.

Framework boundaries

Permitted fields, output encoding and document intake.

Deployment history

PHP-CGI and FPM incidents had concrete prerequisites.

A citizen form cannot approve a mutation.

Citizen-controlled

A validated correspondence note in a permitted draft.

Server-owned

Recorded owner, citizen, district and approval state.

Treat every deed and remark as untrusted input.

01	**Imports**	Bounded data schemas; no untrusted native deserialization.
02	**Attachments**	Inspect bytes; private quarantine; server-generated names.
03	**Rendering**	Keep Blade escaping; encode for the actual output context.
04	**Legacy controllers**	Input cleaning cannot replace validation and output handling.

A land reference carries meaning.

D01 / C01 / 000042

Preserve every character and the full namespace.

42 is a quantity

Numeric conversion discards the leading zeroes.

A different language can prevent some mistakes.

Encode separate meanings

DistrictId, LandReference and Money become distinct types.

Catch wrong combinations earlier

A compiler can reject passing Money where LandReference is required.

Use AI to turn alerts into reviewable evidence.

01

Read

Scope source and the relevant control.

02

Explain

Separate observations from assumptions.

03

Propose

Draft a minimal fix and tests.

04

Check

Run evidence; keep human disposition.

OWASP Top 10:2025 becomes tests.

Top 10:2025

Awareness vocabulary: access, misconfiguration, supply chain, crypto, injection and more.

ASVS

Turn the category into a verifiable requirement and an acceptance test.

Land workflows

Role, district, state, exact identifier and audit assertions.

Closure evidence

Rule, scope, artifact, result, reviewer, retest and remaining limit.

Ask the test to describe the boundary.

Positive contract

The intended user can complete the intended action.

Negative contract

The wrong role, district or state stays blocked.

Eight red tests. One small patch.

LIVE / SYNTHETIC DATA

Read the exact payment-preview contract.

Run the safe incomplete starter: eight failures.

Inspect the minimal solution: 49 checks pass.

Watch for

A precise change.
Preserved denials.
A visible test result.

A passing money function is only one piece.

Trusted amount

Due comes from server state.

Authorized operation

Identity and record scope are enforced.

Reliable collection

Idempotency, ledger integrity and reconciliation.

Give every product team a security champion.

Inside the team

One champion and backup coordinate questions, findings, tests and follow-up.

Across the department

Central security, DevOps and SRE provide expertise, standards and escalation.

Make the role possible, then measure the result.

Protected time

Manager-approved capacity, training and a backup.

Practical support

Central buddy, weekly office hours and a peer network.

Visible follow-through

Owned findings, reviewed fixes and a Day 15/30 check.

Search and dashboards need an access policy too.

Owner search

Purpose, district scope, rate limits and audit.

Leadership dashboards

Authorized level, minimized fields and safe exports.

Citizen nudges

Verified recipient path and minimal message content.

Mission 30 starts with one reliable path to release.

01

Week 1

Flagship tests, telemetry and initial gates.

02

Week 2

Triage owners; stable scanner coverage.

03

Week 3

Reviewed fixes and staged migration.

04

Week 4

Independent checks, rollout and playback.

My proposed Mission 30 commitments.

Lead Goal 6 kickoff

With Prashant and Mriganka: scope, triage and evidence.

Support Goals 1, 3 and 7

Security contracts, pipeline checks and migration review.

Consult on 9, 12 and 13

Search, dashboard and nudge access boundaries.

Measure speed and trust together.

Delivery

Time to reviewed change; review effort.

Quality

Escaped regressions; tested workflow boundaries.

Security

Coverage, reviewed backlog and expired exceptions.

Now turn the idea into evidence.

One table. One workflow. A decision you can explain.

Write the brief / Check the result / Own the next step

Make a table. Give everyone a job.

Workflow expert

Explain the rule and citizen impact.

Driver + reviewer

Work with the tests; challenge the evidence.

Owner + recorder

Capture the decision and next step.

Predict the outcome before seeing the answer.

A / useful preview

A North clerk previews mutation on an active North record.

B / role boundary

The same clerk requests ownership-update preview.

C or D / one more boundary

Compare missing and another district, or a record on hold.

Write the brief. Then inspect the draft.

Constrain the task

Fictional fixture, exact rule and two complete comparisons.

Ask for evidence

Cite source and tests; preserve denials and preview scope.

Challenge the answer

Find an unsupported claim or an assertion that proves too little.

Run Semgrep, then check the whole result.

Run the real rule

python3 tooling/verify_semgrep.py

Read the finding

review-needed.php → one narrow logging candidate; reviewed.php → zero.

Keep the boundary

Run the PHP preview and record what SAST proves—and what it cannot.

Would you accept this change?

Decision

Accept for the fixture, revise, or hold.

Evidence

Each expected result has a reason and a checked comparison.

Uncertainty

Name one thing this fixture cannot establish.

Keep the evidence. Bring the question tomorrow.

Today

A brief, checked outcomes and a review decision.

Tomorrow

Implement and test the relevant boundary in more depth.

Mission 30

One owner, one next action and a review date.

Choose a stack the team can operate.

Strengthen existing PHP

Supported runtime/framework, domain objects and PHPStan.

Pilot Java where owned

Existing skills; domain types; supported Spring/JDK pair.

Use TypeScript deliberately

Strict CI plus runtime input schemas; types are erased.

Prove one boundary before a rewrite.

01

Week 1

Owner, supported target and approved contracts.

02

Week 2

Synthetic parity against the PHP baseline.

03

Week 3

Restricted read-only comparison, where approved.

04

Week 4

Evidence, rollback rehearsal and a decision.

Measure the work that actually ships.

Early 2025: 19% slower

METR: 16 experienced developers, 246 familiar-repo tasks.

2026: magnitude uncertain

METR sees reasons for optimism, with measurement and selection limits.

A good brief beats a long prompt.

01	**Context**	PHP workflow; fictional fixtures; trusted actor.
02	**Boundary**	Edit one rule. Keep the parser and tests unchanged.
03	**Outcome**	Exact match passes; malformed inputs stay denied.
04	**Evidence**	Small diff, commands, results, remaining limits.

Decide what may leave before choosing a model.

Default training input

Fictional fixtures and minimal source excerpts.

Needs an approved route

Department code and internal operational material.

Keep out of this workshop

Citizen records, database dumps and live credentials.

Make the PHP boundary explicit.

01	**Validate raw input**	Require strings and the agreed format.
02	**Preserve identity**	Compare the complete namespace with ===.
03	**Keep authority server-side**	Enforce the actor’s permitted scope.
04	**Review and test**	Types, formats, allowed matches and denials.

Release a change with an evidence packet.

01 A small diff and an independent reviewer.

02 Actual test and scanner results with scope.

03 Telemetry that avoids unnecessary citizen data.

04 An owner, a release decision and a rollback.

The tools differ. The discipline carries over.

Codex

Constrain sandbox and approvals; inspect the diff.

Claude Code

Review permissions, tools and repository instructions.

GLM-based setup

Separate the model from its client and execution tools.

Choose a model for each language task.

ASR

IndicConformer baseline; evaluate an Assamese-specific or larger challenger.

Translation

Distilled IndicTrans2 in both directions; benchmark against fuller models.

TTS

Parler first; authorized reference voices for an IndicF5 comparison.

Publish the contract and its evidence.

Identity and data

Audience, user scope, fields and resource-level policy.

Exposure

Host/method/path, owner, route approval and alternate listeners.

Operations

Version, telemetry owner, timeouts, retries and recovery.

One status tool. Clear decisions to own.

Before the call

Audience, scope, role and record permission.

During the call

Fixed schema, minimal fields and bounded tool authority.

Across conversations

Current permissions, private memory and minimal audit.