GovAI Studio

Product Design Document

AI Governance & Enterprise Implementation Platform

Web Application Specification for Claude Code Development

Matthew Carlson Consulting

February 2026 | Version 1.0

Table of Contents

Right-click and select 'Update Field' after opening in Word.

1. Product Vision and Market Opportunity

1.1 Product Definition

GovAI Studio is a web-based AI Governance and Enterprise Implementation platform that guides organizations from initial assessment through production deployment of AI systems. It replaces fragmented spreadsheets, one-off consulting documents, and manual compliance tracking with an integrated, workflow-driven application that serves every stakeholder from board-level executives to individual developers.

The application is specifically designed for the AI coding agent deployment use case (Claude Code, OpenAI Codex, and similar tools), but its governance engine, scoring system, and reporting layers are extensible to any AI system deployment including LLM applications, ML models, and autonomous agents.

1.2 Market Opportunity

The AI governance platform market is projected to grow from $227 million in 2024 to $4.83 billion by 2034 (CAGR 35.7%). Current market leaders (Credo AI, Holistic AI, Collibra, DataRobot) are priced for Fortune 500 enterprises at $100K-500K+ annual contracts and focus primarily on model governance and regulatory compliance. None provide an integrated implementation delivery workflow that covers the full journey from cold-start discovery through sandbox setup, pilot execution, and production deployment.

COMPETITIVE WHITE SPACE

Every existing AI governance platform assumes the organization has already decided to deploy AI and needs ongoing governance infrastructure. None solve the pre-deployment problem: 'We want to evaluate AI coding agents but don't know where to start, what questions to ask, or how to build a safe environment.' GovAI Studio owns the entire journey from Day Zero, making it the natural entry point that graduates into ongoing governance.

1.3 Competitive Landscape Analysis

Platform	AI Registry	Risk Scoring	Impl. Workflow	Sandbox Setup	Multi-Stakeholder Reports	Est. Price
Credo AI	Yes	Yes	No	No	Limited	$150K-400K/yr
Holistic AI	Yes	Yes	No	No	Limited	$100K-300K/yr
Collibra	Yes	Partial	No	No	Yes	$200K-500K+/yr
IBM watsonx.gov	Yes	Yes	Partial	No	Yes	$100K-400K/yr
Monitaur	Yes	Yes	No	No	Partial	$75K-200K/yr

GovAI Studio	Yes	Yes	YES	YES	YES	$500-5K/mo

1.4 Core Differentiators to Build

- **Implementation Workflow Engine:** No competitor provides guided, step-by-step implementation workflows. They assume post-deployment governance. GovAI Studio guides from discovery through deployment.
- **Sandbox Architecture Generator:** Automated generation of sandbox configuration files (Terraform, Docker Compose, devcontainer.json, managed-settings.json, requirements.toml) based on organizational infrastructure assessment answers.
- **Multi-Stakeholder Reporting:** Role-specific dashboards and exportable reports for Legal, Executive, IT, Engineering, and Marketing personas. Each sees the data relevant to their function.
- **Feasibility Scoring Engine:** Quantitative scoring system that evaluates organizational readiness across multiple dimensions and produces an actionable score, not just a pass/fail.
- **Accessible Pricing:** SaaS model at $500-5,000/month instead of $100K-500K enterprise contracts. Targets the mid-market and consulting firms that cannot afford Credo AI or Collibra.

2. Application Architecture

2.1 Technology Stack Recommendation

Layer	Technology	Rationale
Frontend Framework	Next.js 15 (App Router)	Server components reduce client bundle size. App router provides layouts, loading states, and parallel routes for dashboard complexity. Built-in API routes for BFF pattern.
UI Component Library	shadcn/ui + Tailwind CSS	Composable, accessible components. Not a dependency; components are copied into the project and fully customizable. Tailwind provides consistent design system.
State Management	Zustand + React Query (TanStack)	Zustand for client-side UI state (wizard steps, form state). React Query for server-state cache, mutations, and optimistic updates.
Database	Supabase (PostgreSQL)	Managed PostgreSQL with built-in auth, real-time subscriptions, Row Level Security, and Edge Functions. Eliminates need for separate auth provider and database hosting.
Authentication	Supabase Auth + RBAC	Built-in OAuth, magic link, and email/password auth. Custom claims for role-based access (Admin, Consultant, Executive, IT, Legal, Engineering, Marketing).

File Storage	Supabase Storage	Stores generated reports, exported documents, uploaded client artifacts. Presigned URLs for secure downloads.
PDF/DOCX Generation	React-pdf + docx-js	Server-side generation of professional deliverables. React-pdf for styled PDF reports. docx-js for Word document exports.
AI Integration	Anthropic API (Claude)	Powers intelligent questionnaire branching, automated policy drafting, risk assessment narrative generation, and the AI assistant within the platform.
Timeline / Gantt	Custom (React + DnD Kit)	Build a lightweight Gantt component using DnD Kit for drag-and-drop task reordering, date adjustment, and dependency visualization. Avoids heavy Gantt library lock-in.
Hosting	Vercel	Native Next.js hosting with edge functions, automatic preview deployments, and analytics. Integrates with Supabase for full-stack deployment.

2.2 Database Schema Overview

The following represents the core data model. All tables include standard audit fields (created_at, updated_at, created_by) and leverage Supabase Row Level Security for multi-tenant data isolation.

```
-- Core Entities
organizations           -- Client organizations (multi-tenant root)
projects                -- Governance engagement projects per org
users                   -- All users with role assignments
team_members            -- Project-scoped team assignments

-- Assessment Engine
assessment_templates    -- Configurable questionnaire templates
assessment_questions    -- Questions with branching logic, weights
assessment_responses    -- User responses per project
feasibility_scores      -- Computed scores per dimension

-- Governance Artifacts
policies                -- AUP, IRP addendum, data classification docs
policy_versions         -- Version history with diffs
compliance_mappings     -- Control-to-framework mappings (SOC2, HIPAA...)
risk_classifications    -- Risk tier definitions (High/Medium/Low)
gate_reviews            -- Three-gate approval records

-- Implementation Tracking
workflow_templates      -- Reusable implementation workflow definitions
workflow_phases         -- Phase definitions within a workflow
workflow_tasks          -- Individual tasks with dependencies
task_assignments        -- Who is responsible for each task
task_status_history     -- Status changes with timestamps
```

```
-- Sandbox Configuration
sandbox_configs         -- Infrastructure configuration per project
config_files            -- Generated config files (JSON, TOML, YAML)
environment_validations -- Sandbox health check results

-- PoC Evaluation
poc_projects            -- Proof-of-concept definitions
poc_sprints             -- Sprint evaluation windows
poc_metrics             -- Captured metrics per sprint
tool_evaluations        -- Claude Code vs Codex comparative data

-- Reporting
report_templates        -- Report format definitions per persona
generated_reports       -- Report generation history + file references

-- Timeline / Project Management
timeline_milestones     -- Major milestone definitions
timeline_dependencies   -- Task-to-task dependency links
timeline_snapshots      -- Point-in-time schedule captures for comparison
```

2.3 Application Route Structure

```
app/
  (auth)/
    login/              -- Email/password + OAuth login
    register/           -- New user registration
    forgot-password/    -- Password recovery
  (dashboard)/
    layout.tsx          -- Sidebar nav, role-based menu rendering
    page.tsx            -- Dashboard home (project overview cards)
    projects/
      new/              -- New project wizard
      [id]/
        overview/       -- Project summary, health score, status
        discovery/
          questionnaire/-- Guided assessment questionnaire
          readiness/    -- Readiness Assessment dashboard
          prerequisites/-- Checklist tracker with assignments
        governance/
          policies/     -- Policy editor (AUP, IRP, data classification)
          gates/        -- Three-gate review board
          compliance/   -- Compliance framework mapping
          risk/         -- Risk classification manager
        sandbox/
          configure/    -- Infrastructure questionnaire + config gen
          files/        -- Generated config file viewer/editor
          validate/     -- Sandbox health check results
        poc/
          projects/     -- PoC definition and selection scoring
```

```
        sprints/      -- Sprint evaluation tracker
        compare/      -- Tool comparison dashboard
        metrics/      -- Baseline vs. AI-assisted metrics
      timeline/
        gantt/        -- Interactive Gantt chart (drag/drop)
        milestones/   -- Milestone tracker
        snapshots/    -- Schedule baseline comparisons
      reports/
        generate/     -- Report builder (select persona + sections)
        history/      -- Previously generated reports
      team/           -- Team member management + roles
    settings/         -- Org settings, billing, integrations
  api/
    assessments/      -- Assessment CRUD + scoring engine
    reports/          -- Report generation endpoints
    configs/          -- Sandbox config generation
    ai/               -- Claude API integration endpoints
    export/           -- DOCX/PDF export endpoints
```

3. Core Feature Specifications

3.1 Discovery and Assessment Engine

The Discovery Engine is the entry point for every new engagement. It replaces the manual discovery meeting with a structured, branching questionnaire that adapts based on responses. The engine collects everything needed to produce the Readiness Assessment and determine the project plan.

3.1.1 Questionnaire Architecture

Each questionnaire is built from a template of sections, each containing conditional questions. Questions have types (single-select, multi-select, text, number, file-upload), weights for scoring, and branching rules that show/hide follow-up questions based on previous answers.

```
// Question schema example
{
  id: "infra_cloud_provider",
  section: "infrastructure",
  text: "Which cloud provider(s) does the organization use?",
  type: "multi_select",
  options: ["AWS", "Google Cloud", "Azure", "On-premises only", "Hybrid"],
  weight: 3,
  scoring: {
    "AWS": 5, "Google Cloud": 5, "Azure": 5,
    "Hybrid": 4, "On-premises only": 2
  },
  branches: {
    "AWS": ["infra_aws_accounts", "infra_aws_bedrock"],
    "Google Cloud": ["infra_gcp_projects", "infra_gcp_vertex"],
    "Azure": ["infra_azure_subscriptions"],
    "On-premises only": ["infra_onprem_virtualization"]
  },
```

```
  help_text: "Select all that apply. This determines sandbox architecture.",
  required: true
}
```

3.1.2 Core Discovery Question Domains

The questionnaire covers five domains, each contributing to the feasibility score. Below are the key questions per domain that drive critical implementation decisions.

DOMAIN 1: Infrastructure Readiness (Weight: 25%)

- Cloud provider(s) and account/project structure (determines sandbox model)
- Network segmentation maturity (VLANs, VPCs, micro-segmentation)
- Proxy infrastructure and TLS inspection (determines API connectivity path)
- Developer environment standards (IDE, local vs. cloud workstations)
- Package management and artifact registries (npm, PyPI, Artifactory)
- Existing IaC practices (Terraform, CloudFormation, Pulumi, none)

DOMAIN 2: Security Posture (Weight: 25%)

- Data classification policy maturity (formal/informal/none)
- DLP tooling and coverage (endpoint, network, cloud)
- SIEM platform and log ingestion capacity
- Secrets management solution maturity
- Existing AI/ML security policies
- Incident response plan maturity and last test date

DOMAIN 3: Governance Maturity (Weight: 20%)

- Existing AI acceptable use policy (yes/no/in-draft)
- Compliance framework coverage (SOC 2, HIPAA, PCI-DSS, GDPR, etc.)
- Vendor risk assessment process maturity
- Change management and approval workflow formality
- Previous AI initiative history and outcomes

DOMAIN 4: Engineering Culture (Weight: 15%)

- CI/CD pipeline maturity and SAST/DAST integration
- Code review discipline and PR approval requirements
- Testing culture (unit test coverage baseline, test automation)
- Developer openness to new tooling (survey or proxy indicators)
- Team size and structure (feature teams, platform teams)

DOMAIN 5: Business Alignment (Weight: 15%)

- Executive sponsor identification and commitment level
- Quantified business outcome expectations
- Budget authorization status
- Timeline constraints and decision deadlines

- Risk appetite for AI adoption (conservative/moderate/aggressive)

3.2 Feasibility Scoring Engine

The scoring engine transforms questionnaire responses into a quantitative feasibility assessment. Every question carries a weight and a score mapping. The engine computes per-domain scores, an overall feasibility score, and generates actionable recommendations based on score gaps.

3.2.1 Scoring Model

Domain	Weight	Score Range	Pass Threshold	Remediation Trigger
Infrastructure Readiness	25%	0-100	60+	Below 60: flag infra gaps, recommend Model B or C sandbox
Security Posture	25%	0-100	60+	Below 60: security hardening phase added to timeline
Governance Maturity	20%	0-100	50+	Below 50: governance foundation phase extended by 1-2 weeks
Engineering Culture	15%	0-100	50+	Below 50: additional training and onboarding time in timeline
Business Alignment	15%	0-100	50+	Below 50: executive alignment workshop added to Phase 1

3.2.2 Overall Score Interpretation

Score Range	Rating	Recommendation
80-100	High Feasibility	Organization is well-positioned for rapid AI deployment. Accelerated 45-day timeline recommended. Minimal remediation required.
60-79	Moderate Feasibility	Organization can proceed with targeted remediation. Standard 60-day timeline. 1-3 domain-specific gaps require attention.
40-59	Conditional Feasibility	Significant gaps in 2+ domains. Extended 90-day timeline with pre-work phase. Recommend addressing infrastructure and security gaps before pilot.
Below 40	Not Ready	Organization requires foundational work before AI deployment is advisable. Recommend a governance and infrastructure readiness engagement before proceeding.

3.2.3 Score Visualization

The feasibility score is displayed as an interactive radar chart (5-axis spider chart) showing each domain score relative to the pass threshold. A companion bar chart shows the overall weighted score with color-coded ranges (green/yellow/orange/red). Each domain score is clickable, expanding to show individual question scores and specific gap recommendations.

3.3 Implementation Workflow Engine

The workflow engine is the operational backbone of GovAI Studio. It transforms the assessment results into a customized project plan with tasks, assignments, dependencies, and milestones. The engine dynamically adjusts the plan based on feasibility scores, adding remediation tasks where gaps exist.

3.3.1 Workflow Template Structure

Each workflow template defines phases, tasks within phases, default durations, dependency chains, and role assignments. The system ships with a default 60-day governance project template that automatically adjusts based on assessment results.

Phase	Key Tasks	Default Duration	Owner Role	Gate
1. Discovery	Complete assessment questionnaire, collect documents, identify pilot team, submit vendor terms to Legal	5 days	Consultant	Readiness Assessment delivered
2. Governance	Draft AUP, define risk tiers, establish gate model, draft IRP addendum, RACI matrix, legal sign-off	7 days	Consultant + Legal	AUP approved, Legal sign-off obtained
3. Sandbox Build	Provision infrastructure, configure egress, deploy tools, integrate logging, create test repos, validate isolation	8 days	Consultant + IT	Sandbox passes isolation validation
4. Training & Launch	Security training, collect AUP signatures, baseline metrics, define success criteria, Sprint 1 launch	7 days	Consultant + Engineering	Sprint 1 Go/No-Go
5. Evaluation	Sprints 2-3, expand developers, CI/CD integration, red team exercise, comparative evaluation	23 days	Engineering	Sprint 3 complete, all data collected
6. Readout	Calculate ROI, compile evaluation, prepare briefing, present to leadership, vendor negotiation support	15 days	Consultant + Executive	Executive decision

3.4 Project Timeline and Management Interface

The timeline component provides an intuitive, drag-and-drop Gantt chart view that is accessible to non-technical users. It is the primary interface for project managers, executive sponsors, and stakeholders to track progress.

3.4.1 Gantt Chart Specifications

- **Drag-and-drop task bars:** Users can click and drag task bars to adjust start/end dates. Connected dependencies update automatically with conflict warnings if a date change violates a dependency chain.
- **Zoom levels:** Day, Week, Month, and Quarter views. The default view auto-selects based on project duration: 60-day projects default to Week view.
- **Dependency arrows:** Finish-to-Start (FS), Start-to-Start (SS), Finish-to-Finish (FF), and Start-to-Finish (SF) dependency types visualized as connecting arrows between task bars.
- **Critical path highlighting:** The longest dependency chain is automatically highlighted in red, showing which tasks will delay the project if they slip.
- **Milestone markers:** Diamond-shaped markers on the timeline for gate reviews and key deliverables. Milestones can be linked to gate review approvals.
- **Baseline comparison:** Users can save a baseline snapshot and overlay current vs. baseline schedules to visualize schedule drift.
- **Resource view:** Toggle to a resource-centric view showing each team member's workload across the timeline, highlighting over-allocation.
- **Export:** Export timeline as PDF, PNG, or CSV (for import into MS Project, Jira, or Monday.com).

3.4.2 Task Detail Panel

Clicking any task opens a detail panel with: task description, assigned owner, status (Not Started, In Progress, Blocked, Complete), due date, dependencies (upstream and downstream), attached documents, comments thread, and a status history log.

3.4.3 Implementation Approach

```
// Gantt chart built with:
// 1. @dnd-kit/core + @dnd-kit/sortable for drag-and-drop
// 2. date-fns for date calculations
// 3. Custom SVG renderer for dependency arrows
// 4. Zustand store for task state management
// 5. Supabase real-time subscriptions for live updates

// Key data structure:
interface TimelineTask {
  id: string;
  title: string;
  phase: string;
  start_date: Date;
  end_date: Date;
  duration_days: number;
  assigned_to: string;
  status: 'not_started' | 'in_progress' | 'blocked' | 'complete';
```

```
  dependencies: { task_id: string; type: 'FS' | 'SS' | 'FF' | 'SF' }[];
  progress_percent: number;
  is_milestone: boolean;
  gate_review_id?: string;
}
```

3.5 Multi-Stakeholder Reporting System

Each enterprise stakeholder needs different information, at different detail levels, in different language. The reporting system generates role-specific reports from the same underlying project data.

3.5.1 Report Personas and Content

Persona	Report Contents	Language Style	Format
Executive / Board	Feasibility score summary, ROI projections, risk heat map, competitive positioning, timeline milestones, budget vs. actual, strategic recommendations, go/no-go recommendation	Business outcome focused. No technical jargon. Data-driven with visualizations. 3-5 pages maximum.	PDF with charts + optional PPTX slide deck
Legal / Compliance	Vendor contract analysis, data processing terms, training exclusion clauses, liability/indemnification review, compliance control mapping (SOC 2, HIPAA, etc.), regulatory risk assessment, AUP review status, incident response provisions	Precise legal language. Clause-by-clause analysis. Regulatory framework references. Thorough.	DOCX (editable) with tracked changes capability
IT / Security	Sandbox architecture diagram, network security configuration, egress rules, DLP rules, SIEM integration plan, managed-settings.json details, requirements.toml details, proxy configuration, encryption standards, vulnerability assessment	Technical precision. Configuration-level detail. Network diagrams. Security control specifics.	PDF with architecture diagrams + config file attachments
Engineering / Dev	Tool comparison results (Claude Code vs. Codex), developer satisfaction scores, productivity metrics (velocity, cycle time), code quality impact, CI/CD integration guide, prompt playbook, CLAUDE.md/AGENTS.md templates, setup instructions	Developer-friendly. Code examples. Practical how-to. Metrics-driven.	Markdown (for repo inclusion) + PDF

Marketing / Comms	AI initiative messaging guide, internal communications templates, FAQ for employee concerns, external positioning (if applicable), change management narrative, success metrics for public reporting	Narrative-driven. Employee-facing language. Addresses fears. Positive framing.	DOCX (editable) for comms team customization

3.5.2 AI-Assisted Report Generation

The report generator uses Claude to transform raw project data into polished narrative content. The AI receives structured data (scores, metrics, configuration details) and generates contextually appropriate prose for each persona. The user reviews and edits the AI-generated content before finalizing. Reports are never auto-published; a human always reviews the final output.

3.6 Sandbox Configuration Generator

Based on assessment responses, the platform generates production-ready configuration files for the recommended sandbox architecture. This is the feature that most directly saves implementation time and reduces configuration errors.

3.6.1 Generated Artifacts

Artifact	Format	Generated Based On
Claude Code managed-settings.json	JSON	Data classification answers, file pattern deny rules, network domain allowlist, MCP server policy
Codex requirements.toml	TOML	Security posture answers, sandbox mode selection, approval policy, web search policy
CLAUDE.md project instructions	Markdown	PoC scope definition, technology stack, authorized/unauthorized operations, quality standards
AGENTS.md instructions	Markdown	Same as CLAUDE.md, formatted for Codex agent instruction format
Terraform / CloudFormation module	HCL / YAML	Cloud provider selection, VPC CIDR, subnet configuration, security groups, VPC endpoints
Docker Compose sandbox	YAML	Selected if on-premises or local sandbox model. Pre-configured dev environment.
devcontainer.json	JSON	Selected if Codespaces model. IDE extensions, tool versions, secrets configuration.
LLM proxy config	YAML	LiteLLM or TrueFoundry proxy configuration with model routing and audit logging.
DLP rule set	JSON/CSV	Pattern definitions for secrets, PII, and proprietary code detection in AI API traffic.

Acceptable Use Policy draft	DOCX	Organizational profile, data classification answers, compliance requirements, tool inventory.

3.7 Three-Gate Review System

The gate review system digitalizes the approval workflow for progressive AI deployment access. Each gate has defined criteria, required approvers, evidence requirements, and automated status tracking.

- **Gate 1 (Sandbox Access):** Automated approval. Triggered when developer completes training module and signs AUP digitally. System provisions sandbox access immediately upon both conditions being met.
- **Gate 2 (Pilot Deployment):** Requires manual approval from Security lead and Engineering lead. Evidence checklist: managed settings deployed, CI/CD SAST confirmed, DLP rules active, audit logging confirmed. 3-5 day SLA with escalation.
- **Gate 3 (Production Path):** Full review board. Required approvers: CISO, CTO/VP Engineering, Legal, Executive Sponsor. Evidence: complete pilot evaluation data, security incident log (zero incidents required), compliance mapping, risk assessment, monitoring plan. 2-4 week review cycle.

4. Extensibility Architecture

GovAI Studio is designed for extensibility at every layer, allowing the platform to grow from AI coding agent governance into a comprehensive enterprise AI governance solution.

4.1 Plugin System

- **Assessment Plugins:** New questionnaire domains can be added as plugins. Each plugin defines questions, scoring rules, and remediation recommendations. Example future plugins: ML Model Governance, LLM Application Governance, Autonomous Agent Governance, Data Pipeline Governance.
- **Report Template Plugins:** Custom report templates can be added for industry-specific requirements (healthcare, financial services, government, manufacturing).
- **Config Generator Plugins:** New sandbox architecture patterns can be added. Example: Azure-native sandbox, GCP-native sandbox, air-gapped environment, FedRAMP sandbox.
- **Compliance Framework Plugins:** New regulatory frameworks can be added with their control mappings. Each plugin defines controls, evidence requirements, and assessment criteria. Example: CMMC, FedRAMP, SOX, EU AI Act risk classification.
- **Integration Plugins:** Third-party integrations for data push/pull. Priority integrations: Jira (timeline sync), Slack (notifications), Microsoft Teams (notifications), GitHub/GitLab (sandbox repo provisioning), Terraform Cloud (infrastructure deployment).

4.2 API-First Design

Every feature in GovAI Studio is accessible through a REST API, enabling integration with existing enterprise tools. The API supports JSON responses, pagination, filtering, and webhook callbacks for event-driven integrations. API documentation is auto-generated using OpenAPI 3.1 spec.

5. Development Implementation Plan

5.1 Phase 1: Foundation (Weeks 1-3)

Goal: Functional authentication, database schema, project CRUD, and basic navigation.

1. **Day 1-2:** Initialize Next.js 15 project with App Router, Tailwind CSS, shadcn/ui. Set up Supabase project with authentication. Configure Vercel deployment pipeline. Set up GitHub repository with branch protection.
2. **Day 3-5:** Database schema migration: organizations, projects, users, team_members tables. Supabase Row Level Security policies. Supabase Auth configuration (email, OAuth providers).
3. **Day 6-8:** Authentication flows: login, registration, password recovery. Protected route middleware. Role-based sidebar navigation. Organization and project CRUD.
4. **Day 9-12:** Dashboard layout: project cards, status indicators, team member display. Settings page: organization profile, user profile, role management.
5. **Day 13-15:** Assessment engine database schema: templates, questions, responses, scores. Assessment questionnaire UI: multi-step wizard with conditional rendering.

5.2 Phase 2: Core Engine (Weeks 4-6)

Goal: Working assessment engine, feasibility scoring, and sandbox config generation.

1. **Day 16-20:** Complete questionnaire implementation for all five domains. Branching logic engine. Response persistence. Progress tracking.
2. **Day 21-24:** Feasibility scoring engine: per-domain calculation, weighted aggregate, threshold evaluation. Radar chart visualization (Recharts). Score detail drill-down.
3. **Day 25-28:** Sandbox configuration generator: template engine for JSON/TOML/YAML/HCL generation. Cloud provider-specific templates. Config file preview and editing interface.
4. **Day 29-32:** Policy document generator: AUP template engine with organizational context injection. DOCX export using docx-js. Policy version management.

5.3 Phase 3: Workflow and Timeline (Weeks 7-9)

Goal: Working project timeline, task management, and gate review system.

1. **Day 33-38:** Timeline data model: phases, tasks, milestones, dependencies. Workflow template engine that generates customized plans from assessment results.
2. **Day 39-44:** Gantt chart component: SVG rendering, drag-and-drop (DnD Kit), dependency arrows, zoom levels, critical path calculation. Task detail panel.

3. **Day 45-48:** Gate review system: review request creation, evidence checklist, approver routing, status tracking, automated Gate 1 approval logic.

5.4 Phase 4: Reporting and AI Integration (Weeks 10-12)

Goal: Multi-stakeholder reports, AI-assisted content generation, and Polish.

1. **Day 49-54:** Report template system for all five personas. Data aggregation pipelines that pull from assessment, timeline, PoC, and governance data. PDF generation with React-pdf.
2. **Day 55-58:** Claude API integration: AI-assisted report narrative generation, intelligent questionnaire recommendations, policy draft assistance. Prompt engineering and output validation.
3. **Day 59-63:** PoC tracking module: sprint evaluation forms, metrics capture, tool comparison dashboard, baseline vs. AI-assisted charting.
4. **Day 64-68:** Compliance mapping interface: framework-to-control mapping editor, evidence linking, audit trail. Pre-built mappings for SOC 2, HIPAA, NIST 800-53.
5. **Day 69-72:** Testing, bug fixes, performance optimization, accessibility audit, documentation.

5.5 Phase 5: Launch Preparation (Weeks 13-14)

1. **Day 73-78:** End-to-end testing with real engagement data. User acceptance testing with pilot customers. Security review of authentication and data isolation.
2. **Day 79-84:** Landing page, pricing page, documentation site. Stripe integration for billing. Onboarding flow for new users.

6. Competitor Feature Integration Roadmap

The following features are drawn from competitive analysis of Credo AI, Holistic AI, Collibra, IBM watsonx.governance, Monitaur, DataRobot AI Governance, Fiddler AI, and OvalEdge. These represent the most impactful capabilities that should be incorporated into GovAI Studio post-MVP.

6.1 Priority Features from Competitors

Feature	Source Competitor	Priority	Effort	Target Release
AI System Registry / Inventory	Credo AI, Collibra	HIGH (v1.1)	3-4 weeks	Month 5
Policy Pack Library (EU AI Act, NIST RMF, ISO 42001)	Credo AI	HIGH (v1.1)	4-5 weeks	Month 5
Shadow AI Detection Alerts	Reco, CloudEagle	HIGH (v1.2)	3 weeks	Month 6
Model Card / AI Card Generation	Credo AI, Monitaur	MEDIUM (v1.2)	2-3 weeks	Month 6

Automated Compliance Evidence Collection	OvalEdge, Holistic AI	MEDIUM (v1.3)	4 weeks	Month 7
Real-Time AI Usage Monitoring Dashboard	Fiddler AI, Arthur AI	MEDIUM (v1.3)	5-6 weeks	Month 8
Vendor Risk Assessment Automation	Credo AI, Holistic AI	MEDIUM (v1.4)	3 weeks	Month 8
Bias/Fairness Assessment Suite	Holistic AI, Arthur AI	LOW (v2.0)	6-8 weeks	Month 10+
Data Lineage Tracking	Collibra, Atlan	LOW (v2.0)	8+ weeks	Month 12+

7. Claude Code Development Specification

This section defines how to use Claude Code as your primary development tool for building GovAI Studio. Place the following CLAUDE.md file in your project root.

7.1 CLAUDE.md for GovAI Studio Development

```
# CLAUDE.md - GovAI Studio Development Instructions

## Project Overview
GovAI Studio is a Next.js 15 (App Router) web application for AI governance
and enterprise implementation management. It uses Supabase for database,
auth, and storage, with shadcn/ui for components and Tailwind CSS for styling.

## Technology Stack (DO NOT CHANGE)
- Framework: Next.js 15 (App Router, Server Components)
- Language: TypeScript (strict mode)
- UI: shadcn/ui components + Tailwind CSS 4
- Database: Supabase (PostgreSQL with RLS)
- Auth: Supabase Auth with RBAC custom claims
- State: Zustand (client) + TanStack Query (server)
- Charts: Recharts
- DnD: @dnd-kit/core + @dnd-kit/sortable
- PDF: @react-pdf/renderer
- DOCX: docx (docx-js)
- Deployment: Vercel

## Architecture Rules
1. Use Server Components by default. Add 'use client' only when needed.
2. All database access through Supabase client (server-side) or API routes.
3. Every table has RLS policies. Never bypass RLS.
4. Use Zod for all form validation and API input validation.
5. Error handling: use error.tsx boundaries per route segment.
```

```
6. Loading states: use loading.tsx per route segment.
7. All API routes in app/api/ return typed JSON responses.

## File Organization
src/
  app/              -- Next.js App Router pages and layouts
  components/
    ui/             -- shadcn/ui base components (DO NOT EDIT)
    shared/         -- Shared components used across features
    features/       -- Feature-specific components
      assessment/   -- Questionnaire, scoring, readiness views
      governance/   -- Policy editor, gate reviews, compliance
      sandbox/      -- Config generator, file viewer
      timeline/     -- Gantt chart, milestones, task management
      reports/      -- Report builder, persona views
      poc/          -- PoC tracking, sprint eval, comparison
  lib/
    supabase/       -- Supabase client, server client, types
    scoring/        -- Feasibility scoring engine
    config-gen/     -- Sandbox config file generators
    report-gen/     -- Report generation logic
    ai/             -- Claude API integration
    utils/          -- Shared utilities
  types/            -- TypeScript type definitions
  stores/           -- Zustand stores
  hooks/            -- Custom React hooks

## Coding Standards
- Named exports only (no default exports except pages)
- Functional components with explicit return types
- Use 'satisfies' for type-safe object literals
- Prefer server actions for mutations over API routes
- All user-facing text must be extractable for i18n
- Minimum test coverage: utility functions and scoring engine

## Database Conventions
- Table names: snake_case, plural (e.g., assessment_responses)
- Column names: snake_case
- All tables include: id (uuid), created_at, updated_at
- Foreign keys: [referenced_table]_id
- Soft deletes: deleted_at column (nullable timestamp)
- Enums: stored as text with CHECK constraints

## Security Rules
- NEVER store API keys in code or environment variables in client code
- All AI API calls go through server-side API routes only
- Input sanitization on all user inputs (DOMPurify for rich text)
- Rate limiting on API routes (use Vercel's built-in or custom)
- CSP headers configured in next.config.js

## Current Sprint
[Update this section each sprint with current objectives]
```

7.2 Initial Claude Code Commands

Use the following sequence to bootstrap the project with Claude Code:

```
# Step 1: Project initialization
claude "Initialize a new Next.js 15 project with TypeScript, Tailwind CSS 4,
  and shadcn/ui. Use App Router. Set up the project structure defined in
  CLAUDE.md. Install all dependencies from the technology stack. Create the
  folder structure for components, lib, types, stores, and hooks."

# Step 2: Supabase setup
claude "Create a Supabase migration for the core tables: organizations,
  projects, users, team_members. Include RLS policies that ensure users
  can only access data within their organization. Create the Supabase
  client utilities for both server and client components."

# Step 3: Auth flow
claude "Build the authentication pages: login (email + Google OAuth),
  registration with organization creation, password recovery. Create
  middleware that protects all routes under (dashboard). Implement
  role-based access using Supabase custom claims."

# Step 4: Dashboard shell
claude "Build the dashboard layout with a collapsible sidebar navigation.
  Navigation items should render based on user role. Include: Dashboard
  home, Projects list, Settings. The home page shows project cards with
  status, feasibility score, and last activity date."

# Step 5: Assessment engine
claude "Create the assessment questionnaire system. Build a multi-step
  wizard component that renders questions from a JSON template. Implement
  conditional branching (show/hide questions based on previous answers),
  progress tracking, and response persistence to Supabase. Start with
  the Infrastructure Readiness domain questions."
```

8. Pricing and Monetization Strategy

Tier	Starter ($500/mo)	Professional ($2,000/mo)	Enterprise ($5,000+/mo)
Users	Up to 5	Up to 25	Unlimited
Projects	1 active	5 active	Unlimited
Assessment Engine	Standard domains	All domains + custom	All + custom plugins
Reports	Executive + IT only	All 5 personas	All + custom templates

Config Generator	Basic configs	All architectures	All + custom templates
AI Assistance	Limited (50 calls/mo)	Standard (500 calls/mo)	Unlimited
Timeline/Gantt	View only	Full edit + export	Full + API sync
Compliance Mappings	SOC 2 only	SOC 2, HIPAA, GDPR	All + custom frameworks
Integrations	Export only	Jira, Slack	All + API access
Support	Community + docs	Email (24hr SLA)	Dedicated CSM

CONSULTING UPSELL MODEL

GovAI Studio is the product; consulting services are the high-margin upsell. The platform generates work that requires expert execution: sandbox architecture implementation, security review, executive presentations, vendor negotiations. Position the platform at accessible SaaS pricing to generate consulting leads. Target: 30% of platform customers convert to consulting engagements at $10K-50K per engagement.

9. Success Metrics

- **Platform Adoption:** 100 registered organizations within 6 months of launch. 25 active paying customers by month 6.
- **Engagement Conversion:** 30% of platform users engage consulting services within 90 days.
- **Time-to-Value:** Users complete assessment and receive feasibility score within 60 minutes of first login. Full sandbox configuration generated within 4 hours.
- **Retention:** 80%+ monthly active user retention. Net Promoter Score 40+.
- **Revenue:** $15K MRR by month 6. $50K MRR by month 12 (platform + consulting combined).