Project Title: CompliAI
Tagline: *AI-Powered Compliance, From Policy to Audit*
Version: 1.0 MVP
Date: 30 June 2025

1. Introduction

1.1 Purpose

CompliAI is an agentic AI assistant designed to streamline governance, risk, and compliance (GRC) tasks. It allows users to ask compliance-related questions, map internal controls to frameworks like ISO 27001 and SOC 2, generate audit-ready policies, and prepare for audits with agent-powered workflows.

1.2 Scope

The system will be delivered as a SaaS platform where GRC professionals can interact with an AI assistant to:

- Understand clauses in compliance frameworks
- Upload and analyze documents
- Generate and manage policies
- Track compliance readiness

2. Overall Description

2.1 Product Functions

- Chat-based interface for clause-level questions
- Policy upload and parsing with semantic control mapping
- Policy generator and rewriter
- Agent-based planning for ISO/SOC2 audits
- Downloadable/exportable reports

2.2 User Classes

Role	Description
GRC Manager	Uses the platform to manage compliance
CISO / Security Lead	Tracks readiness and gaps
Consultant	Uses platform for client compliance prep
Admin	Manages users, access, and integrations

3. Functional Requirements

FR1: Conversational Clause Assistant

- Chat interface using LLM
- Support question answering for ISO 27001, NIST CSF, PCI-DSS, SOC 2
- Return control details, context, and use cases

FR2: Upload and Parse Policy Documents

- Allow `.docx`, `.pdf`, or `.txt` uploads
- Extract and embed content into vector DB
- Identify and highlight relevant controls

FR3: Internal Control Mapping

- Users can input controls manually or via form
- AI maps controls to standard clauses
- Confidence level and rationale shown

FR4: Policy Generator & Editor

- Generate policies based on selected frameworks
- Customize tone and format
- Option to reword uploaded policies

FR5: Agentic Compliance Planner

- For a selected framework (e.g., ISO 27001), AI creates:
    - Control checklist
    - Missing controls report
    - Suggested next steps
- Multi-step flow using LangGraph/Autogen

FR6: Export & Reporting

- Export generated policies, mapping reports as `.docx`, `.pdf`
- Export audit-readiness checklist

FR7: User Management

- Role-based access control (Admin, Reviewer, Viewer)
- OAuth or email/password login

4. Non-Functional Requirements

NFR Code	Requirement
NFR-1	Responses under 3 seconds for chat interactions
NFR-2	Uploaded docs stored encrypted at rest
NFR-3	System uptime 99.9%
NFR-4	Scalable to 500 users (MVP)
NFR-5	GDPR & SOC2-compliant infrastructure

5. System Architecture

5.1 Tech Stack

Layer	Tool
Frontend	Next.js, Tailwind CSS, shadcn/ui
Backend	FastAPI / NestJS
LLM Agent	OpenAI GPT-4 / Claude + LangChain or LangGraph
Vector DB	Pinecone / Weaviate
Auth	Firebase Auth / Clerk.dev
DB	Firestore / MongoDB Atlas
File Storage	AWS S3 / Firebase Storage
Deployment	Vercel (frontend), Railway/Render/AWS (backend)

6. UI Mockups to Build

- Dashboard
- Chat Interface
- Policy Upload & Mapping Viewer
- Policy Generator Form

- Audit Readiness Checklist
- User Roles/Settings Panel

7. Development Plan

🔹 Week-by-Week Roadmap (8-Week MVP Plan)

Week	Goals
Week 1	Set up frontend + auth (login/signup + RBAC)
Week 2	Integrate OpenAI API + basic chat interface
Week 3	Add policy upload & embedding with Pinecone
Week 4	Build control mapping feature + clause search
Week 5	Add policy generator & rewriter UI
Week 6	Build audit readiness planner agent
Week 7	PDF/Markdown export + reporting
Week 8	Testing, final polish, deploy to Vercel & launch beta

8. Future Roadmap (Post-MVP)

- Slack/Teams integration
- SOC 2 and ISO 42001 auto-audit planner
- Evidence collection tool
- AI-based vendor risk analysis
- Public API for integrations (e.g., Jira, GitHub)