| Project assets | | |
| --- | --- | --- |
| Files File | No type Description | Drop-downs Status |
| File |  | Not started |
| File |  | In progress |
| File |  | Under review |
| File |  | Approved |

Okay, here's a compact, bullet-point overview of the ISA project's roadmap, based on our comprehensive docs/blueprint.md which reflects the "Ultimate Vision":

**Intelligent Standards Assistant (ISA) - Compact Roadmap**

* **Overall Goal:** Develop ISA into an "Albert Einstein of GS1 standards development"—a highly intelligent, verifiable, and proactive AI assistant.
* **Phase 1: Foundational Strengthening & Core Capability Enhancement (COMPLETED)**
  + Optimized Firebase App Hosting configuration.
  + Hardened Firestore security rules (default deny-all).
  + Implemented robust secrets management (.gitignore, .env placeholders).
  + Outlined CI/CD (GitHub Actions for App Hosting) & basic monitoring.
  + Refined AI flow error handling comprehensively (no output!, schema-compliant errors).
  + Matured core RAG pipeline:
    - Structured input for Q&A (documentChunks).
    - AI-generated source citations & reasoning steps (Q&A, Error Detection).
  + Implemented core AI features: Document Q&A, Standards Analysis, Error Detection, NL to Formal, Independent Research.
  + Enhanced webSearch tool (mock with structured output).
  + Conceptual RAG/KG tooling:
    - generateDocumentEmbeddings flow (uses ai.embed(), assigns chunkId).
    - queryVectorStoreTool (mock, simulates Vertex AI call & metadata fetch by chunkId).
    - answerGs1QuestionsWithVectorSearch flow & UI (explicit RAG pipeline: embed->search->synthesize).
    - queryKnowledgeGraphTool (mock) & demonstrateKgQuery flow & UI.
    - Conceptual design for KG-Augmented RAG.
  + "Interactive Identifier Validator" feature: UI & AI flow with refined mock rules & initial symbolic checks.
  + Centralized Zod schemas (src/ai/schemas.ts).
  + UI polish (placeholder images, improved error display, KG demo rendering, retrievedChunksCount in vector search UI).
  + Comprehensive docs/blueprint.md updated to "Ultimate Vision" and reflecting all Phase 1 work.
  + Developer experience enhancements (.vscode/settings.json, README.md, code hygiene).
  + Firebase Emulator usage strategy clarified in blueprint.
  + **processDocumentForRAG flow designed, with Document AI SDK call structure implemented.**
* **Phase 2: Infrastructure Maturation & Advanced Feature Integration (ACTIVE)**
  + **Phase 2A: Live RAG & Basic KG Implementation (Next 3-6 Months)**
    - **Implement ETLVRE Pipeline v1:**
      * Setup GCS for source docs.
      * Develop ETLVRE flow (processDocumentForRAG) using live Document AI parsing, semantic chunking, generateDocumentEmbeddings flow, and prepare data for Vertex AI Vector Search & Firestore (metadata).
      * Provision Vertex AI Vector Search & Firestore; integrate queryVectorStoreTool with live Vertex AI & Firestore metadata lookup.
      * Transition /advanced/qa-vector-search to use this live RAG pipeline.
    - **Establish Basic Knowledge Graph (KG v1):**
      * Define KG v1 schema (core GS1 entities/relationships).
      * Provision KG storage (AlloyDB AI / Spanner Graph).
      * Populate KG v1 (scripts, LLM-assisted extraction from ETLVRE output).
      * Replace mock queryKnowledgeGraphTool with real Genkit tool for KG v1.
      * Test /advanced/kg-query-demo with live KG.
    - **Initial KG-RAG Integration:** Pilot flow using live KG to augment live Vector Search queries.
    - **Enhance "Interactive Identifier Validator":** More symbolic rules, integrate with KG v1 for rule/constraint lookups.
    - **MLOps v1:** Vertex AI Pipeline to automate RAG index/metadata updates.
  + **Phase 2B: Early NeSy & Deeper KG-RAG (6-18 Months)**
    - Expand ETLVRE Pipeline (v2) & KG (v2) (broader docs, advanced parsing, more KG rules/relationships, versioning).
    - Develop Early Neuro-Symbolic (NeSy) Components (formalize GS1 rules, symbolic reasoner, Genkit tools for it, pilot NeSy for LLM output verification).
    - Advanced KG-RAG & Context Fusion (multi-hop KG queries, RRF).
    - Initial Multi-modal RAG/Ingestion (tables, diagrams from GS1 docs using Document AI/Gemini).
    - Develop "Automated Standard Impact Analyzer" (Pilot).
    - MLOps v2 (KG update pipelines, RAG/KG monitoring).
* **Phase 3: Scalable Vision, Ultimate AI & Future-Proofing (1.5–3+ Years)**
  + Globally Scalable Architecture (Serverless-first, API Gateway, global distribution).
  + Mature NeSy, Causal AI, GS1-RLAIF, Predictive Capabilities (concept forecasting, verifiable standard generation - simple cases).
  + Full Multi-Modal Integration (understanding & generation).
  + Proactive & Personalized Assistance.
  + Deeper XAI and Trust Mechanisms.
  + Future-Proofing Strategies (Modular design, API-first, Continuous Learning Framework, Ethical/Responsible AI).

This compact roadmap should give a clear overview of where we are and where we're headed with ISA!