# 13. Future Roadmap

* Admin UI for document ingestion & versioning
* Summarization for large responses
* Real-time legislation update feed
* PDF upload for firm-specific documents
* Multi-language support (out of scope for MVP)
* Front - Tailwind CSS
* **Retrieval Pipeline:** Hybrid - Vector + BM25
* Data & Document Management
  + Admin UI: Future feature for doc ingestion and versioning

# 🔍 GAP Fill: Potential Enhancements Not Yet in Scope

1. **Performance & Scalability Expectations**

**System Should Handle**

* **Target response time: < 2 seconds**
* **Availability goal: 99.9% uptime SLA**
* **Scalability target: ~1,000 concurrent users**
* **100,000 queries per day at scale**
* **Action: Define benchmarks and incorporate load testing in CI/CD pipeline**

**Scalability**

* **Auto-scaling backend based on load**
* **Caching of common queries to reduce LLM costs**
* **Optimized database queries with appropriate indexing**
* **Token usage batching to reduce database writes**

**2. Encrypted Storage**

* **All documents and chat data should be encrypted at rest and in transit**
* **Action: Verify Fly.io and Supabase encryption defaults; document for GDPR/compliance clarity**

**3. Confidence Scoring / Ambiguity Handling**

* **Enable the LLM to return a confidence score or prompt for clarification when uncertain**
* **Use case: Reduces hallucinations, improves legal reliability**
* **Action: Tag as advanced RAG pipeline feature for later phase**

**4. Expanded Subscription Tiers**

* **Beyond Free and Paid, consider Business, Pro, Enterprise tiers**
* **Action: Reassess after launch depending on user segmentation or organizational interest**

**5. Historical Legal/Tax Toggle**

* **Allow querying legislation by year or effective period (e.g., “2021 tax rules”)**
* **Action: Requires storing versioned docs and enabling UI/UX toggle**
* **Use case: Supports backdated filings, audits, tax planning**

**6. Automated Document Reasoning (LLM Analysis on Corpus)**

* **Iterate through the chunked dataset using a reasoning-focused model (e.g., Claude Sonnet 3.5, GPT-4-o, or Gemini Pro) to generate:\n - Summaries\n - Metadata/insights\n - Potential user questions\n - Tax planning possibilities**
* **Action: Build a CLI or pipeline to augment stored metadata after ingestion**

**7. Table Extraction Service**

* **Support structured table extraction from legislation and guidance docs**
* **Use tools like Vectorize.io, LLamparser, or custom extraction services**
* **Action: Add “Extraction as a Service” module to pipeline for future ingestion rounds**

**8. NER-Based Annotation**

* **Run Named Entity Recognition to tag:\n - Tax Headers (e.g., CGT, CAT)\n - Deductions\n - Thresholds / Rates**
* **Use case: Improves filtering, advanced search, and context building**
* **Action: Use open-source models or fine-tune a legal/tax NER pipeline**

**9. Contractor Code Protection**

* **Protect RAG core logic and business-sensitive code from contractors or third-party devs**
* **Approach:**
  + **Minimize backend exposure via API boundaries**
  + **Obfuscate or compile logic-heavy components**
  + **Keep key logic in private Git repositories**
* **Action: Security policy to define contract dev access and API-only boundaries**

**10. LLM-Based Document Validation**

* **Input: .md + .pdf versions of legislation**
* **Task: Ask LLM to check if PDF is fully transcribed or missing sections**
* **Use case: Prevents incomplete ingestion and ensures legal completeness**
* **Action: Incorporate into CLI or admin document pipeline (pre-embedding)**