**45-Hour System Design Intensive Plan**

**Anchor Case Study:** [LexiFlow Repo](https://github.com/bhagwat-chate/lexi-flow)

**🔹 Phase 1 (Week 1) — DDIA Foundations + Infra Mapping (10 hours)**

**Goal:** Build DDIA mental models → Map them to GenAI Pyramid → Apply to LexiFlow’s infra.

**Activities:**

1. Read DDIA Ch. 1–3 (Data models, Storage engines, replication basics).
2. Draw LexiFlow HLD: Route53 → WAF → ALB → ECS/Fargate → S3/Vector DB.
3. Map DDIA → LexiFlow infra:
   * Replication → ECS multi-AZ + Vector DB replicas.
   * Partitioning → FAISS/Weaviate shards.
   * Logs → CloudWatch + SQS events.

**Deliverables (Artifacts):**

* 1 **DDIA → Pyramid → LexiFlow mapping table (for 3 chapters)**.
* 1 **LexiFlow HLD diagram (Infra layer)**.

**Time Split:**

* 4h DDIA reading & notes.
* 3h AWS infra deep dive (VPC, ECS, ALB/WAF).
* 3h mapping + diagrams.

**🔹 Phase 2 (Week 2) — RAG Design Patterns for LexiFlow (12 hours)**

**Goal:** Implement + analyze retrieval strategies with DDIA trade-offs.

**Activities:**

1. Study DDIA Ch. 5–6 (Replication, Partitioning).
2. Implement Hybrid Retrieval (BM25 + FAISS) in LexiFlow branch.
3. Add Dual-Stage RAG (recall → rerank with cross-encoder).
4. Compare **scalability trade-offs**: ANN HNSW vs IVF-PQ vs BM25.
5. Implement Context Cache (embedding cache with Redis/S3).

**Deliverables:**

* 1 **comparison notebook**: Hybrid vs Pure vector vs BM25 (with R@K, nDCG).
* 1 **LexiFlow retrieval design doc** showing trade-offs (latency, accuracy, cost).

**Time Split:**

* 4h DDIA reading & mapping.
* 6h coding retrieval experiments in LexiFlow.
* 2h eval + documentation.

**🔹 Phase 3 (Week 3) — Orchestration & Reliability Patterns (12 hours)**

**Goal:** Apply orchestration + infra reliability patterns to LexiFlow pipelines.

**Activities:**

1. Study DDIA Ch. 7–9 (Transactions, Distributed Systems, Consensus).
2. Implement LexiFlow DAG (LangGraph): retrieval → rerank → summarizer.
3. Add reliability patterns:
   * Circuit breaker (on retriever failure).
   * Loop guards & tool budget caps.
   * Backpressure → SQS between stages.
4. Stress test: simulate 100 parallel queries with autoscaling.

**Deliverables:**

* 1 **LangGraph DAG diagram** for LexiFlow pipeline.
* 1 **AWS backpressure + scaling test report**.
* Updated **LexiFlow architecture doc** with reliability patterns.

**Time Split:**

* 4h DDIA study.
* 6h coding orchestration + infra reliability.
* 2h experiments + diagrams.

**🔹 Phase 4 (Week 4) — Eval, Guardrails & Storytelling (11 hours)**

**Goal:** Add eval-in-loop, guardrails, rollback → package LexiFlow as case study.

**Activities:**

1. Study DDIA Ch. 10–12 (Batch, Stream, Future systems).
2. Implement LexiFlow eval suite: TruLens + DeepEval (R@K, nDCG, factuality).
3. Add Guardrails AI schema validation + PII filter.
4. Configure ECS rollback runbook (Blue/Green deploy + shadow testing).
5. Prepare storytelling pack:
   * 1-pager architecture
   * Sequence diagram (retrieval flow)
   * Eval dashboard screenshot

**Deliverables:**

* 1 **Eval & Guardrails doc** for LexiFlow.
* 1 **Rollback runbook** in AWS ECS.
* 1 **final storytelling case study deck** (HLD + LLD + trade-offs).

**Time Split:**

* 3h DDIA reading & notes.
* 6h coding eval + guardrails + rollback.
* 2h case study packaging.

**📊 Final Deliverables (End of 45h)**

1. **LexiFlow System Design Case Study** (HLD + LLD + Sequence + Trade-offs).
2. **DDIA → Pyramid → LexiFlow mapping table (all 12 chapters condensed)**.
3. **Retrieval benchmarking notebook** (Hybrid vs Pure vector).
4. **LangGraph DAG orchestration demo**.
5. **Eval dashboard + rollback runbook**.

**🎯 Why This Works**

* **DDIA coverage**: you’ll have touched all 12 chapters, but always through LexiFlow.
* **GenAI Pyramid coverage**: Infra, RAG, Orchestration, Eval/Guardrails all implemented in LexiFlow.
* **Interview readiness**: you’ll have concrete answers + diagrams for scalability, reliability, maintainability.
* **Portfolio**: at least 4–5 polished artifacts you can publish as posts/videos.