



Suhas Bhairav

AI Developer Onboarding Codebase Guide Setup

An actionable onboarding AI guide for engineering teams to navigate complex codebases

Deliver a practical, reproducible setup to build an interactive AI search assistant for new engineers.

Suhas Bhairav

suhasbhairav.com

Codebase Indexing & Domain Graph

Define the MVP scope for the onboarding AI. Build a navigable index of code, docs, and dependencies that a new engineer can query.

Key move Launch MVP for one service; validate with 3-5 onboarding queries.

PRACTICAL CHECKLIST

- 01** Automate collection of modules, packages, dependencies, and key interfaces from the repo.
- 02** Create a domain graph showing how components relate and where services call each other.
- 03** Ingest README, architecture docs, API specs, and notable test plans as a knowledge base.
- 04** Use non-sensitive snippets and summaries to protect secrets and IP.
- 05** Route human review for sensitive areas and high-risk modules before exposing results.
- 06** Deliver an interactive chat and code-search UI that surfaces mapped relationships.
- 07** Start with one service domain to test, then expand iteratively.

Suhas Bhairav AI Resource

Deliver a practical, reproducible setup to build an interactive AI search assistant for new engineers.

Data, Tools & Guardrails

This page outlines the data sources, tooling, and guardrails to collect and transform into the onboarding AI. It distinguishes automated components from human-reviewed elements.

Key move Establish data-sensitivity gates before indexing; secrets never leave secure storage.

PRACTICAL CHECKLIST

- 01** Data sources: repository metadata, docs, API specs, and test plans.
- 02** Automate indexing excerpts and embeddings; store in a secure vector store.
- 03** Human review points for architecture decisions, confidential data, and high-risk modules.
- 04** Mask secrets and credentials; separate code from output for the user.
- 05** Tools: code-search, embeddings, chat interface, and prompt templates.
- 06** Access controls, audit logs, and data retention policies.
- 07** Evaluation criteria for accuracy: relevance, completeness, and scaffold usefulness.

Suhas Bhairav AI Resource

Deliver a practical, reproducible setup to build an interactive AI search assistant for new engineers.

Weekly Adoption Plan & Ownership

Outline a realistic, incremental rollout with weekly milestones. Define ownership, approvals, and success metrics to keep alignment with business goals.

Key move Define ownership and schedule quarterly guardrail reviews.

PRACTICAL CHECKLIST

- 01** Week 1: data ingestion, indexing, and seed knowledge base; perform 5 test queries.
- 02** Week 2: build basic chat interface with search results; no code execution.
- 03** Week 3: integrate architecture questions with graph queries; enable onboarding prompts.
- 04** Week 4: security review, access controls, and secrets masking.
- 05** Ongoing weeks: measure usage, adjust prompts, and improve accuracy.
- 06** Metrics: time-to-first-meaningful-answer, onboarding task completion, and user satisfaction.

Suhas Bhairav AI Resource

Deliver a practical, reproducible setup to build an interactive AI search assistant for new engineers.

ABOUT THE CREATOR

Suhas Bhairav

I build AI systems that combine distributed architecture, retrieval, knowledge graphs, security, and practical workflow design. My focus is AI that can be used repeatedly by real teams: understandable, reliable, and grounded in the work people already do.

EXPERTISE

- Principal systems architecture and applied AI engineering
- Production-grade AI systems, agentic workflows, RAG, and knowledge graphs
- Full-stack workflow applications across AWS, Google Cloud, Azure, on-premise, and hybrid environments
- Practical AI implementation patterns for automotive, manufacturing, finance, logistics, cybersecurity, sales, and customer operations
- Security, governance, observability, approvals, and failure-mode thinking for operational AI systems

WEBSITE

suhasbhairav.com

EMAIL

suhasbhairav@gmail.com