Frameworks for Developing GenAI Applications

Open-Source LLM Frameworks

1. Hugging Face

- **Key Features**: Massive model hub, Transformers library, easy fine-tuning
- **Best For**: Researchers, developers wanting customization options
- **Notable Models**: Access to thousands of open models (BERT, T5, LLaMA derivatives)
- **Deployment**: Can be self-hosted or used with Hugging Face Inference API
- **Pricing**: Free open-source library, paid enterprise options

2. LangChain

- **Key Features**: Framework for building LLM applications, composable chains
- **Best For**: Building complex, multi-step LLM workflows
- **Integration**: Works with various LLM providers (OpenAI, Anthropic, etc.)
- **Components**: Document loaders, prompt templates, memory systems, agents
- **Pricing**: Open-source framework, pay for underlying LLMs

3. LlamaIndex (formerly GPT Index)

- **Key Features**: Data framework for LLM applications, RAG-focused
- **Best For**: Connecting custom data to LLMs, knowledge retrieval
- **Components**: Data connectors, indices, retrievers, query engines
- **Integration**: Compatible with many LLMs
- **Pricing**: Open-source, pay for underlying LLMs

Commercial LLM Providers

4. Anthropic Claude

- **Key Features**: Strong reasoning, longer context window, reduced hallucinations
- **Best For**: Safety-critical applications, complex reasoning tasks
- **Models**: Claude 3 family (Opus, Sonnet, Haiku, etc.)
- **API**: Well-documented REST API
- **Pricing**: Usage-based, competitive with OpenAI

5. Google AI (Gemini)

- **Key Features**: Multimodal capabilities, Google integration
- **Best For**: Applications requiring Google ecosystem integration

- **Models**: Gemini Pro, Gemini Ultra
- **API**: Vertex AI, Google AI Studio
- **Pricing**: Usage-based tiers

6. Meta AI

- **Key Features**: Open model weights, local deployment options
- **Best For**: Organizations wanting more control or lower inference costs
- **Models**: Llama family, CodeLlama
- **Deployment**: Self-hosted or via partners
- **Pricing**: Free model weights (with license), pay for deployment

7. Cohere

- **Key Features**: Specialized in embeddings and RAG
- **Best For**: Enterprise search, knowledge retrieval
- **Models**: Command, Embed
- **API**: Simple REST API
- **Pricing**: Usage-based tiers

Development Platforms

8. Replicate

- **Key Features**: Run hundreds of open-source models via API
- **Best For**: Experimentation with diverse models, quick prototyping
- **Integration**: Simple API calls to run various models
- **Models**: Access to many open-source models
- **Pricing**: Pay-per-second computing

9. Together AI

- **Key Features**: Platform for running open models
- **Best For**: Scaling open-source models in production
- **Models**: Supports most popular open models
- **Deployment**: Cloud API, enterprise deployment options
- **Pricing**: Competitive pay-as-you-go

10. LM Studio

- **Key Features**: GUI for running LLMs locally
- **Best For**: Local prototyping, privacy-focused applications
- **Installation**: Desktop application for Windows, Mac
- **Models**: Compatible with many open-source models
- **Pricing**: Free application, no API costs

Application Development Frameworks

11. Streamlit

- **Key Features**: Quick AI application development with Python
- **Best For**: Prototyping AI web apps, data science interfaces
- **Integration**: Works well with many LLM frameworks
- **Components**: Custom UI widgets for AI applications
- **Pricing**: Open-source, free hosting tiers available

12. Gradio

- **Key Features**: Create simple interfaces for ML models
- **Best For**: Demos, rapid prototyping of AI interfaces
- **Integration**: Easily integrated with Hugging Face models
- **Components**: Built-in UI components for AI applications
- **Pricing**: Open-source, free to use

13. LangServe

- **Key Features**: Deploy LangChain applications as REST APIs
- **Best For**: Productionizing LangChain applications
- **Integration**: Built for LangChain
- **Deployment**: Local or cloud deployment
- **Pricing**: Open-source

Vector Database Solutions (for RAG)

14. Pinecone

- **Key Features**: Vector database for semantic search
- **Best For**: Production-ready retrieval augmented generation
- **Integration**: Works with most LLM frameworks
- **Scale**: Built for large-scale vector operations
- **Pricing**: Free tier, paid plans based on vectors and operations

15. Chroma

- **Key Features**: Open-source embedding database
- **Best For**: Local development, smaller RAG applications
- **Integration**: Python API, works with LangChain
- **Deployment**: Self-hosted or managed
- **Pricing**: Open-source, free to use

16. Weaviate

- **Key Features**: Open-source vector database
- **Best For**: Multi-modal search, complex vector operations
- **Integration**: GraphQL API, Python client
- **Scale**: Enterprise-ready
- **Pricing**: Open-source, cloud service has free tier

Model Fine-tuning Solutions

17. Weight & Biases

- **Key Features**: MLOps platform for experiment tracking
- **Best For**: Managing multiple fine-tuning experiments
- **Features**: Experiment tracking, model versioning
- **Integration**: Works with many frameworks
- **Pricing**: Free tier, team plans available

18. Ludwig

- **Key Features**: Declarative machine learning framework
- **Best For**: No-code/low-code model training
- **Features**: Simplified model development
- **Integration**: Compatible with Hugging Face models
- **Pricing**: Open-source

Evaluation Frameworks

19. RAGAS

- **Key Features**: Evaluation framework for RAG systems
- **Best For**: Testing retrieval quality and answer relevance
- **Features**: Multiple evaluation metrics
- **Integration**: Works with LangChain, LlamaIndex
- **Pricing**: Open-source

20. TruLens

- **Key Features**: LLM application evaluation
- **Best For**: Measuring relevance, groundedness, feedback
- **Features**: Feedback functions, instrumentation
- **Integration**: Works with major frameworks
- **Pricing**: Open-source

Getting Started Recommendations

For those new to GenAI development:

1. **Start with LangChain or LlamaIndex** for application structure
2. **Choose a model provider** based on your needs (Anthropic, OpenAI, or open-source)
3. **Add a vector database** like Chroma for knowledge retrieval
4. **Build interfaces** with Streamlit or Gradio
5. **Evaluate performance** with TruLens or RAGAS