# Chat with PDF Project

Absolutely! Here's a **detailed, hierarchical syllabus of Generative AI** — broken down into **core modules**, **sub-topics**, and **leaf-level topics**. This is designed to take you from foundational AI concepts to building advanced GenAI apps like ChatGPT-like tools, image generators, and retrieval-augmented systems.

---

# 🧠 COMPLETE GENERATIVE AI SYLLABUS — DETAILED LEAF TOPICS

---

## 📗 Module 1: Foundations of AI, ML, and Deep Learning

### 1.1 Introduction to AI

- What is Artificial Intelligence (AI)?
- Types of AI: Narrow, General, Super
- AI vs ML vs DL vs GenAI
- Use Cases of AI

### 1.2 Machine Learning Basics

- Supervised, Unsupervised, Reinforcement Learning
- Regression and Classification
- Decision Trees, SVMs, KNN

### 1.3 Deep Learning Essentials

- Neural Networks (ANN)
- Forward and Backward Propagation
- Activation Functions (ReLU, Sigmoid, Tanh, Softmax)
- Loss Functions (Cross-Entropy, MSE)

- Optimization Algorithms (SGD, Adam)

## 1.4 Tools & Frameworks

- Python, NumPy, Pandas, Matplotlib

- scikit-learn, TensorFlow, PyTorch basics

---

# 🔧 Module 2: Transformers & Foundation Models

## 2.1 Understanding Transformers

- Sequence Models (RNN, LSTM overview)

- Attention Mechanism

- Scaled Dot-Product Attention

- Self-Attention vs Cross-Attention

- Positional Encoding

- Encoder-Decoder Architecture

## 2.2 Modern Transformers

- BERT, RoBERTa

- GPT-1, 2, 3, 3.5, 4

- T5, BART, LLaMA, Mistral, Claude

- Vision Transformers (ViT)

- Instruction tuning vs Fine-tuning vs RLHF

---

# 💡 Module 3: NLP with Large Language Models (LLMs)

## 3.1 Natural Language Processing

- Tokenization (Byte-Pair Encoding, WordPiece)

- Stopwords, Lemmatization, Stemming

- NER, POS tagging, Chunking

- Sentence and Document Embeddings

## 3.2 Large Language Models

- Architecture of GPT & LLaMA

- Pre-training and Transfer Learning

- In-context Learning & Few-shot Learning

- Prompt Engineering (detailed in Module 4)

## 3.3 Hands-On with APIs

- Using OpenAI GPT API (chat/completions)

- Using HuggingFace Transformers

- Model inference with `pipeline()` & `AutoModel`

---

# 🎯 Module 4: Prompt Engineering (PE)

## 4.1 Prompt Design Techniques

- Zero-shot Prompting

- One-shot and Few-shot Prompting

- Chain of Thought (CoT)

- Role-based Prompting

## 4.2 Prompt Engineering Tools

- OpenAI Playground

- LangChain PromptTemplates

- LlamaIndex Prompt Frameworks

## 4.3 Prompt Injection & Safety

- Jailbreak Prompts

- Preventing Prompt Injection

- Guardrails & Moderation APIs

# 📐 Module 5: Embeddings & Vector Databases

## 5.1 Embedding Concepts

- What are Embeddings?

- Word2Vec, GloVe, FastText

- SentenceTransformers & OpenAI Embeddings

## 5.2 Vector Similarity

- Cosine Similarity

- Euclidean Distance

- Approximate Nearest Neighbor Search (ANN)

## 5.3 Vector Databases

- FAISS: IndexFlatL2, IndexIVFFlat

- ChromaDB

- Pinecone

- Weaviate, Qdrant (intro only)

# 🔗 Module 6: LangChain & LlamaIndex

## 6.1 LangChain Core Concepts

- Documents and Loaders

- Text Splitters

- Embedding & Vector Store

- RetrievalQA Chain

- Conversational Memory

- Agents and Tool Usage

## 6.2 LlamaIndex Essentials

- Document ingestion & chunking

- Indexing documents

- Query Engines (Simple, Retriever, Composable Graphs)

- Response Synthesizers

# 📄 Module 7: Retrieval-Augmented Generation (RAG)

## 7.1 RAG Pipeline

- Extracting + Chunking PDFs, DOCs

- Generating and storing embeddings

- Query + Top-k Retrieval

- Building the Prompt + LLM Response

## 7.2 Applications of RAG

- Chat with PDF

- Chat with YouTube

- Internal Docs Chatbot

- Legal or Medical AI Assistant

# 🧩 Module 8: Multimodal GenAI (Text, Image, Audio, Video)

## 8.1 Text-to-Image Generation

- DALL·E, Midjourney, Stable Diffusion

- Prompting for Image Generation

- ControlNet, Inpainting

## 8.2 Image-to-Text & Vision Models

- BLIP, GPT-4o, Gemini Vision

- Visual Question Answering (VQA)

- CLIP (Contrastive Language-Image Pretraining)

## 8.3 Audio + Video

- Whisper for Speech-to-Text

- Bark / Tortoise / ElevenLabs for Text-to-Speech

- RunwayML for Video Gen

- Sora (OpenAI) — Concepts only

---

# 🔧 Module 9: Model Fine-tuning & Training

## 9.1 Fine-Tuning Fundamentals

- Full fine-tuning vs LoRA

- PEFT: Parameter Efficient Fine Tuning

- Dataset preparation for training

## 9.2 Tools for Training

- Hugging Face `transformers`, `datasets`, `accelerate`

- `peft`, `bitsandbytes`, `transformers` Trainer

## 9.3 Evaluating Models

- BLEU, ROUGE, METEOR (for text gen)

- Embedding similarity evaluation

- Human evaluation techniques

---

# 🚀 Module 10: LLM App Development & Deployment

## 10.1 Backend Development

- FastAPI, Flask basics

- RESTful API creation for GenAI

- Upload endpoints, file parsing

## 10.2 Frontend Development

- React.js, Tailwind CSS

- shadcn/ui, file upload components

- Gradio / Streamlit for Python-based UI

## 10.3 Full-stack Integration

- Connecting React frontend to FastAPI

- Hosting Vector DBs (locally or on cloud)

- Fetch + display LLM responses

## 10.4 Deployment

- Vercel (frontend)

- Render / Railway (backend)

- Hugging Face Spaces (streamlit/gradio)

- Docker (optional)

---

# 🛡️ Module 11: Ethics, Safety, and Responsible GenAI

## 11.1 Ethical Considerations

- Bias in Language Models

- Toxicity, Misinformation

## 11.2 Security

- Prompt Injection Attacks

- Data Privacy in GenAI

## 11.3 Governance & Fair Use

- Model licensing (Open vs Closed)

- Copyright & Fair Use in Generated Content

---

# 🧪 Module 12: Capstone Projects (Build & Deploy)

- 🧾 Chat with PDF (RAG + LangChain + Chroma)

- 📚 Chat with YouTube Transcript

- 🗣️ AI Voice Assistant (Speech-to-Text + LLM + TTS)

- 🧠 AI Resume Analyzer / Generator

- 🎨 AI Image Generator from Prompt

- 🔍 Semantic Search Engine

- 🧑‍⚕️ Medical Q&A Chatbot

- 💼 AI Job Interviewer (Q&A)

---

Would you like this as a downloadable **Notion template**, **PDF**, or **Excel planner** to track your GenAI journey?