FLORIDA ATLANTIC

COT 6930 - Generative Intelligence
and Software Development Lifecycle

Topic 1 -
Generative Intelligence and Software Development Lifecycles

Dr. Fernando Koch
kochf@fau.edu
http://www.fernandokoch.me

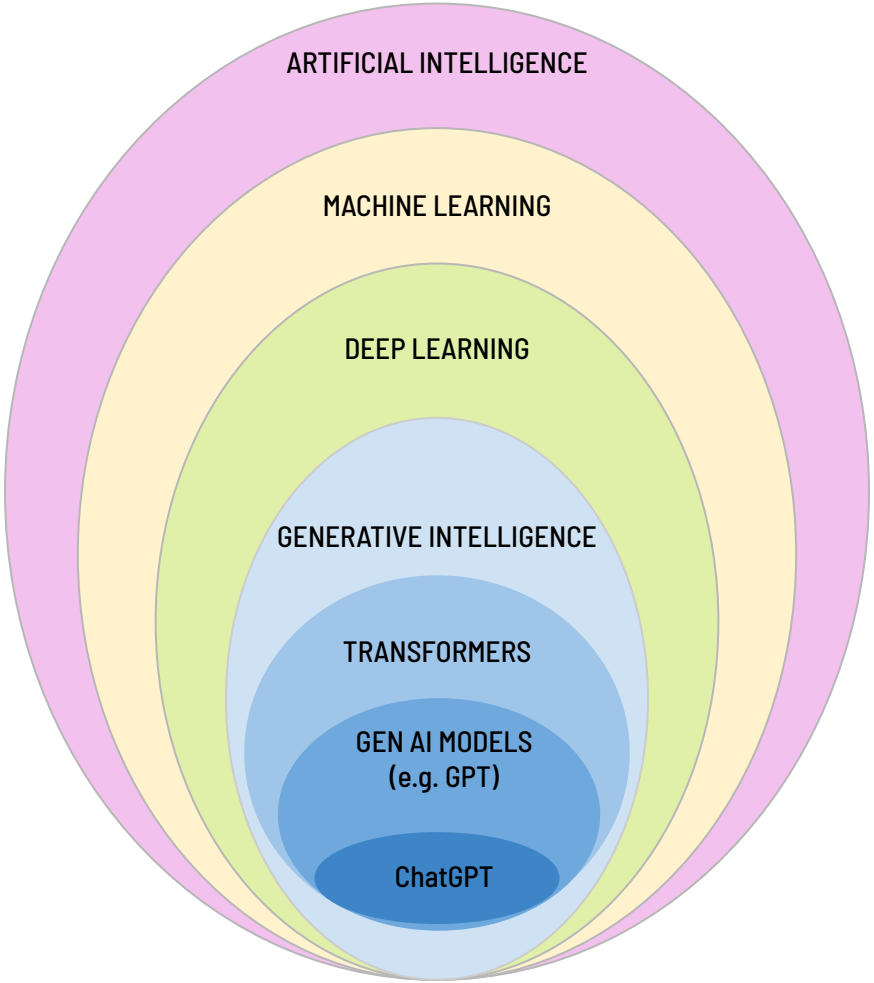
Agenda

- Introduction to GenAI
- Impact of GenAI on SDLC
- Use Cases
- Exercises

Our Key Question

How do Generative Intelligence Systems impact software development lifecycle by enhancing productivity and creativity across development stages?

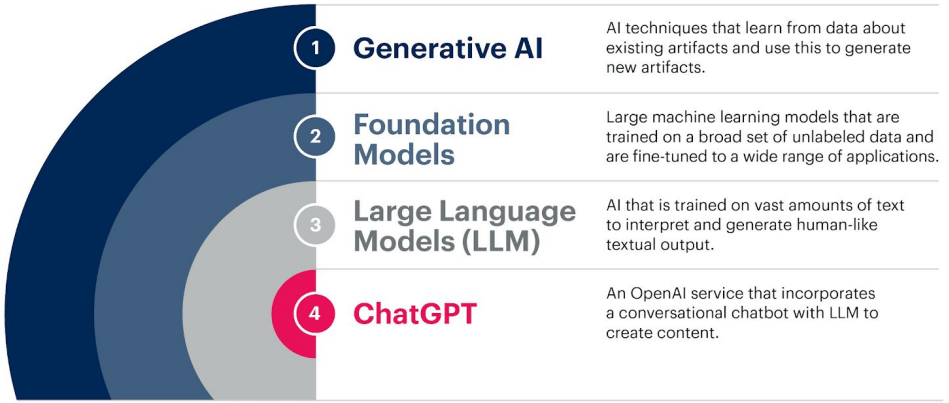
What is Generative AI?



Generative AI

AI that creates new, coherent content

- Generates text, code, images, audio, and more
- Trained on large datasets to learn patterns and structures
- Foundation Models act as the core engines powering generative capabilities
- Capable of producing original outputs, not just ‘retrieval’
- Applications span creative writing, design, coding, and synthesis
- Examples: GPT, DALL·E, Stable Diffusion



Source: Gartner

Gartner

Why is Generative AI different?

1950s-

Computation Models

If-Then-Else
Finite State Machines
Hardcode Rules

1970s-

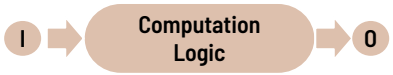
Retrieval Models

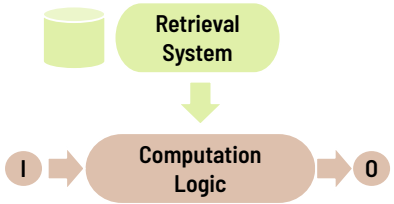
Databases
Rule-Base Systems
Decision Trees

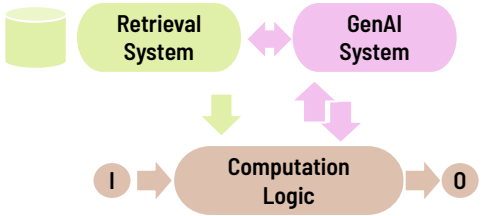
2010-

Generative Models

Large Language Models
Generative Adversarial Networks
Diffusion Models

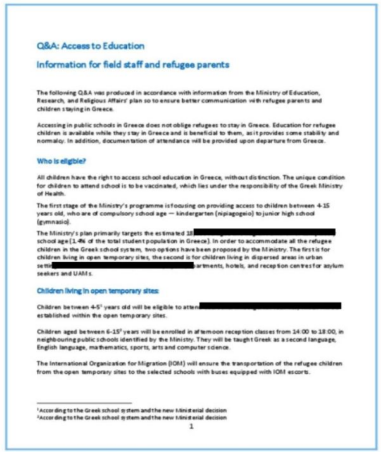






The Process of Creating Generative Models

(1) Inference:
Generate the content to complete the document based on predictions by the Model.



Q&A: Access to Education

Information for field staff and refugee parents

The following Q&A was produced in accordance with information from the Ministry of Education, Research, and Religious Affairs' plan so to ensure better communication with refugee parents and children staying in Greece.

Accessing in public schools in Greece does not oblige refugees to stay in Greece. Education for refugee children is available while they stay in Greece and is beneficial to them, as it provides some stability and normalcy. In addition, documentation of attendance will be provided upon departure from Greece.

Who is eligible?

All children have the right to access school education in Greece, without distinction. The unique condition for children to attend school is to be vaccinated, which lies under the responsibility of the Greek Ministry of Health.

The first stage of the Ministry's programme is focusing on providing access to children between 4-15 years old, who are of compulsory school age — kindergarten (nipiagogeio) to junior high school (gymnasio).

The Ministry's plan primarily targets the estimated [redacted] In order to accommodate all the refugee children in the Greek school system, two options have been proposed by the Ministry. The first is for children living in open temporary sites, the second is for children living in dispersed areas in urban set[redacted]artments, hotels, and reception centres for asylum seekers and UAMs.

Children living in open temporary sites:

Children between 4-5[1] years old will be eligible to atten[redacted] established within the open temporary sites.

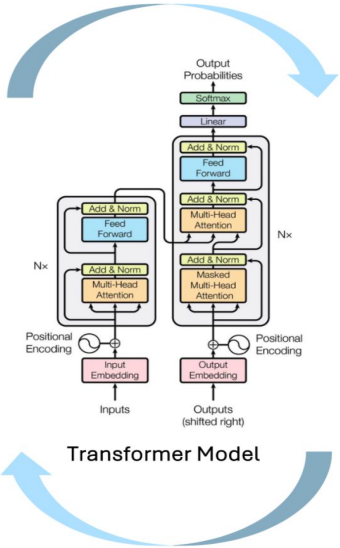
Children aged between 6-15[2] years will be enrolled in afternoon reception classes from 14:00 to 18:00, in neighbouring public schools identified by the Ministry. They will be taught Greek as a second language, English language, mathematics, sports, arts and computer science.

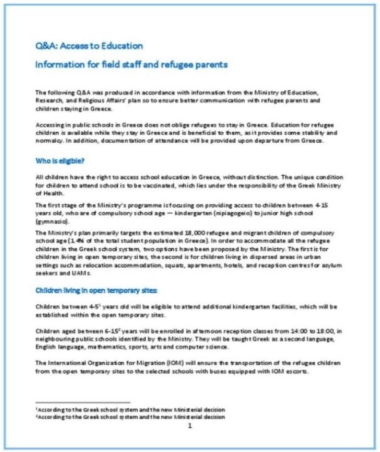
The International Organization for Migration (IOM) will ensure the transportation of the refugee children from the open temporary sites to the selected schools with buses equipped with IOM escorts.

[1] According to the Greek school system and the new ministerial decision
[2] According to the Greek school system and the new ministerial decision

1

(A) Document with Obfuscated Parts





Q&A: Access to Education

Information for field staff and refugee parents

The following Q&A was produced in accordance with information from the Ministry of Education, Research, and Religious Affairs' plan so to ensure better communication with refugee parents and children staying in Greece.

Accessing in public schools in Greece does not oblige refugees to stay in Greece. Education for refugee children is available while they stay in Greece and is beneficial to them, as it provides some stability and normalcy. In addition, documentation of attendance will be provided upon departure from Greece.

Who is eligible?

All children have the right to access school education in Greece, without distinction. The unique condition for children to attend school is to be vaccinated, which lies under the responsibility of the Greek Ministry of Health.

The first stage of the Ministry's programme is focusing on providing access to children between 4-15 years old, who are of compulsory school age — kindergarten (nipiagogeio) to junior high school (gymnasio).

The Ministry's plan primarily targets the estimated 18,000 refugee and migrant children of compulsory school age (1.4% of the total student population in Greece). In order to accommodate all the refugee children in the Greek school system, two options have been proposed by the Ministry. The first is for children living in open temporary sites, the second is for children living in dispersed areas in urban settings such as relocation accommodation, squats, apartments, hotels, and reception centres for asylum seekers and UAMs.

Children living in open temporary sites:

Children between 4-5[1] years old will be eligible to attend additional kindergarten facilities, which will be established within the open temporary sites.

Children aged between 6-15[2] years will be enrolled in afternoon reception classes from 14:00 to 18:00, in neighbouring public schools identified by the Ministry. They will be taught Greek as a second language, English language, mathematics, sports, arts and computer science.

The International Organization for Migration (IOM) will ensure the transportation of the refugee children from the open temporary sites to the selected schools with buses equipped with IOM escorts.

[1] According to the Greek school system and the new ministerial decision
[2] According to the Greek school system and the new ministerial decision

1

(B) Completed Document

(2) Training:
Learn how to complete the missing parts provided enough examples.

Traditional AI

- Focus on classification, prediction, and decision rules
- Relies on structured data (e.g., tables, labels, metrics)
- Uses supervised learning with predefined outcomes

Generative AI

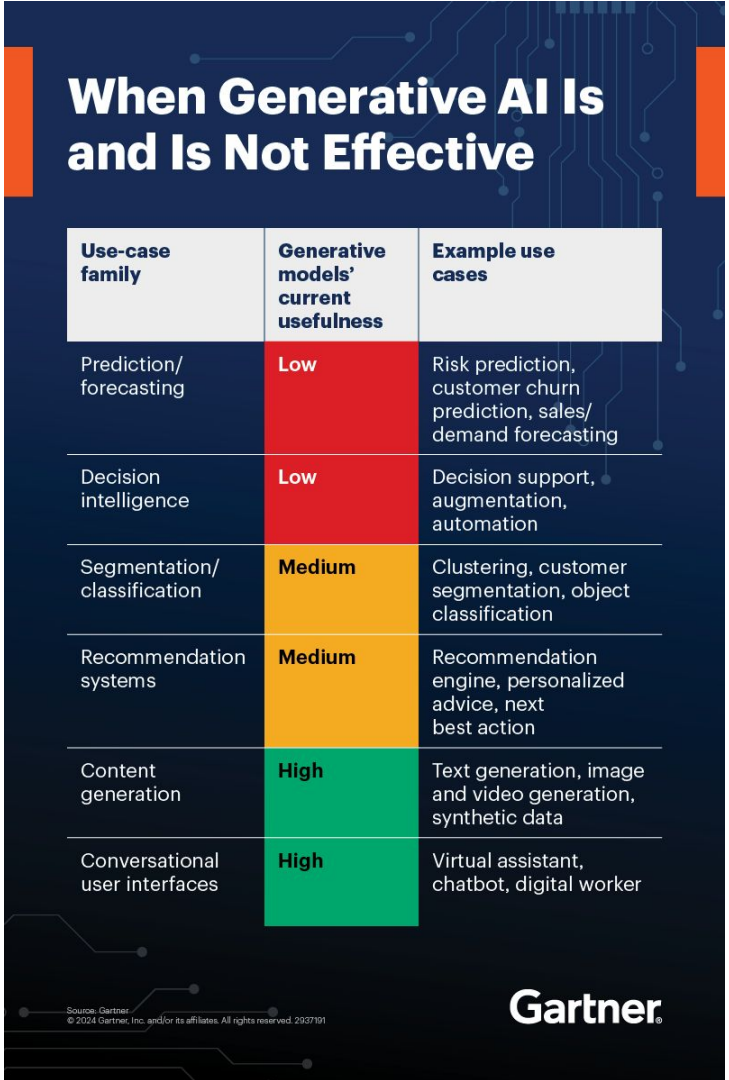
- Creates novel data like images, text, and code
- Learns from unstructured sources using self-supervised techniques
- Often unsupervised or self-supervised

What Generative AI can do?

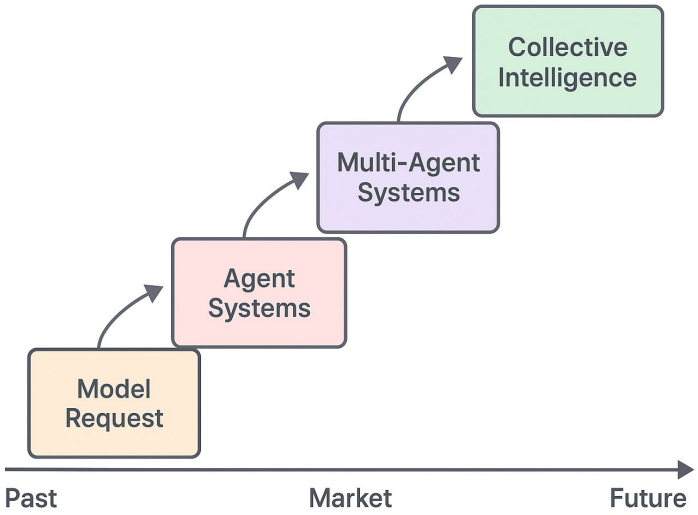
The most common GenAI tasks today

Retrieval-augmented generation Based on documents or dynamic content, create a chatbot or question-answering feature. Building a Q&A resource from a broad knowledge base, providing customer service assistance	**Summarization** Transform text with domain-specific content into personalized overviews that capture key points. Conversation summaries, insurance coverage, meeting transcripts, contract information	**Content generation** Generate text content for a specific purpose. Source Code, SQL, descriptions, posts, articles, email, others
Named entity recognition Identify and extract essential information from unstructured text. Audit acceleration, SEC 10K fact extraction	**Insight extraction** Analyze existing unstructured text content to surface insights in specialized domain areas. Medical diagnosis support, user research findings	**Classification** Read and classify written input with as few as zero examples. Sorting of customer complaints, threat and vulnerability classification, sentiment analysis, customer segmentation

Where GenAI does not help?



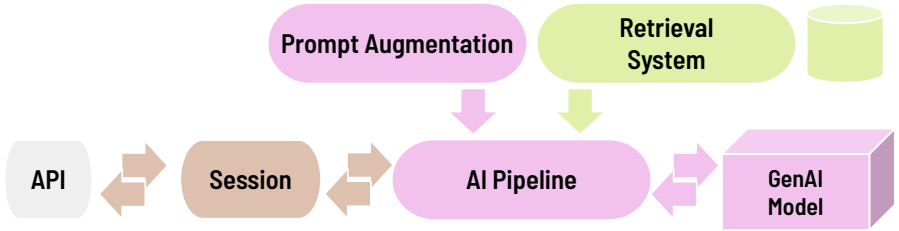
Evolution of Utilization of Generative AI



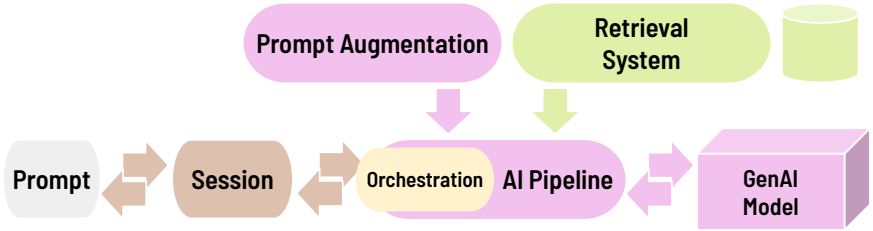
Generative Intelligence System

End-to-end systems with autonomous content generation

- Session-aware workflows.
- Automated prompt engineering.
- Context-aware content generation
- Adaptive behaviour combining orchestration, memory, retrieval augmentation, others.
- End-to-end automation for LGM consumption.



Generative Intelligence System



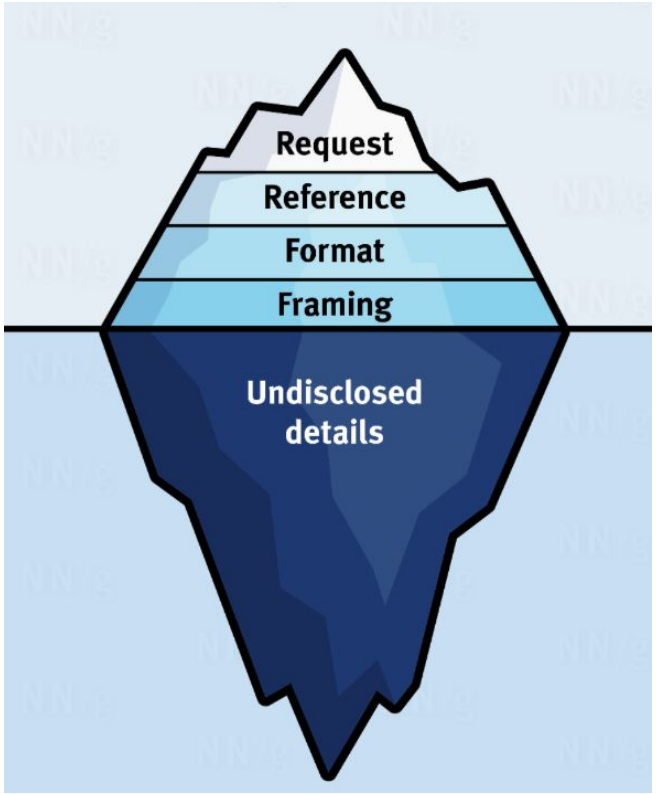

```python
# ---------- API handler ----------
repo = SessionRepository()

def api_handler(input_message: str, context: str) -> str:
    # 0) Load session by context
    session = repo.get(context)

    # 1) Accept new request into session
    session.current.request = input_message

    # 2) Prompt Augmentation (build the 🔶 prompt)
    pre  = "You are a helpful assistant.\n=== BEGIN USER REQUEST ===\n"
    post = "\n=== END USER REQUEST ===\nProvide a clear, concise reply."
    session.current.prompt = pre + session.current.request + post   # 🔶 highlight

    # 3) Retrieval (optional) → enrich the 🔶 prompt
    ctx = retrieval_system(query=session.current.request)
    if ctx:
        session.current.prompt += f"\n\nContext:\n{ctx}"           # 🔶 keep prom

    # 4) AI Pipeline → GenAI model (configurable)
    llm = GenAIModel(
        model="llama3",        # or "llama3:70b"
        temperature=0.3,
        num_ctx=8192,          # acts like context window for Ollama
        num_predict=512
    )
    session.current.response = llm.generate(session.current.prompt)

    # 5) Persist session and return response to chatbot
    repo.save(session)
    return session.current.response

# ---------- Example ----------
reply = api_handler(
    input_message="What is LangGraph?",
    context="tenantA:user42:chat1"  # the routing/index key
)
print(reply)
```

What are Prompts?

Prompts are the inputs or instructions given to a generative AI model to guide its response.

- Act as the "query" or "command" for the AI
- Shape the style, tone, and depth of the output
- Can be questions, statements, or structured instructions
- The quality of a prompt directly affects the quality of the response

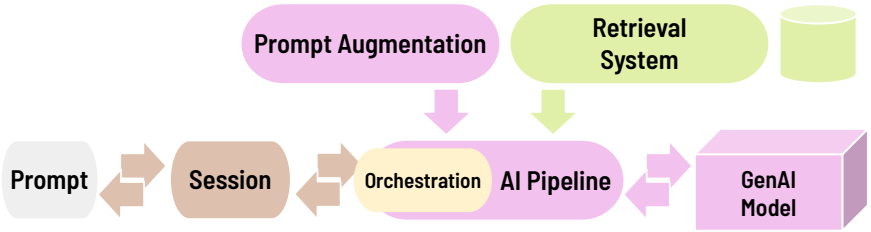
Note: we will have a deeper understanding of ‘the math’ behind Prompts as part of our deep dive in Topic 2: Large Generative Models.



What are Sessions?

A session is the ongoing interaction between a user (or system) and the AI model.

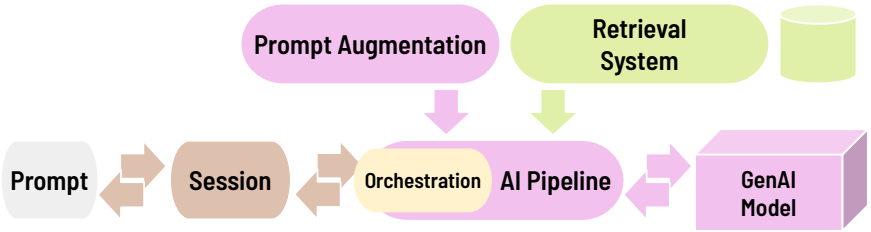
- Tracks the sequence of prompts and responses
- Maintains context across multiple turns
- Enables refinement and iteration over answers
- Critical for conversational AI and complex workflows



What is an AI Pipeline?

Structured sequence that transforms user inputs into model responses.

- Links prompts, retrieval, and model interaction
- Handles data preparation and enrichment
- Supports modular composition of components
- Enables repeatable and adaptive workflows

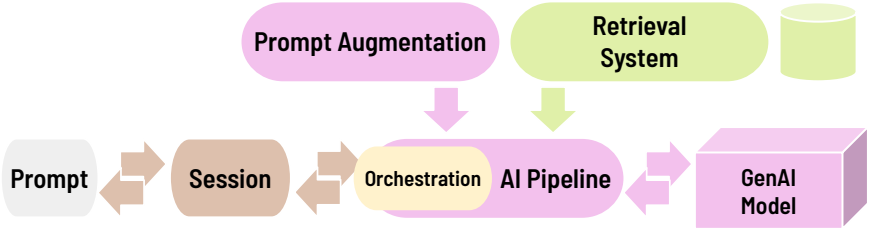


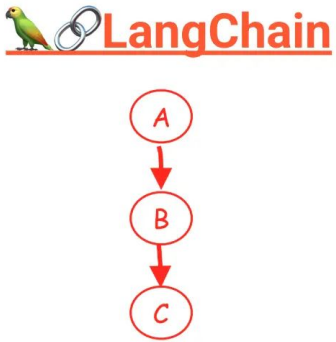
What are Orchestration Workflows?

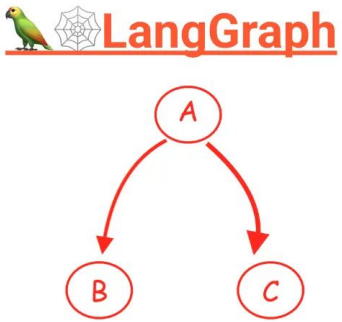
Frameworks to manage AI pipeline execution.

- Schedule, route, and monitor tasks
- Enable conditional branching and looping
- Support multi-step dependencies

- **LangChain**: modular chaining of model calls
- **LangGraph**: graph-based orchestration for adaptive flows



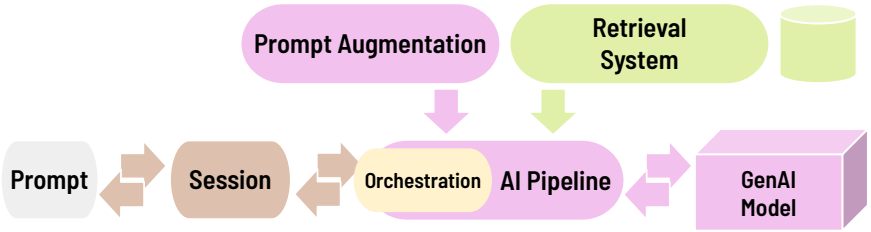




What s the role of the Retrieval System?

Mechanism to fetch domain-specific data for augmentation.

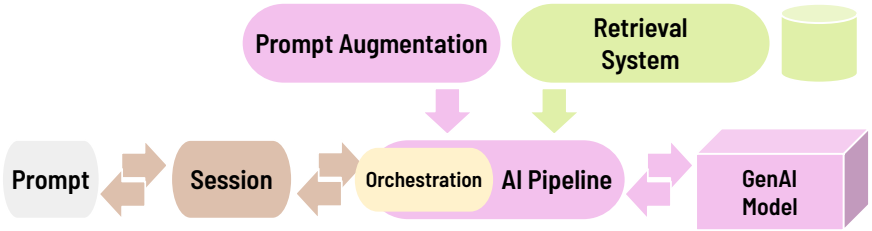
- Enhances prompts with relevant information
- Reduces model hallucinations
- Ensures context-aware responses
- Example:
    a. Database retrieval
    b. Knowledge-Based Retrieval
    c. Knowledge Tree, others



How to connect to GenAI Models?

Methods to access generative AI capabilities.

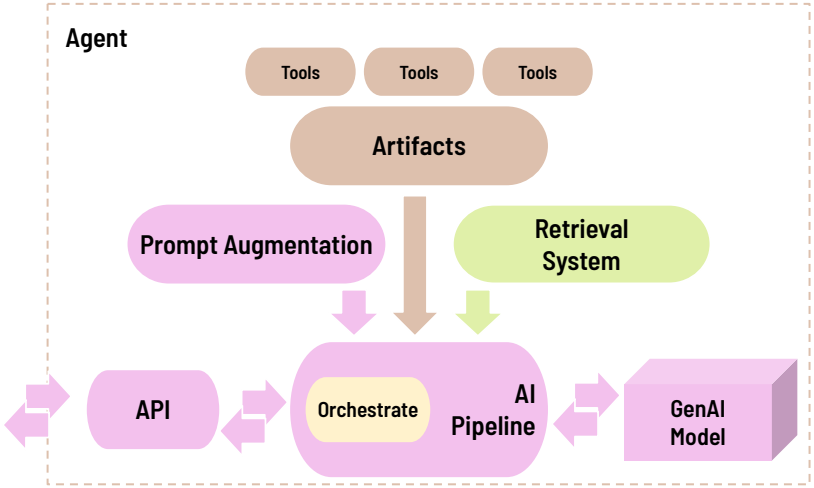
- Cloud API endpoints (e.g., OpenAI, Anthropic)
- Local API endpoints (e.g. Ollama)
- SDK endpoints
- Request parameters:
  a. Context Window
  b. max_output_tokens / max_tokens.
  c. Temperature
  d. top_p, top_k, frequency/presence_penalty



What are Agentic Systems?

Agentic systems =
LLMs + tools + memory + autonomy.

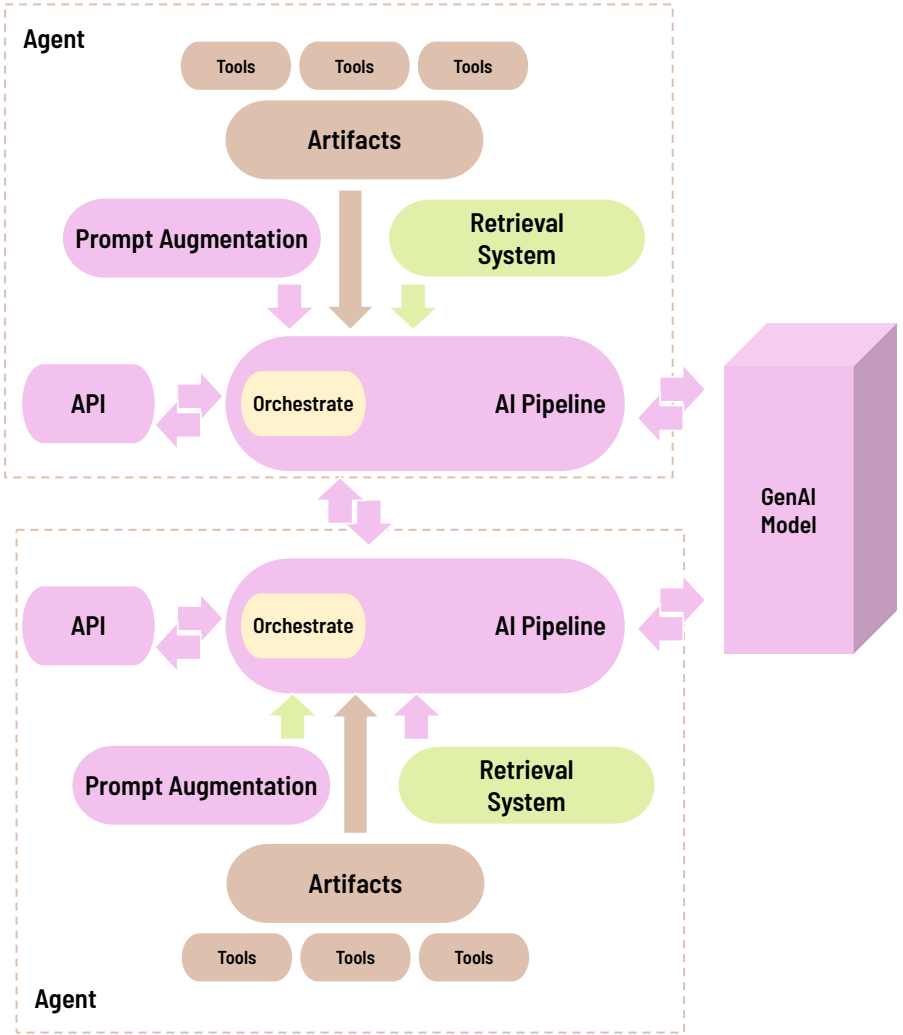
- **LLM-driven components capable of planning**, executing, and reasoning over sequences of actions.
- **Tools** provide external capabilities (e.g., APIs, search, code execution)
- **Prompt Augmentation:** Dynamically reformulates inputs based on past state, tools, and user goals.
- **Orchestration Layer:** Manages decisions, branching logic, and execution flows



What are Multi-Agent Systems?

Composed of multiple autonomous agents that collaborate, delegate, or compete to accomplish complex tasks. Each agent:

- Has its own goal, memory, tools, and orchestration logic.
- May perform specialized roles (e.g., planner, coder, tester, executor).
- Can communicate with other agents via shared context or message passing.



How people are using GenAI?

Generative AI

How People Are Really Using Gen AI in 2025

by Marc Zao-Sanders

April 9, 2025

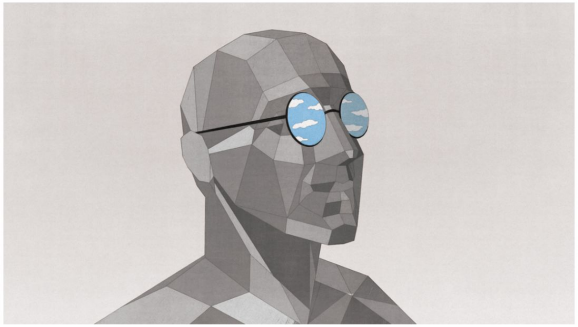


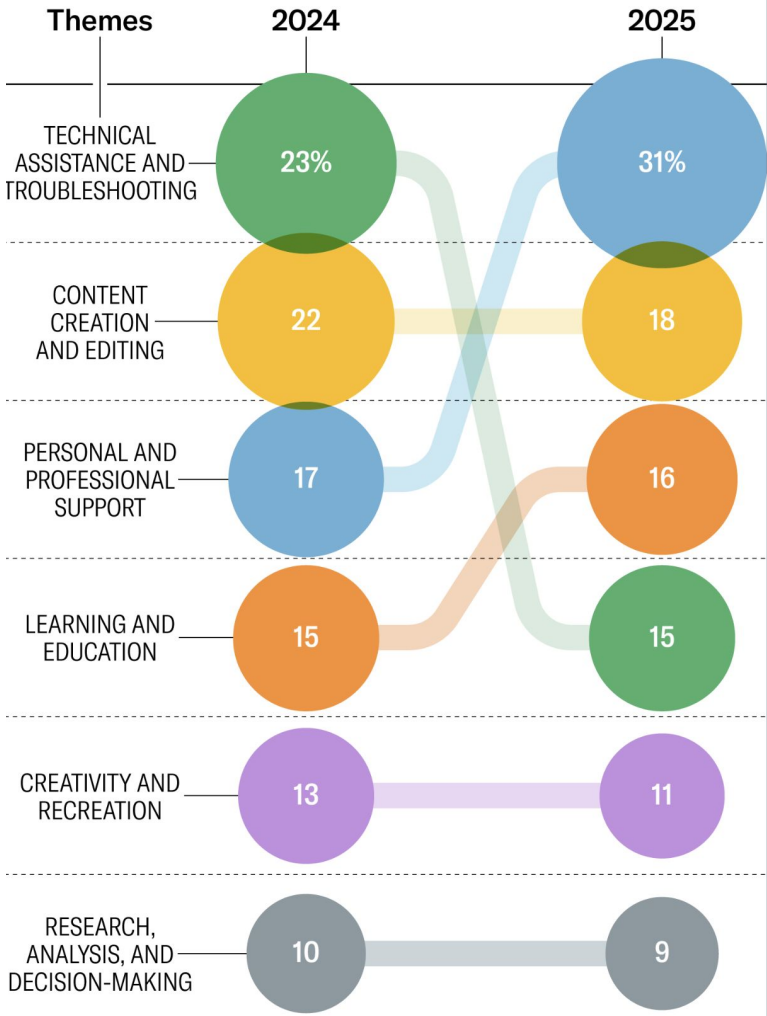
Illustration by Andrea Ucini

Summary. Last year, HBR published a piece on how people are using gen AI. Much has happened over the past 12 months. We now have Custom GPTs—AI tailored for narrower sets of requirements. New kids are on the block, such as DeepSeek and Grok, providing more... **more**

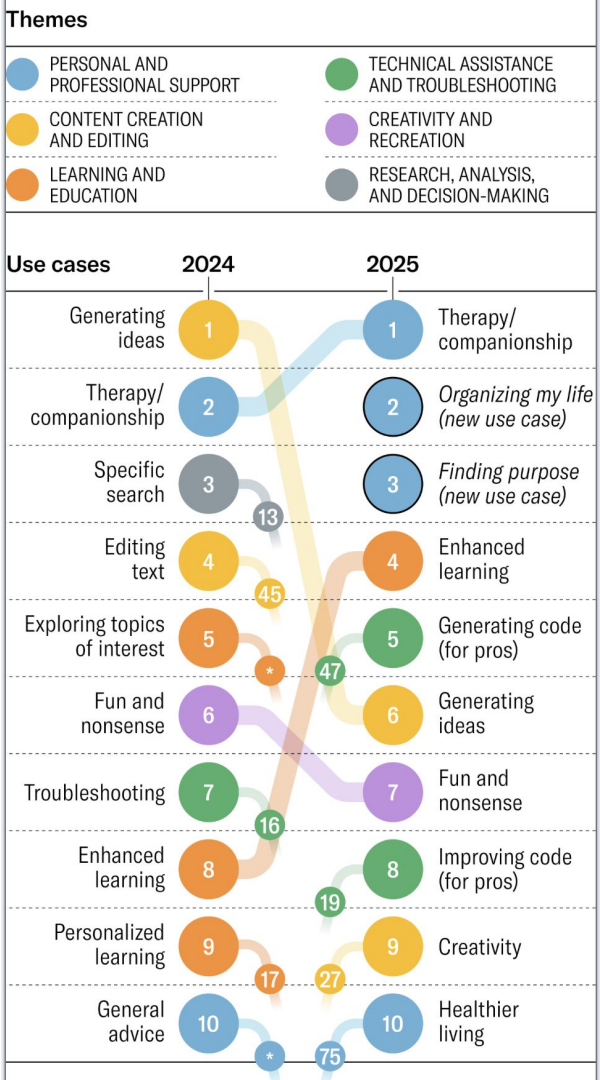
https://hbr.org/2024/03/how-people-are-really-using-genai

A year ago, I wrote a piece here about how people were really using gen AI. That article seemed to hit a note: It was popular, featured in viral posts, and the beautiful accompanying infographic has been shared far and wide. The use cases split almost equally between personal and business needs, with roughly half spanning both.

https://hbr.org/2024/03/how-people-are-really-using-genai

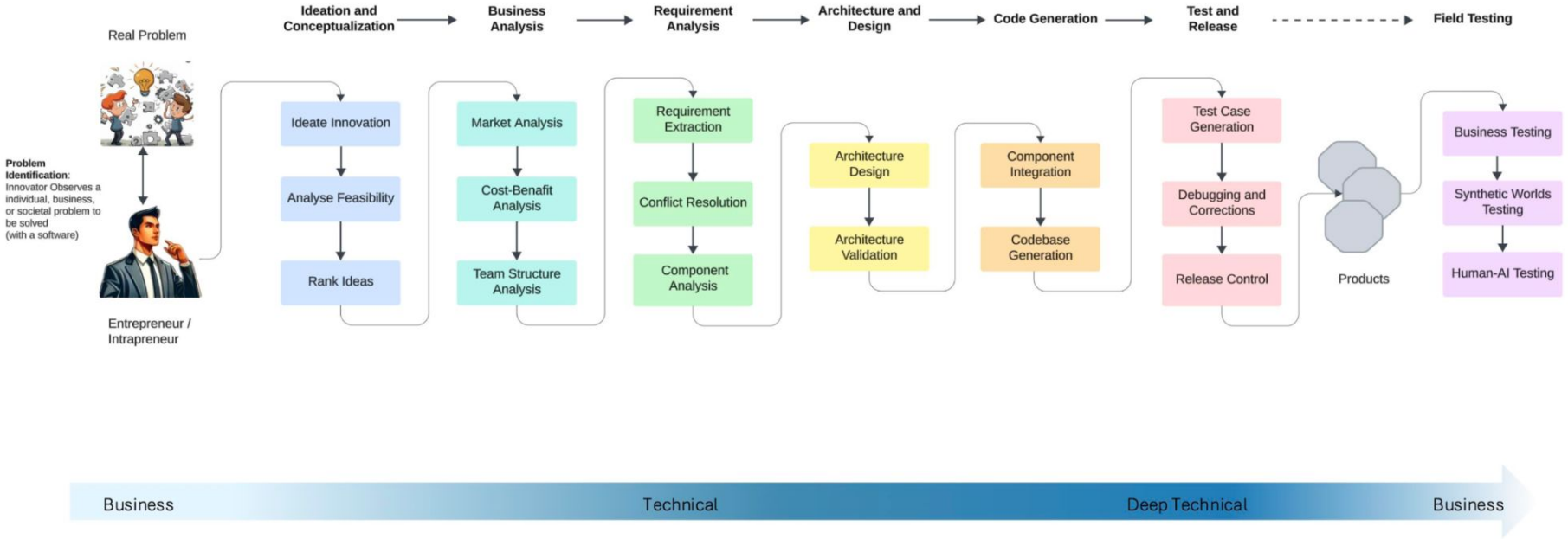


https://hbr.org/2024/03/how-people-are-really-using-genai



GenAI and SDLC

Software Development Lifecycles



Steps in the SDLC Process

Problem Identification

- Understand stakeholder needs
- Define the core problem
- Align with organizational objectives

Solution Ideation

- Collaborate to generate diverse ideas
- Evaluate feasibility and impact
- Develop a conceptual framework

Requirement Analysis

- Clarify detailed requirements
- Prioritize requirements
- Define acceptance criteria

System Design

- Translate requirements into system architecture.
- Incorporate technologies
- Ensure alignment between design and objectives

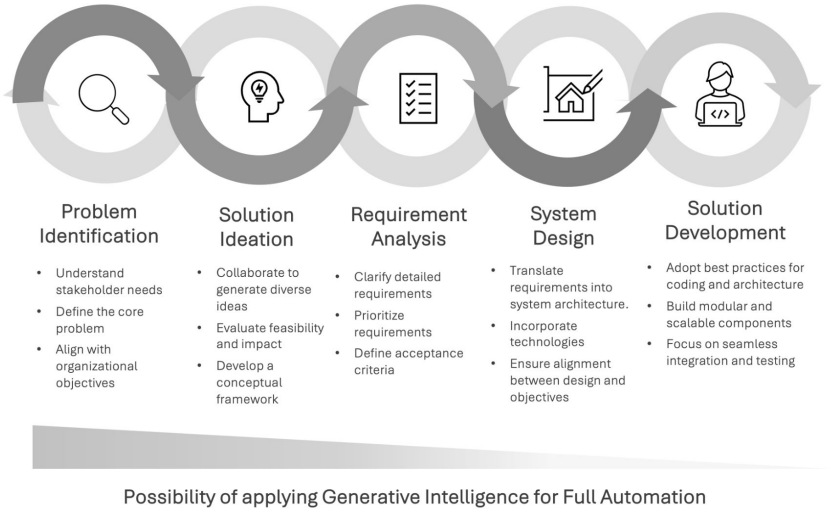
Solution Development

- Adopt best practices for coding and architecture
- Build modular and scalable components
- Focus on seamless integration and testing

Impact of GenAI on the SDLC Process

1. **Problem Identification**
   - Analyze user feedback and logs to surface pain points.
   - Cluster user needs using LLMs + embedding-based retrieval.

2. **Solution Ideation**
   - Brainstorm multiple design ideas using prompt-based generation.
   - Generate conceptual frameworks or architectural option



Impact of GenAI on the SDLC Process

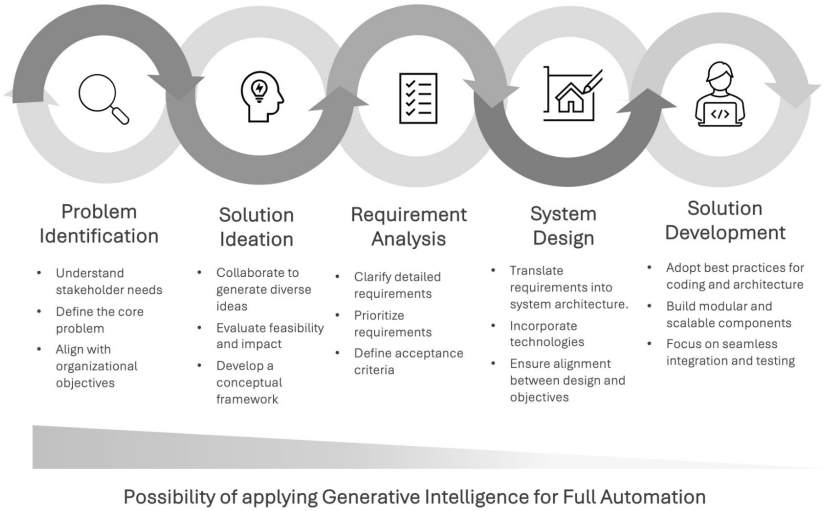
3. Requirement Analysis

- Convert informal business needs into formal requirements.
- Generate user stories and acceptance criteria.

4. System Design

- Propose system architectures aligned with requirements.
- Visualize component interactions or data flows using AI-generated diagrams.

5. Solution Development



EXERCISE

Exercise 1 - Getting Started with MyBot

Objective: Apply OwlMind Framework to develop a basic conversational bot to familiarize with GenAI interfaces and simple interactions.

https://github.com/genilab/owlmind/wiki/Exercises

COT 6930 - Generative Intelligence and Software Development Lifecycles

Dr. Fernando Koch
kochf@fau.edu
http://www.fernandokoch.me