

# AN OFFLINE AI ASSISTANT FOR BIOINFORMATICS PIPELINES

Authors

Martinus Grady Naftali, Ligan Cai, Zeynal Mardanli

Supervisors

Sébastien Lafond, Hergys Rexha, Adriano Di Pasquale, Nicolas Radomski, Andrea de Ruvo



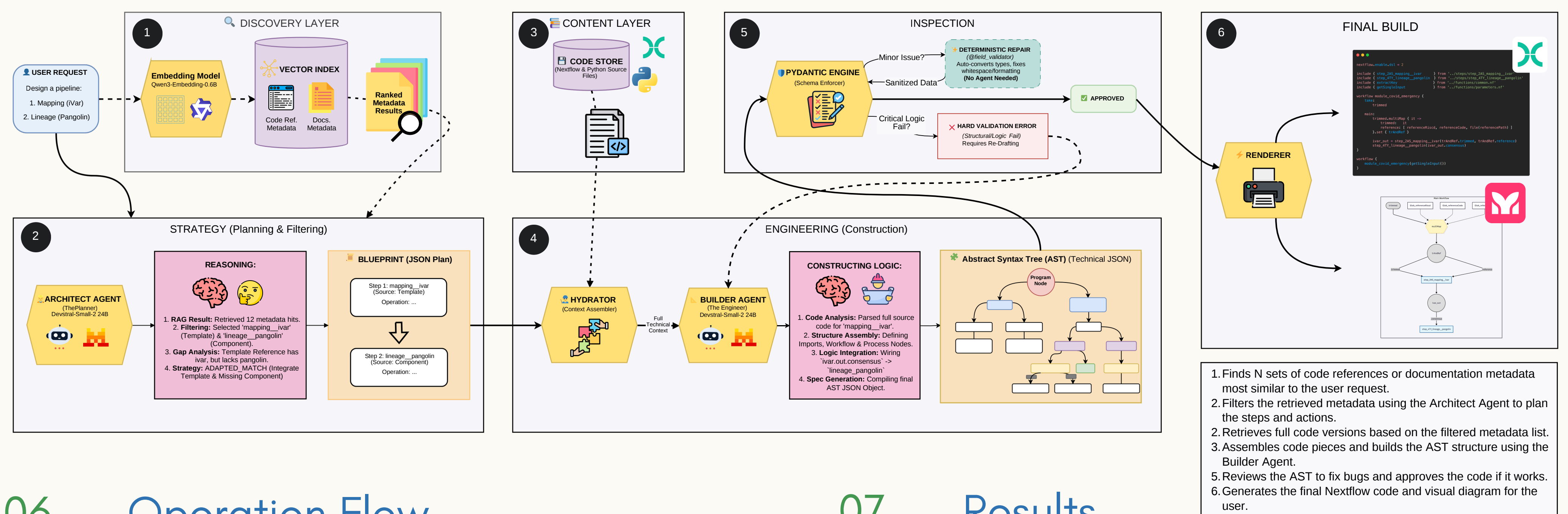
## 01 Problem Statement

Bioinformatics uses computational methods to analyze biological data. The COHESIVE bioinformatics platform manages pathogen data using Nextflow pipelines. However, modifying these pipelines requires programming skills. This makes it hard for scientists to work independently and slows down research. We need an offline LLM Assistant to translate natural language into executable code to keep sensitive data safe.

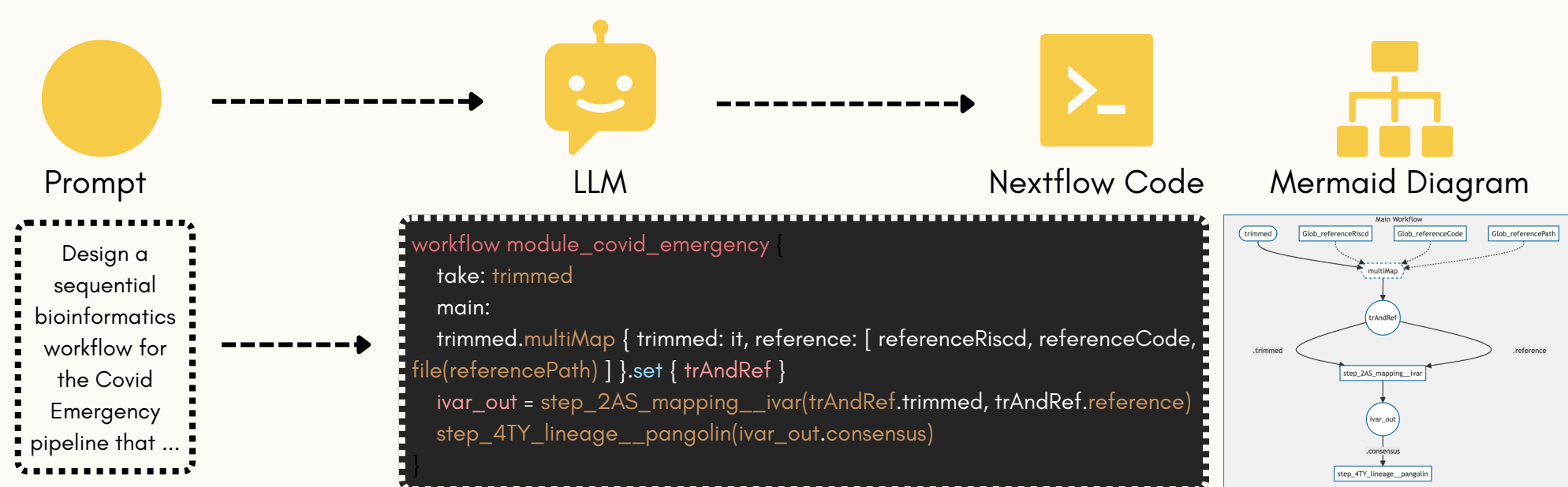
## 02 Objective

- Bridge the Skills Gap
- Ensuring Data Privacy and Security
- Enhancing Reliability and Consistency
- Improving Usability and Verification
- Utilizing Context-Aware Resources - COHESIVE Files

## 05 Pipeline



## 06 Operation Flow



## 08 Challenges

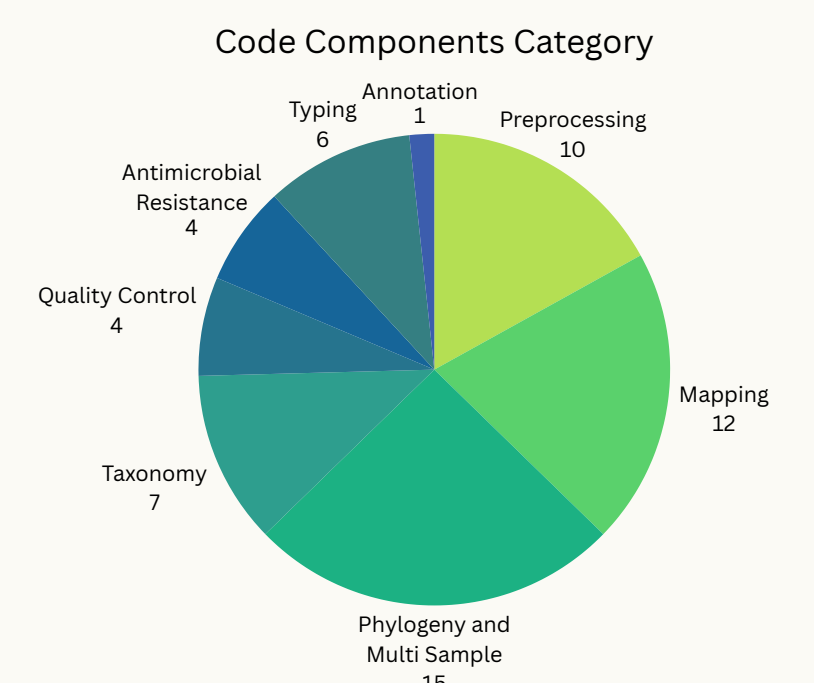
- Forcing the model to only pick real tools from COHESIVE.
- Writing strict Nextflow DSL2 code so versions do not mix.
- Using Pydantic validations to keep the logic and syntax correct.
- Handling a large context from the existing code references.

## 09 Future Work

We want to make the system faster and more reliable. We plan to add strong guardrails to stop prompt injection. We also need to make the pipeline more efficient so it runs much faster. Finally, we will build an automatic nightly update to download new COHESIVE tools every night.

## 03 Dataset

Our dataset is composed of existing Nextflow code from COHESIVE system repository.



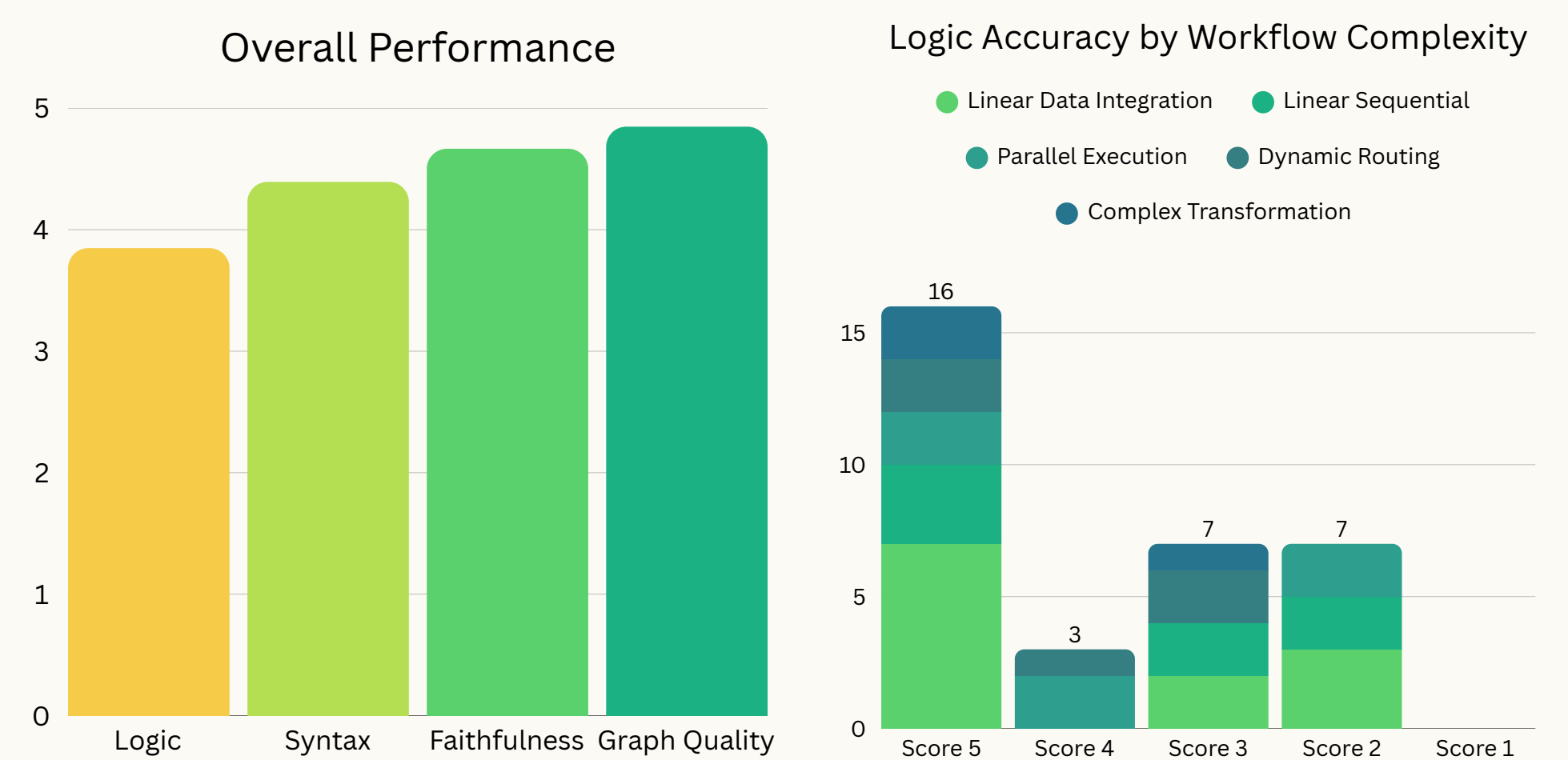
## 04 Model

Devstral Small 2 performed the best in our test when handling a lot of RAG data. It made very few errors and hallucinated less.

- Lightweight: 24 Billion Parameters.
- Context Window: A 256k context window.
- Agentic Coding: Devstral is designed to excel at agentic coding tasks, a great choice for software engineering agents.

## 07 Results

We evaluated the agent on 33 Nextflow generation scenarios (recreating, customizing, and designing new pipelines). Outputs were scored via an LLM-as-a-judge approach across 4 metrics.



- Logic: Accuracy of data routing, channel synchronization, and architectural design.
- Syntax: Code validity and successful Nextflow DSL2 compilation.
- Faithfulness: Strict adherence to provided local tools without hallucinating external modules.
- Graph Quality: Accuracy of the Mermaid flowchart in visualizing the generated code.