Executive Summary

This project systematically evaluated three Large Language Models (LLMs)- GPT-5, Gemini Pro, and Gemma3:4b, across three developer-focused tasks: Code Summarization, API Question Answering, and API Documentation Generation. Each task was tested using three prompting strategies: CLEAR, Few-shot, and Chain-of-Thought (CoT), generating a total of 27 outputs for comparative analysis.

The evaluation used a 4-point rubric assessing Accuracy, Completeness, Coherence, and Domain Appropriateness, with blind scoring to ensure objectivity. Findings showed GPT-5 and Gemini Pro consistently outperformed Gemma3:4b, both achieving an average score of 3.95, while Gemma3:4b followed closely with 3.64, demonstrating strong efficiency for its smaller scale.

Chain-of-Thought prompting emerged as the most effective for reasoning-heavy tasks, while CLEAR prompts excelled at precise, direct outputs.
The key recommendation is to leverage GPT-5 or Gemini Pro for production-grade workflows while reserving Gemma for rapid prototyping and cost-efficient experimentation.

Methodology

The evaluation aimed to measure how different LLMs and prompting strategies influence developer-focused workflows.
Three representative tasks were selected within the technology domain:

- Code Summarization - simplifying complex code into developer-friendly explanations.
- API Question Answering - responding accurately to developer queries using provided documentation.
- API Documentation Generation - creating structured, professional API docs from raw specifications.

Each task was approached with three prompting techniques:

- CLEAR: context-rich, structured instructions.
- Few-shot: demonstration-based prompting with sample outputs.
- Chain-of-Thought (CoT): step-by-step reasoning for complex outputs.

All prompts were tested across three models: GPT-5, Gemini Pro, and Gemma3:4b.
This produced nine outputs per task (3 prompt styles × 3 models), totaling 27 outputs for evaluation.

Outputs were randomized and scored blindly using a 4-point scale across four metrics:

- Accuracy, Completeness, Coherence, Domain Appropriateness & Language.
- A flag variable tracked hallucinations or bias.

Limitations:

- Small dataset limits statistical generalizability meaning the results represent more directional insights rather than definitive performance benchmarks.
- Some qualitative judgments were subjective, mitigated by blind evaluation and predefined rubrics.
- This structured, reproducible methodology ensured fair comparison across both models and prompting approaches.

Results

Key findings from the excel sheet:
GPT-5 and Gemini Pro were equally consistent, receiving an average score of 3.95
Gemma performed well as well, receiving an average of 3.64, considering it's a much smaller model.

Models	Average Score Across the test
GPT-5	3.95
Gemini 2.5 Pro	3.95
Gemma3:4b	3.64

GPT-5 and Gemini Pro were nearly tied for top performance, excelling in accuracy and coherence, particularly in reasoning-intensive tasks like API Documentation Generation.
Gemma3:4b, while smaller, demonstrated strong speed and efficiency, making it ideal for lightweight deployments, though it struggled with completeness, occasionally omitting key sections such as error handling notes.

Specific Notes for Models:

1. GPT-5
   Exceptional reasoning with minor verbosity issues.
   It performed really well across all the tasks but it struggled with code summarization with Chain-of-thought prompting for me. The response was almost identical to the previous task output, indicating limited adaptability to prompt variations. Thus, I even tried removing the line of wanting the response in 2-3 sentences, but it didn't change the response. Thus, I felt the response was fine, but lacked completeness as compared to the other models or as I would have hoped for. Apart form this, it performed really well with all the tasks and prompts.
   Also, for the same task, it took much more time with CoT prompt even though in the end, the results were almost identical to the other prompting techniques.

2. Gemini Pro
   Concise and accurate with slight creativity issues.
   Gemini struggled with the same issue where the response it gave was fine but with less information as compared to its counterparts in some aread. It struggled in Code summarization part as well. But it performed well in giving examples in Task-2 whereas Gemma didn't.

3. Gemma3
   Excellent speed and cost-efficiency and good trade-off since it is a smaller model.
   Gemma performed well as compared to the other models given that it was a much smaller model. It didn't hallucinate anywhere or had bias but some of the answers were not coherent, or complete. It worked really well in the Documentation task but lacked in the Code summarization and question-answering tasks, as compared to the other models.

Prompting Strategy Performance

- CLEAR prompts were good to get the concise and reliable answers. The drawback was that they were less reliable for the multi-step or dependency tasks.
- Few-Shot prompts produced highly structured responses. The drawback was that they relied heavily on the quality of the examples as a little change in the examples gave me much better outputs.
- Chain-of-thought prompts were good with great accuracy and reasoning but they were time taking or slower and probably more tokens inducing as well.

Patterns

- Gemma being the smallest model, it was consistently the fastest among the three models.
- I was also amused by the fact that Gemini and Gemma had quite similar reponses throughout, like it was noticeable that both the models are from the same parent organization in some way.

Bias and Hallucination

- No hallucinations or explicit bias were observed across outputs.
- This was likely due to the controlled dataset and task design, which minimized ambiguity.

Failure Cases

- GPT and Gemma gave the same responses in one of the prompts where I removed the limitation on the number of sentences. But Gemini adapted and gave a better and more detailed response to that.
- GPT seemed faster in giving the responses, but it looked like that because it streams the output where Gemini gives the output in blocks.
- Gemma was not able to provide examples multiple times where its counterparts did.
- Gemini, in general, was overly concise in some places.

Why?

GPT-5 excelled at reasoning but was verbose and slower because it has a larger context window and a built-in reasoning mode, meaning it would take the prompts and process the instructions step by step, which boosts the reasoning but increases the token usage and the response time.

Gemini Pro was slightly more concise because it is optimized for production use cases, prioritizing concise and high-level outputs. It might be using aggressive summarization heuristics, which can cause a slight loss of information.

Gemma was the fastest model as it has fewer parameters and lower computational overhead, so the outputs were generated quickly. But limited parameters meant lower reasoning depth and reduced ability to maintain the context over multi-step tasks.

Discussion

The evaluation revealed distinct patterns in how models and prompting techniques interact with the tasks at hand.

Model Trends:

GPT and Gemini, both demonstrated superior reasoning depth, especially in multi-step tasks. Gemma was highly efficient and offers a strong balance considering the size of the model. While it lacked at places where reasoning or thinking was needed, it still performed considerably well.

Prompting Techniques:

All the three techniques provide different and better results for different type of tasks.

For documentation, CLEAR prompts were good and can be done with much lighter model such as Gemma, but it would definitely depend on the size of the codebase, complexity of the codebase and many other factors.

Few-shot was great where I wanted the outputs in structured and specific manner.

Chain-of-thought worked well wherever I needed the reasoning quality but it comes at the cost of response time and token usage.

Context Sensitivity:

If I have to choose, I will have CLEAR and CoT for the Code Summarization and Question/Answering and will have CLEAR and Few-Shot for documentation purposes.

Failure Analysis:

While the models could have performed better at some places, this highlights the areas where post-processing and human validation are essential for deployment.

Limitation:

The sample size was too small and limits the statistical confidence, so these results represent qualitative trends rather than definitive conclusions.

Recommendation

For production grade and heavy workflows, GPT-5 and Gemini and recommended because of their superior capabilities in accuracy, coherence, and adaptability across the task types.

Both the models are great at complex reasoning tasks such as code explanation and technical documentation generation, Gemini being slightly more concise, would make it great for tasks or workflows where conciseness is needed.

Gemma would work great for rapid prototyping and cost-sesitive environments, where speed is prioritized over absolute completeness.

For prompting techniques, CoT and CLEAR work great together. CoT works great for reasoning-heavy workflows and Few-shot would be a great choice for well defined structured formats.

Combining the right model and prompting technique yields optimal performance while minimizing cost and error risk.

Appendices

1. Excel Evaluation Sheet
2. Technical Report and Appendix