

Testpassport**Q&A**



H i g h e r Q u a l i t y

B e t t e r S e r v i c e !

We offer free update service for one year
[Http://www.testpassport.com](http://www.testpassport.com)

Exam : CCAR-P

**Title : Claude Certified Architect -
Professional**

Version : DEMO

1.A production RAG system serves a corpus mixing exact legal citation lookups with conceptual paraphrase questions. After a corpus refresh, citation queries are confidently wrong while paraphrase queries stay fine. The team runs dense-only retrieval and tracks only final-answer accuracy.

Which combined remediation best addresses the diagnosis?

- A. Retrain the model on the refreshed corpus while keeping dense-only retrieval in place
- B. Rewrite the system prompt to demand citations and lower the sampling temperature for stability
- C. Add a sparse BM25 path fused with dense retrieval for exact citations, and track Recall@k to localize where failures occur
- D. Replace fixed-window chunking with semantic chunking and add more chunks to the final prompt

Answer: C

Explanation:

Confident-wrong citation answers after a refresh signal a retrieval gap: dense-only retrieval misses exact-term citations. Fusing a sparse BM25 path restores exact matching while preserving paraphrase coverage, and Recall@k localizes the regression to retrieval rather than the model or prompt.

2.A RAG assistant produces wrong answers, but the team only measures end-answer accuracy and cannot tell whether retrieval or generation is at fault. They want to localize the problem before making changes.

Which approach best isolates the failure stage?

- A. Compare two prompt versions on live traffic and pick the higher accuracy
- B. Increase the number of retrieved chunks until end-answer accuracy improves
- C. Replace the embedding model and re-measure the final answer accuracy afterward
- D. Measure retrieval metrics (Recall@k, MRR) separately from answer quality, so a retrieval miss can be told apart from a generation error

Answer: D

Explanation:

End-answer accuracy alone cannot separate retrieval failure from generation failure. Retrieval-specific diagnostics like Recall@k and MRR localize the problem to the retrieval stage before any change is made.

3.A Claude-backed service is being retired next quarter, and several downstream teams depend on it. What does a proper deprecation phase require?

- A. A silent shutoff once the replacement service reaches feature parity
- B. Immediate removal to force dependent teams onto the newer tooling
- C. Stakeholder notification and a documented migration path for dependents before the service is shut down
- D. Keeping the service running long past the retirement date to avoid disruption

Answer: C

Explanation:

Deprecation is the deliberate closing phase of the lifecycle: affected stakeholders are notified and a migration path is defined before the service is shut down. A silent or immediate shutoff breaks dependent teams and any commitments that rely on the service.

4.An extraction prompt produces inconsistent formatting. An engineer proposes adding forty near-

identical happy-path examples to the prompt. A colleague argues for a smaller change.

Which of the following BEST improves output consistency?

- A. Add the forty near-identical examples so the model repeatedly sees the same pattern
- B. Drop the examples and rely on a single longer instruction paragraph
- C. Raise the model size further until the formatting stabilizes on its own
- D. Use a handful of diverse examples that cover edge cases and output formats

Answer: D

Explanation:

A handful of diverse examples covering edge cases drives output consistency more than many near-identical happy-path ones; examples are the strongest single lever for steering format.

5. During discovery, stakeholders list what the system must do: classify tickets, draft replies, escalate edge cases. They also state it must answer within 300 milliseconds and stay available 99.5 percent of the month.

How should the architect categorize these last two targets?

- A. As functional requirements, since latency and uptime describe system behaviors.
- B. As acceptance criteria that remove the need for separately defined requirements.
- C. As non-functional requirements, quality attributes like latency and availability that often become the SLOs and SLA terms.
- D. As implementation details to defer until the design phase actually begins.

Answer: C

Explanation:

Latency and availability state how well the system must perform, so they are non-functional requirements. These quality attributes commonly become the SLOs the team tracks and the terms promised in an SLA.