hr | avatar
Illuminating Talent

Test Results and Interview Guide

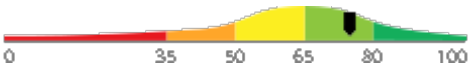
Candidate: **Elizabeth Wantsajob**
Assessment: Large Language Model (LLM)-Based Application Concepts
Completed: July 5, 2026
Prepared for: Sara Maple
Example Company

What's Included

- Overall Score
- Competency Summary Table
- Comparison Matrix
- Detailed Competency Results with Interview Guide

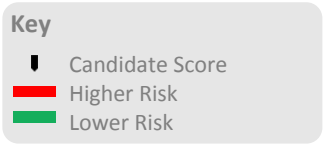
Important Note: The Large Language Model (LLM)-Based Application Concepts assessment measures one or more important competencies, and collects audio or video responses to specific questions. Attribute types measured vary by test, but can include cognitive ability, skills, knowledge, personality characteristics, emotional intelligence, and past behavioral history. Various types of analysis may be conducted on the recorded responses depending on the test configuration. Note that these results should always be used as a part of a balanced candidate selection process that includes independent evaluation steps, such as interviews and reference checks.

Overall

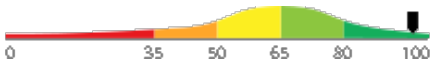
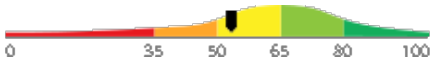
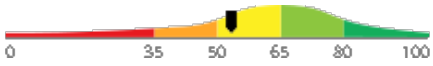
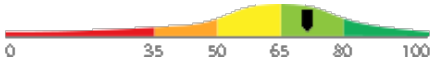
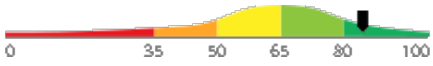
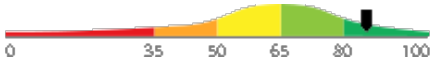
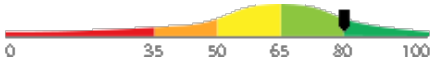
Candidate	Score	Interpretation
Elizabeth Wantsajob	75	

beth.wantsajob@gmail.com
Large Language Model (LLM)-Based Application Concepts
July 5, 2026

The candidate demonstrates a solid and well-rounded understanding of how to effectively integrate Large Language Models into business applications. Proficiency across most assessed areas — including API integration, tokenization, output handling, and cost management — is evident, with only minor gaps that would likely be addressed through practical experience or targeted development.



Competency Summary

Competency	Score	Interpretation
Skills/Knowledge (relates to immediate readiness)		
API Integration	96	
API Integration (Free Text Responses)	53	
Prompt Engineering (Free Text Responses)	53	
Context and Conversation Management	71	
Output Handling and Validation	84	
Prompt Engineering	85	
Tokenization and Cost Management	80	

Comparison

Percentile scores indicate how the candidate compares to other test-takers within various groups. The candidate scored equal to or better than the fraction of test-takers indicated by the percentile.

Test-Taker Group	Percentile
Global	75th
North America	62nd
United States	62nd
Example Company	69th

Artificial Intelligence (AI) Generated Scores

This table includes one or more scores derived from a large language model AI query. AI-derived scores are non-deterministic. That is, they are not precisely repeatable. Therefore, these scores should always be treated as supplementary information and should never be used exclusively or compared to hard cutoff values.

Estimated Value	Score	Confidence	Interpretation
Knowledge, Skills, and Abilities Summary	-	-	Summary Points (AI): • (Generic Text for Sample Report) Strong performer in Drag and Drop Files tasks, indicating comfort with file management and basic computer interactions. • Demonstrates solid numerical accuracy in Recognizing and Confirming Numbers, a valuable asset in detail-oriented roles. • Moderate overall performance in Analytical Thinking and Attention to Detail, with adequate grammar skills but room for improvement. • Struggles with Reading and Analyzing Problems, which may limit effectiveness in roles requiring critical reading and complex problem-solving. • Lowest performance in Navigating Between Screens, suggesting difficulty with multi-screen software workflows that could impact productivity in computer-intensive roles. Narrative (AI): Elizabeth Wantsajob demonstrates a mixed profile of knowledge, skills, and abilities across the assessed competencies. Elizabeth shows a strong aptitude in Drag and Drop Files, performing well on this technical task and suggesting she is comfortable with this type of computer interaction. This is a notable strength that would translate well into roles requiring file management and basic computer navigation tasks. In the area of Analytical Thinking and Attention to Detail, Elizabeth performs at a moderate level. She demonstrates solid ability in Recognizing and Confirming Numbers, which suggests she is careful and accurate when working with numerical data — a valuable skill in detail-oriented work environments. Her Grammar performance is adequate but leaves room for improvement, indicating she may occasionally make written communication errors. Her weakest area within this competency is Reading and Analyzing Problems, where she struggled to consistently interpret and work through written problem scenarios. This may impact her effectiveness in roles that require critical reading, written comprehension, or complex problem-solving. Elizabeth's most significant area for development is Navigating Between Screens, where she scored considerably lower than the other competencies. This suggests she may have difficulty efficiently moving through software interfaces or multi-screen workflows, which could slow productivity in roles that rely heavily on navigating computer applications or data entry systems. Overall, Elizabeth brings some useful technical strengths, particularly in file management and numerical accuracy, but would benefit from targeted development in software navigation and analytical problem-solving to be fully effective in roles that demand these skills. Computed on: April 2, 2026, 11:09:49PM EDT

Detail

Candidate:	Elizabeth Wantsajob, beth.wantsajob@gmail.com
Assessment:	Large Language Model (LLM)-Based Application Concepts
Authorized:	July 5, 2026, by Sara Maple, Example Company, qamailsaram.mike@hravatar.com
Started:	July 5, 2026, 5:59:18PM EDT
Completed:	July 5, 2026, 5:59:18PM EDT
Overall Score:	75

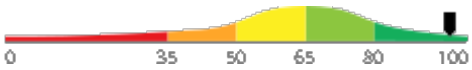
Knowledge and Skills Detail

This section contains a list of job-related knowledge areas and skills that have been evaluated. Low scores in these areas often indicate that additional learning may be required before top performance can be achieved.

Detail | Interview Guide

API Integration

Score: 96



Description:

API integration covers how to connect an application to an LLM service by making calls to its endpoints, passing inputs, and handling the responses returned. This includes understanding how to authenticate requests, structure API calls, and process the data returned by the LLM. This skill is essential for embedding LLM capabilities into software applications.

Interpretation:

Candidate should achieve superior job performance in this area with little or no training.

The candidate exhibits a high level of proficiency in API integration for LLM-based applications, reflecting comprehensive knowledge of endpoint interaction, request authentication, and response handling. They are well-equipped to independently design, implement, and optimize LLM integrations within software applications.

Describe how you would integrate an LLM API into a business application, including how you would handle authentication, structure the request, and process the response for use within the application.

1	2	3	4	5
Provides only a vague description with little detail about authentication, request structure, or response handling.		Covers most key steps with reasonable detail but may omit one area such as error handling or response parsing.		Gives a complete, detailed description covering authentication, request structure, response parsing, and error handling.

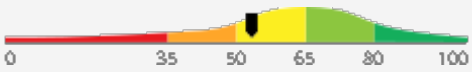
Can you explain what an API is and describe at a basic level how you would use one to send a message to an LLM and receive a response?

1	2	3	4	5
Cannot explain what an API is or cannot describe any part of sending and receiving a request.		Explains what an API is and describes the general idea of sending a request and getting a response.		Clearly explains APIs, describes request structure, authentication, and how to handle the response.

Detail

API Integration (Free Text Responses)

Score: 53



Description:

Covers the end-to-end process of planning, building, testing, and deploying AI-enabled applications for both internal staff and external customers. Includes managing iteration cycles, versioning, model monitoring, and coordinating cross-functional teams through each phase of the product lifecycle.

Interpretation:

The candidate exhibits average writing skills, which can hinder high performance in some jobs.

The candidate possesses a moderate understanding of AI product management, demonstrating basic familiarity with lifecycle management, strategic assessment, and process orchestration, though proficiency across these areas is inconsistent. With targeted coaching and hands-on experience, this individual has the potential to develop into a capable contributor in managing AI-enabled application initiatives.

Overall AI Score:	60.0
High words per minute detected while composing one or more essays:	27.3 words per minute. Possible copy/paste or use of AI tools. Average WPM while composing is about 15.
AI Confidence Level:	80
Argument Strength (AI):	70.0
Clarity and Coherence (AI):	80.0
Match with Ideal Response (AI):	60.0
Other Errors per 100 Words:	0.0
Spelling errors per 100 words:	0.0

Please see below to view the essay submitted.

Interview Guide

Describe a time you managed or contributed to an AI product through multiple lifecycle stages. What were the most significant challenges you encountered between phases, and how did you address them?

1	2	3	4	5
Candidate provides a generic or superficial example that lacks detail about AI-specific lifecycle challenges. Does not clearly articulate their personal role or the decisions they made between phases.		Candidate shares a relevant example with reasonable detail, identifying at least one meaningful challenge such as stakeholder alignment or testing delays. However, the response may lack specificity about how AI-related factors (e.g., model performance, data readiness) influenced lifecycle decisions.		Candidate provides a detailed, concrete example that demonstrates ownership across multiple lifecycle phases. Clearly describes AI-specific challenges such as model validation failures, shifting requirements, or deployment infrastructure issues, and articulates the specific actions they took to resolve them and keep the product on track.

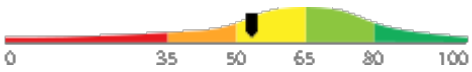
Can you walk me through the basic stages you would follow to take an AI-enabled product from an initial idea to a live deployment?

1	2	3	4	5
Candidate provides a vague or incomplete description of the lifecycle, omitting key phases such as testing, validation, or deployment. May conflate AI product development with general software development without acknowledging AI-specific considerations like model training or data pipelines.		Candidate identifies the major phases (discovery, development, testing, deployment) and acknowledges some AI-specific considerations, but struggles to articulate how the phases connect or how cross-functional teams are coordinated throughout.		Candidate clearly outlines a structured lifecycle including discovery, requirements, development, model validation, testing, deployment, and monitoring. Demonstrates awareness of AI-specific challenges such as data quality, model drift, and iterative retraining, and explains how they would coordinate stakeholders across phases.

Detail

Prompt Engineering (Free Text Responses)

Score: 53



Description:

Covers the end-to-end process of planning, building, testing, and deploying AI-enabled applications for both internal staff and external customers. Includes managing iteration cycles, versioning, model monitoring, and coordinating cross-functional teams through each phase of the product lifecycle.

Interpretation:

The candidate exhibits average writing skills, which can hinder high performance in some jobs.

The candidate possesses a moderate understanding of AI product management, demonstrating basic familiarity with lifecycle management, strategic assessment, and process orchestration, though proficiency across these areas is inconsistent. With targeted coaching and hands-on experience, this individual has the potential to develop into a capable contributor in managing AI-enabled application initiatives.

Overall AI Score:	60.0
High words per minute detected while composing one or more essays:	27.3 words per minute. Possible copy/paste or use of AI tools. Average WPM while composing is about 15.
AI Confidence Level:	80
Argument Strength (AI):	70.0
Clarity and Coherence (AI):	80.0
Match with Ideal Response (AI):	60.0
Other Errors per 100 Words:	0.0
Spelling errors per 100 words:	0.0

Please see below to view the essay submitted.

Interview Guide

Describe a time you managed or contributed to an AI product through multiple lifecycle stages. What were the most significant challenges you encountered between phases, and how did you address them?

1	2	3	4	5
Candidate provides a generic or superficial example that lacks detail about AI-specific lifecycle challenges. Does not clearly articulate their personal role or the decisions they made between phases.		Candidate shares a relevant example with reasonable detail, identifying at least one meaningful challenge such as stakeholder alignment or testing delays. However, the response may lack specificity about how AI-related factors (e.g., model performance, data readiness) influenced lifecycle decisions.		Candidate provides a detailed, concrete example that demonstrates ownership across multiple lifecycle phases. Clearly describes AI-specific challenges such as model validation failures, shifting requirements, or deployment infrastructure issues, and articulates the specific actions they took to resolve them and keep the product on track.

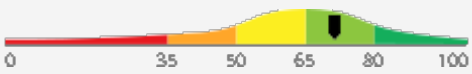
Can you walk me through the basic stages you would follow to take an AI-enabled product from an initial idea to a live deployment?

1	2	3	4	5
Candidate provides a vague or incomplete description of the lifecycle, omitting key phases such as testing, validation, or deployment. May conflate AI product development with general software development without acknowledging AI-specific considerations like model training or data pipelines.		Candidate identifies the major phases (discovery, development, testing, deployment) and acknowledges some AI-specific considerations, but struggles to articulate how the phases connect or how cross-functional teams are coordinated throughout.		Candidate clearly outlines a structured lifecycle including discovery, requirements, development, model validation, testing, deployment, and monitoring. Demonstrates awareness of AI-specific challenges such as data quality, model drift, and iterative retraining, and explains how they would coordinate stakeholders across phases.

Detail

Context and Conversation Management

Score: 71



Description:

Context and conversation management refers to how conversation history is tracked and passed to an LLM so that it can produce coherent, relevant responses across multiple exchanges. Because LLMs do not retain memory between calls, applications must manage and supply prior messages as part of each request. Understanding how to do this effectively, while staying within token limits, is critical for building useful conversational applications.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate demonstrates a solid understanding of context and conversation management, including how to track and pass conversation history to an LLM to maintain coherent, multi-turn interactions. They are likely capable of building functional conversational applications while managing token constraints with reasonable effectiveness.

Interview Guide

How would you manage conversation history in a multi-turn chat application built on an LLM, and what challenges would you need to address as conversations grow longer?

1	2	3	4	5
Gives a vague answer without addressing statelessness, token limits, or strategies for managing history length.		Describes passing history with each request and mentions token limits as a concern.		Describes a clear strategy for managing history, addresses token limits, and proposes solutions such as summarization or truncation.

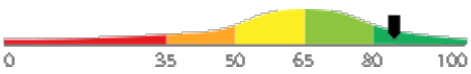
Why do you think it might be important to include previous messages when sending a new question to an LLM in a chat application?

1	2	3	4	5
Cannot explain why prior messages matter or gives an incorrect or irrelevant answer.		Explains that the LLM needs prior messages to understand the conversation but lacks detail.		Explains that LLMs are stateless, describes how history is passed in requests, and notes token limit trade-offs.

Detail

Output Handling and Validation

Score: 84



Description:

Output handling and validation involves checking and processing the text or data returned by an LLM before it is used in a business workflow. This includes detecting hallucinations, formatting issues, or incorrect content, and applying strategies to ensure the output is reliable and safe to use. This skill is important for maintaining the quality and trustworthiness of LLM-powered features.

Interpretation:

Candidate should achieve superior job performance in this area with little or no training.

The candidate exhibits an advanced and comprehensive mastery of output handling and validation concepts for LLM-powered applications. They are highly skilled at detecting and mitigating hallucinations, formatting issues, and unreliable content, and can design and implement robust validation strategies that ensure high output quality and safety across complex business workflows.

Interview Guide

How would you validate and handle the output from an LLM in a business application to make sure it is accurate and safe to use before presenting it to users or passing it to another system?

1	2	3	4	5
Provides only a general answer with no specific validation steps or techniques mentioned.		Describes at least one or two validation steps, such as checking format or flagging uncertain responses.		Describes a multi-step validation approach including format checks, confidence signals, human review triggers, and fallback handling.

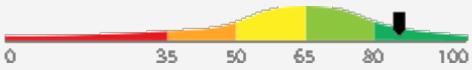
What is a hallucination in the context of an LLM, and why could it be a problem in a business application?

1	2	3	4	5
Cannot define hallucination or does not connect it to any real business risk.		Defines hallucination as incorrect or made-up output and notes it could mislead users.		Defines hallucination clearly, gives a concrete business example, and describes how it could be detected or mitigated.

Detail

Prompt Engineering

Score: 85



Description:

Prompt engineering involves designing and refining the text inputs sent to an LLM to produce outputs that meet business needs. This includes structuring instructions clearly, providing relevant context, and using techniques such as few-shot examples to guide model behavior. It is one of the most frequently applied skills when working with LLMs in business applications.

Interpretation:

Candidate should achieve superior job performance in this area with little or no training.

The candidate demonstrates a comprehensive and sophisticated understanding of how to integrate and utilize Large Language Models within business applications. They are expected to be highly proficient across all assessed areas, including advanced prompt engineering, API integration, token management, output validation, retrieval-augmented generation, cost and security considerations, model selection, and the identification and mitigation of hallucinations and other output failures.

Interview Guide

Walk me through how you would design a prompt for a business task, such as summarizing customer feedback, and what techniques you would use to improve the consistency and quality of the output.

1	2	3	4	5
Describes a vague or unstructured prompt with no mention of techniques or quality considerations.		Describes a reasonable prompt with at least one technique, such as adding instructions or examples.		Describes a well-structured prompt using multiple techniques and explains how each improves output quality.

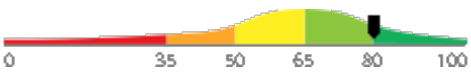
Can you describe what a prompt is and give an example of how you might change a prompt to get a better response from an LLM?

1	2	3	4	5
Cannot define a prompt or gives a vague, unhelpful example with no reasoning.		Defines a prompt correctly and gives a basic example of improving it.		Clearly defines prompts, gives a specific example, and explains why the change improves output.

Detail	Interview Guide

Tokenization and Cost Management

Score: 80



Description:

Tokenization refers to how LLMs break text into smaller units called tokens, which are the basis for both processing limits and usage costs when calling an LLM API. Understanding how tokens work helps developers write prompts and manage inputs and outputs in ways that stay within model limits and control expenses. This knowledge is important for building LLM applications that are both functional and cost-effective.

Interpretation:

Candidate should achieve superior job performance in this area with little or no training.

The candidate exhibits a comprehensive and advanced understanding of tokenization and cost management within LLM-based applications. They are well-equipped to independently design efficient, cost-effective solutions that optimize token usage, manage API expenses, and operate effectively within model processing limits across a wide range of application scenarios.

How would you approach managing token usage in an LLM-based application to keep costs reasonable while still meeting the needs of the business?

1	2	3	4	5
Gives a vague answer without identifying specific strategies for reducing or managing token usage.		Identifies one or two strategies such as shortening prompts or limiting response length.		Describes multiple strategies such as prompt optimization, response length limits, caching, and monitoring usage, with clear reasoning.

Can you explain what a token is in the context of an LLM and why the number of tokens in a request might matter?

1	2	3	4	5
Cannot explain what a token is or does not connect tokens to limits or costs.		Explains that tokens are units of text and that there are limits or costs associated with them.		Explains tokens clearly, connects them to both context limits and API costs, and notes practical implications for application design.

Free Text Responses

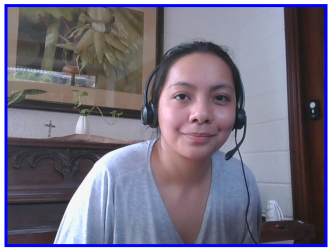
During the assessment, the candidate was asked to answer one or more questions using text, audio, video, or an uploaded text file. Their responses are included below for review.

Question or Task	Response
After an AI product is deployed, what is model monitoring and why is it a necessary part of the product lifecycle?	Model monitoring is a technique for ensuring that the model does not wander or become overtrained after an extended period of repeated queries that have the same or similar prompts. This is very important for preventing hallucinization. It's also a key aspect of any guardrails strategy. **Comments (AI):** The answer is clear and coherent but lacks depth in explaining the importance of model monitoring. The phrase 'hallucinization' is not commonly used in this context and may confuse readers. The answer could be improved by providing more specific examples of model performance metrics and how they are tracked. The argument strength is moderate as it does not fully explain why model monitoring is necessary in the product lifecycle. **Misspelled Words:** guardrails (1), hallucinization (1)

Identity Confirmation Photos

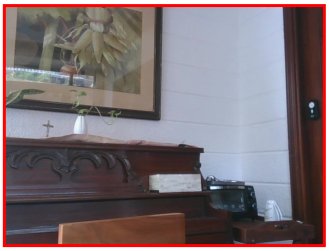
The following photos of the candidate and any identification were uploaded during the assessment session.

Photo Analysis Results	
- Risk:	Medium risk of cheating based on image inconsistencies
- Percent match among processed faces	100%
- Total images processed	17
- Total images with valid faces	14 (82%)
- Total pairs of faces compared	13
- Pairs in which faces matched	13 (100%)

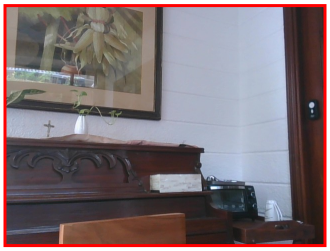


Pre/Post-Test Photo

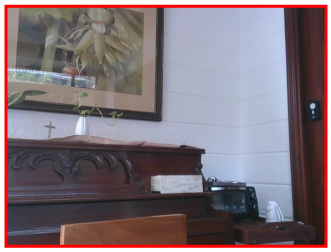
ID Photo



In-Test Error Detected (No Face Detected)

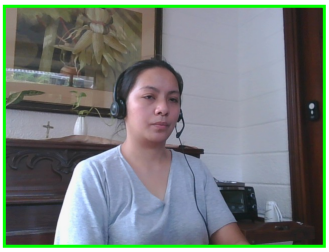


In-Test Error Detected (No Face Detected)

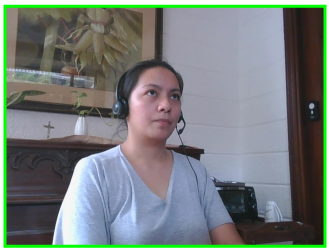


In-Test Error Detected (No Face Detected)

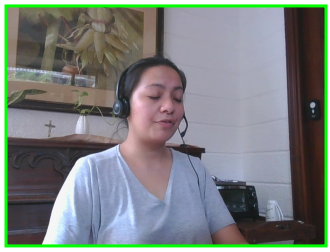
In-Test Photo



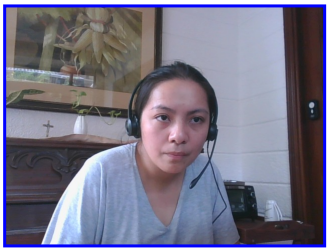
In-Test Photo



In-Test Photo



In-Test Photo



Pre/Post-Test Photo

Resume or CV

Summary Updated on

Motivated career professional with extensive experience in office administration and management. Proven track record of improving efficiency, reducing costs, and enhancing office operations through strategic initiatives and technology implementation.

Objective

I am seeking a role where I can use my many skills and my exceptional judgment and empathy for customers to make a difference to a growing company.

Education

- Associate of Applied Science in Office Administration, Portland Community College, 2020

Experience

- General Office Clerk, Paramount Office Management, 09/2023 – Present
- Administrative Assistant, Global Enterprises Inc., 04/2021 – 08/2023
- Administrative Assistant, Innovative Business Solutions Ltd., 07/2019 – 03/2021

Other Qualifications

- Microsoft Office Specialist (MOS) Certification
- Certified Administrative Professional (CAP)
- International Association of Administrative Professionals (IAAP) Certification

Report Preparation Notes

- Hiring decisions should never be based on a single source of information. The most effective use of this assessment report is as a part of a multi-faceted program of candidate evaluation that includes resume review, interviews, and reference checks.
- Overall vs Percentiles Scores: The overall score reflects the success in the test, based on the mean (average) and standard deviation of the test scores. The percentile score reflects the percentage of test-takers who scored equal or below this overall score. We recommend you use the Overall Score as your primary evaluation criteria. However, percentile scores can often be useful in comparing specific candidates against one another and with a group, such as for test takers in a certain organization or within a certain account.
- Note that comparison information is calculated based on completed instances of this assessment at that time the assessment is scored. As additional instances are completed, the comparative data may change. You can always update a report to the current values by clicking on 'Recalculate Percentiles' within the online results viewing pages at www.hravatar.com.
- Most competency scores are norm-based, which means that they can be interpreted in terms of their distance from the average or mean score. For all scales, a score equal to the mean receives a score of 65 and scores above and below this value are set so that a score change of 15 equals one standard deviation.
- For linear competencies, higher is better across the entire scale. For these scales a score between 65 and 80 (light green) represents 0 to 1 standard deviation above the mean and a score above 80 (dark green) represents more than one standard deviation above the mean. Similarly, a score of 50 - 65 (yellow) represents 0 to 1 standard deviation below the mean, while a score of 35 - 50 (orange) equates to 1 to 2 standard deviations below the mean, and a score below 35 represents more than 2 standard deviations below the mean.
- Sim ID: 20918-1, Key: 0-0, Rpt: 104, Prd: 9740, Created: 2026-07-05 17:59 EDT
- UA: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; Touch; rv:11.0) like Gecko

Score Calculation Detail

The following table provides a summary of how the overall score was calculated from each of the individual competency scores. First, all competency scores are calculated on a scale of 0-100. Note that some competencies use their color category rather than their actual numeric score in the overall calculation. For these, a standard score associated with the assigned color category is used in the overall score calculation rather than the actual numeric score. This is reflected in the ''Score Value Used'' column. Next, a weighted average of scores is computed using individual competency weights, typically set using job analysis data provided by the US Government Occupational Information Network (O*Net).

Competency	Score	How applied to overall	Score Value Used	Weight (%)
API Integration	96.5843	Numeric Score	96.5843	14.2857
API Integration (Free Text Responses)	53.8624	Numeric Score	53.8624	14.2857
Context and Conversation Management	71.7909	Numeric Score	71.7909	14.2857
Output Handling and Validation	84.9021	Numeric Score	84.9021	14.2857
Prompt Engineering	85.9239	Numeric Score	85.9239	14.2857
Prompt Engineering (Free Text Responses)	53.8624	Numeric Score	53.8624	14.2857
Tokenization and Cost Management	80.3767	Numeric Score	80.3767	14.2857
Weighted Average:				75.3289
Final Overall Score:				75

Notes

(This area is intentionally blank - it's reserved as space for your notes.)