Cursor SWE Agent Problem Reviewing Rubric

Reviewers use this rubric to evaluate responses. Any task scoring a 1-2 on any category will be SBQ’d, and they should be approved. **Your overall rating of a task should be based on the lowest category score a task receives.**

Please also refer to the Common Errors section for common issues that warrant a task failure.

Deliverable Categories

When evaluating task quality, reviewers will evaluate the quality of two problem deliverables plus the overall structure of the broader GitHub issue/PR the contributor is decomposing:

- **Prompt**
  - Reviewers will evaluate the quality of the input prompts the user writes to ensure they are providing sufficient detail and solving the correct problem.
- **Rubrics**
  - Reviewers will evaluate the thoroughness and specificity of the rubrics, ensuring that they follow the rubrics' best practices and that there is no overlap between the different criteria.
  - Note that the code and prompt rubric should be considered as one whole

Evaluation Rubric

Type	Field	1-2 (Fail)	3 (Okay)	4-5 (Good/ Perfect)	Explanation
Question Primary Category	**Prompt**	Question Category selected is incorrect	N/A	Question Category selected is appropriate	

Naturalness (Prompt)	**Prompt**	The prompt is unreasonable and is highly unlikely to be asked by a SWE using Cursor in the real world	N/A	Prompt feels natural; could be reasonably asked by a SWE working on a problem	Prompts should ask reasonable questions that a SWE working on a project might ask to help them better understand the repo or problem. If the prompt feels overly contrived or like something that
Self Contained & Solvable (Prompt)	**Prompt**	The prompt requires external information to solve	N/A	The prompt is self contained and can be answered without referencing external sources	All information needed to solve the prompt should be contained within the repo or should be reasonable context for an LLM to know. A prompt should not require internet search or reference of external sources to answer the question
Correctness (File Mapping)	**Prompt**	The file mapping contains at least one error	N/A	The file mapping is correct	The file mapping maps the files referenced in the prompt (e.g., @foo.py) to the full file path (e.g., packages/modules/foo.py)
Classification Correctness	**Rubric**	2 or more rubrics are not correctly classified, or any rubric labeled as a critical failure is not actually a critical failure	1 rubric item is not correctly classified	All rubric items are correctly classified	Every rubric is labeled according to whether it is critical (must have) or non-critical (nice to have). The Claude Sonnet response is then evaluated against the rubric to determine if the response passes or fails the rubric. Your job is to evaluate whether the rubrics are correctly classified along these axes.

Completeness	**Rubric**	Rubric misses any critical items that a good response must include	N/A	Rubric is fully complete and provides a good means of assessing a good response to the prompt. All potential critical items are captured.	Consider **all** the elements you would want to include to create a perfect response and put them into the rubric. This means including not only the facts and statements directly requested by the prompt, but also the supporting details that provide justification, reasoning, and logic for your response. Each of these elements should have a criterion because each criterion helps to develop the answer to the question from a slightly different angle.
No Overlapping	**Rubric**	There are 3 or more pairs of rubric items that overlap with each other	There are 1-2 pairs of rubric items that assess overlapping areas	No rubric items overlap	The same error from a model shouldn’t be punished multiple times.
Difficulty - Overall	**Rubric**	Rubric items are very easy and it 80%+ of the rubric items are passed by the model	Rubric items are somewhat difficult and reference solution passes 60-80% of rubric items	Rubric includes many difficult/aspirational items and the reference solution passes under 60% of rubric items	A good rule of thumb is that the model fails on 50% of rubrics items, otherwise the task might be too easy. Good rubrics should include “aspirational” items that current best models do not include

Difficulty - Critical Failures	**Rubric**	- There are no critical rubric items where the reference solution fails	- There is one critical rubric item where the reference answer fails	There are two or more critical rubric items where the reference answer fails	CBs must classify each rubric by whether the reference answer passes or fails on it and whether the rubric is critical or “nice to have” for a good response. A passing rubric must have at least one critical failure.
Difficulty - Critical Failures Mis labeling	**Rubric**	If a CB labels a rubric as “critical failure” but it should be non-critical, this is also an auto fail.	N/A	Critical Failure is labeled correctly	
Atomicity	**Rubric**	2+ rubric items or any critical items are not atomic	1 rubric item is not atomic	All rubric items are atomic	Each rubric criterion should evaluate exactly one distinct aspect. Avoid bundling multiple criteria into a single rubric. Most stacked criteria with the word “and” can be broken up into multiple pieces.
Specificity	**Rubric**	2+ rubric items or any critical rubric items are not specific	1 rubric item is not atomic	All rubric items are objectively true/false	Criteria should be binary (true or false) and objective.

Self Contained	**Rubric**	2+ rubric items or any critical rubric items are not self contained	1 rubric item is not self-contained	All rubric items are self-contained	Each criterion should contain all the information needed to evaluate a response, e.g. ❌ Mentions the capital city of Canada. ✅ Mentions the capital city of Canada is Ottawa. Criterion should be verifiable without requiring external search. Without having to search externally we should be able to answer the rubric question
Diversity	**Rubric**	The rubric items are highly repetitive with less than 3 distinct categories overall	Rubric items are moderately diverse but have some repetition. There are at least three distinct "types" of rubric items	Rubric items are diverse and assess responses across a variety of areas	The rubric items should include variable types of information. If all criteria are like "the response mentions A", "the response mentions B", then this is not a good rubric.
Volume	**Rubric**	There are less than 10 total rubric items OR there are less than 2 "process rubrics"	N/A	There are 10+ rubric items and there are 2+ "process rubrics"	Rubric items are counted in sum between the code and commentary rubrics A process rubric is a rubric item about the process an agent follows to reach an answer (e.g., grepping the codebase, reading file `foo.py`
Rubric Supporting Context	**Rubric Supporting Context**	3+ rubric supporting context items are missing or incorrect	1-2 rubric supporting context items are missing or incorrect	All rubric supporting context items are correct	Evaluate that the rubric supporting context is correct and accurate to the commit diff or other sources of GTFA