

Acceptance Criteria for Acceptance Criteria Checklist



Jobs To Be Done By Acceptance Criteria

- Specify functional and non-functional requirements.
- Determine conditions of satisfaction.
- Define boundaries of a user story/product feature.
- Clarify the expected outcome.
- State the story's/feature's intent, but not its solution.

The Acceptance Criteria Checklist

UNDERSTANDABLE BY ALL STAKEHOLDERS



- Communicate intent rather than technical requirements or solutions.
- Pro tip:** AC that defines technical requirements risk communicating a solution, instead of intent.
- Avoid business or technical jargon.
- Reserve the backlog's non-functional requirements for technical user stories and AC.

WRITTEN IN FIRST PERSON, ACTIVE VOICE



- Write AC in the voice of the person drawing value from the story.
- Use active voice, in which the user is doing the action described in the AC.
- Communicate the value that the user will gain by doing the action described in the AC.
- Consider techniques that steer your user in the direction of the solution without actually stating the implementation. **Example:** "I can search for a name."

WRITTEN USING CONCISE, PROSAIC SENTENCES



- Brevity is the key to good AC. Keep it simple.
- Omit unnecessary words. Make definite assertions.
- Use the words the user would use.
- Avoid redundancy or repetition.



INDEPENDENT OF IMPLEMENTATION

- AC is evergreen; write AC so the user receives value regardless of whether the implementation is made.
- Avoid writing AC that is prescriptive; doing so constrains creativity and innovation.



STAY IN THE PROBLEM SPACE, NOT THE SOLUTION SPACE

- Write AC in the Problem Space. Focus on the problem to be solved, the job to be done.
- Write AC that points to a specific user problem, need, or benefit that the product should address.
- Pro tip:** AC written in the Solution Space describes a specific implementation that might not address the customer need or requirement.