

Twelve questions for a software or AI proposal

For the person in the room who has to decide

You do not need to understand the technology to evaluate the proposal. You need to understand the work it claims to improve, and whether the person proposing it does.

Ask these in order. The first four are about the problem, and a vendor who cannot answer them has not done the work — whatever the demo looked like.

About the problem

☐ **What task does this change, who currently does it, and how long does it take them?**

A weak answer talks about capabilities rather than tasks. If nobody can name the task, nobody has measured it, and there is no way to tell afterwards whether it worked.

☐ **Where does that task currently go wrong, and how often?**

If the honest answer is "it does not, it is just slow", that is fine — but it changes what success looks like and usually what it is worth.

☐ **What happens today when the output is wrong?**

This is the question that decides how much autonomy the system can have. Ask it before anything else about the technology.

☐ **Would fixing the process remove the need for this?**

Frequently yes. Automating a broken process makes it fail faster and more consistently.

About the proposal

☐ **What will it NOT do?**

The single most useful question in the list. A vendor who cannot name a limitation is either not being straight with you or does not know the product.

☐ **What does it need from us that we do not currently produce?**

Clean data, a decision nobody has made, a process that exists in one person's head. This is where projects actually stall.

☐ **Who maintains it, and what happens when they leave?**

Applies to the vendor as much as to you.

☐ **What does the second year cost?**

First-year pricing is a sales decision. The second year is the real number.

About afterwards

☐ **How would we know in six months whether this worked?**

If the answer is not a number you are already measuring, agree one before you sign.

☐ **What is our exit? Can we get our data out, in a form something else can read?**

Ask for a sample export, not an assurance.

☐ **Who inside our organization will understand this well enough to change it?**

If the answer is nobody, you have bought a dependency, not a capability. Sometimes that is an acceptable trade — but make it deliberately.

☐ **What is the smallest version of this we could try first?**

A vendor who will not scope a small first step is telling you something about their confidence.

If a proposal survives all twelve, it is probably a good proposal. If it falls over on questions one to four, the problem is not the technology — and no amount of evaluating the technology will find that out.

Glover Works LLC - Navigate Complexity. Build What Works.

Business and organizational consulting, education and training, safety and compliance, technology and systems.

gloverworks.com - hello@gloverworks.com

Use this freely inside your own organization. It is a starting point, not professional advice for your specific situation.