



How to hire a software development firm

A BUYER'S GUIDE

Written by

Jon Berbaum
President, Highland Solutions

312-546-3494 (direct)

jberbaum@highlandsolutions.com

[linkedin.com/in/jonberbaum](https://www.linkedin.com/in/jonberbaum)

[@jonberbaum](#)



Highland has been building custom software for over twenty years.

During that time, we've worked on over a hundred custom software projects. For the vast majority of projects we worked on, we were a good fit for the client, accomplishing the work we set out to do.

Some of the time, we were an excellent fit – with a highly capable team who also aligned with the client's culture and vision – and were able to create a huge amount of value through our work.

A few times, we've been a bad fit – working outside of our comfort zone or poorly matched to the client's values and work approach.

In those bad fit projects, some predictable things happened. Either the work got done and everyone was miserable doing it. Or the client fired us. And in a few rare cases, we fired the client.

While my experience is primarily on the vendor side of a custom software purchase (not the client side), I've learned to evaluate each of our potential clients rigorously. Yes, the client is spending money. But everyone is investing their time, energy, and reputation into this work.

So we need to ask ourselves honestly:

Should this company hire us?

Will we be successful together?





PART 1

FINDING FIT

1. TRUST

Even with the most highly-skilled teams, developing custom software will never go entirely as planned. Anyone who tells you otherwise is lying.

Good organizations always want to work with people they trust, and the often unpredictable nature of software development makes trust particularly important. Suffice it to say that not every development firm is committed to doing right by their clients, and not every client is committed to taking care of their development firm.

Before selecting a development firm, take time specifically to talk about trust. This can happen when only a few potential firms remain, though you should get a sense throughout the evaluation process.

A few questions you can ask to help determine trustworthiness:

- **Ask about values.** What are they? How do they live them out?
- **Ask about the times when unexpected things happened and how they handled them.** Is there a time the project lead went on parental leave during an intense build? Or when a third-party dependency blew up? Or when a product concept wasn't working?
- **Ask about the times the budget got incredibly tight** and how they handled hard conversations.



My experience is that the level of vulnerability offered in response to these questions is a good indicator of trustworthiness. If there are no failure stories and hard-worn learnings, the development firm is either doing very simple work or maintaining an image.

And before a development firm works with you, they should be asking you these same kinds of questions.

If they don't ask, take the initiative. Talk about your values and how they show up. Talk about when you've worked with digital firms before and how it went. (Expect a good development firm to get extra curious about any stories of "bad" development firms because sometimes the problem is the previous firm and sometimes it's not.) Talk about how your organization and leaders have responded in the past to big initiatives or challenging circumstances.



2. LEVEL OF UNCERTAINTY

Level of uncertainty is a gauge of how certain your organization is on what needs to be built.

A simple low, medium, high uncertainty ladder can illuminate the significant differences here. The levels have no value: low is no better than high. But different levels require different strengths from a development firm.

In my experience, most clients evaluate development firms based on process, quality, and reputation, but have no category for thinking through level of uncertainty. Yet getting this wrong can doom a project.



LEVELS OF UNCERTAINTY, EXPLAINED



Low Uncertainty

Execution-Oriented

What it looks like:

Developing a back-office application to replace a legacy system is low uncertainty. What will be built is already known at a fairly high level of detail.

An execution-oriented development firm that can deliver the defined application efficiently and with high quality is the best fit.

Medium Uncertainty

User-Oriented

What it looks like:

Developing a next-generation customer platform for an existing business is mid-level uncertainty. What will be built is largely known, but there is room for improvement and innovation.

An execution-oriented development firm could deliver a working application but will miss many opportunities along the way.

Building the right thing requires interviewing and observing customers beforehand, regular user testing throughout, and some emergence of features and user experience throughout.

This level of uncertainty is best suited to a firm that has not only development strength but also UX and Product Management capabilities.

High Uncertainty

Product-Oriented

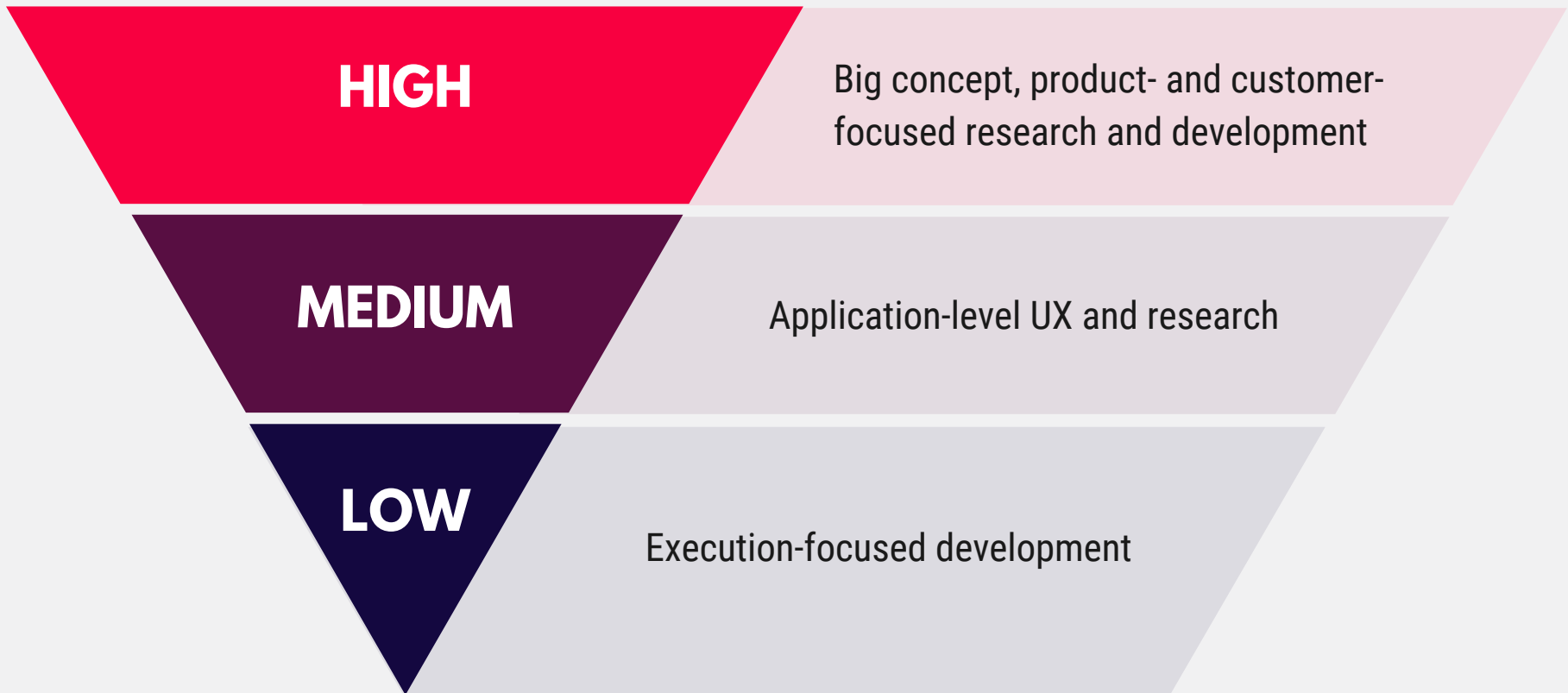
What it looks like:

Developing a digital product for a new business or new customers, or transforming an entire digital experience involves high levels of uncertainty.

The organizational strategy and goals are known, but exactly how to get there is not. Building the right thing requires deeper levels of upfront research and strategy, exploratory design and testing, and iterative design and development execution.

This level of uncertainty is best suited to a firm that has strategy, research, design and development strengths and is accustomed to working in high uncertainty initiatives.

LEVELS OF CUSTOM SOFTWARE UNCERTAINTY



3. PRICE

The price for software development should be related to the value being created, the level of uncertainty the firm is operating at, and the quality of the firm's process and talent.

Your organization should ask hard questions about price and value. Development firms should also be asking questions about price, well beyond "What is your budget?".

They should also ask "What does success look like for this product?" and understand the risks and rewards in play for your organization, both in dollars and other measurables.

Sometimes the value being created for a buying organization isn't sufficient to justify the cost of building custom software. Development firms are in the best position to identify this disconnect and help you consider more feasible alternatives. Anything less is malpractice.



RATES

While not all software is billed at an hourly rate, most pricing can be decomposed into an hourly rate. At the time of writing this (late 2019), the rough ranges for normal hourly rates at US-based boutique firms are:

FIRM TYPES X HOURLY RATES	LOW	HIGH
EXECUTION-FOCUSED FIRMS	\$100	\$150
USER-FOCUSED FIRMS	\$140	\$200
PRODUCT/CUSTOMER- FOCUSED FIRMS	\$180	\$225

Offshore firms working in the US are basically always execution-focused. Their hourly rates are under \$100 per hour. Larger firms are typically \$200-\$300 per hour and are a mixed bag in terms of focus and price to value ratio.



PRICING MODELS

While hourly rate prices are the norm in development, there are other pricing models. Your organization and the development firm should agree on the contractual model they are using and why.

The **time and material model** is most common by far, presented as either hourly, weekly, or monthly rates, both per person and as a collective team. There are several strengths to this model: cost is transparent throughout and roughly corresponds to effort, and your organization owns both the reward and risk of building software and executing the strategy that software supports.

There are a few weaknesses as well: inefficient development teams are rewarded with additional revenue, and breakthrough creative contributions from the development firm go unrewarded. That being said, this model roughly – though imperfectly – aligns the buying organization and the development firm.



PRICING MODELS (CON'T)

First, **fixed-price contracts**, where the development firm promises to deliver an application for a specific price. The strength of this model is its ability to control costs for the buying organization and reward the development firm for efficiency.

The weakness – which almost always overwhelms the strength – is that the development firm is incentivized to complete as little work as possible. Fixed price contracts encourage rigidity in features and low quality work. The horror stories of work in this pricing model abound and should be avoided by both parties for anything but the very simplest software projects.

Second, a variety of **partnered, value-based, or outcome-based models**. These can be quite creative and include the buying organization offering equity, future revenue, an outcome-based bonus, or other incentives with the development firm.

These models appear when startup organizations are buying software, or when the buying organization and development firm are seeking long-term alignment because of real synergy.

These models can be powerful when done well, as they distribute both risk and reward between the organizations. Like all true partnerships, they should be used only where there is a significant amount of trust and synergy between the organizations.





PART 2

EVALUATING SKILL

4. REPUTATION

High levels of reputation indicate quality work and people, plus some amount of mastery, generosity, and curiosity.

The typical way to evaluate reputation is to talk to references. Two or three reference conversations are standard, and the details of the specific work in a previous engagement is much less important than the experience of the previous clients in working with the firm.

It's assumed that the development firm will provide references who are happy with their work, but even the ability to produce happy clients is a basic test.

A few hours searching and reading online can greatly expand your ability to evaluate a firm's reputation.

I would suggest focusing on these areas to get a good sense of a firm's reputation:

- Do they write, speak, and teach in their areas of practice? Most established firms have a reputation among other practitioners and something of a focus within their craft. Look at a survey of their **thought leadership** and activity within the practitioner community.
- Find and read through listings on software development review sites, starting with **Clutch**.
- Read through the last few years of reviews on **Glassdoor**.
- Run a few web searches for **key leaders** to see where they are writing and speaking outside of the firm. This will provide a good sense of the reputation and contribution of the firm.



5. PROCESS

Maturity of process indicates experience and expertise, as well as level of talent. The question you want to answer here is:

How does the development firm perform the work?

Basically every firm is “agile” these days, so you’ll want to dig deeper below the surface, particularly on firms that practice Scrum. While Scrum teams can be excellent, the unfortunate reality is that inexperienced practitioners in the custom software space use Scrum, as it can be a framework where it is easy to hide under-skilled team members and lack of good process.

A Kanban or hybrid process is more often an indicator of a mature development process.

Your organization should also ask how design, user experience, and user testing fit into the development process, as well as how a development project is governed for features, timeline, and cost.

Ask upfront how often you will meet for project management purposes, what those conversations will include, and how budget, actuals, and projections will be calculated and displayed.

There is a goldilocks zone in project management: too little or too much can indicate a less mature process and a less skilled team.



6. QUALITY

Finally, ask explicitly about how the development firm approaches quality in software.

Frankly, this can be difficult to assess, since you can't look directly at their handiwork, like a carpenter or a different kind of craftsperson.

However, here are a few questions that can help you determine the quality of a firm's approach to custom software, even if you don't consider yourself a technical expert:

- **Does the firm use automated testing? A baseline expectation is that developers are writing unit tests (at least) and using automated tooling to run the test suite anytime code is merged.**
- **What is the internal process for code reviews?**
- **Are there coding standards?**
- **What other quality processes are built into the development flow?**



Quality gets its own evaluation apart from the general process because far too many development firms operate with a lack of quality.

Low quality makes software difficult and expensive to maintain and ultimately shortens its life. Over the years, more than one organization has come to us with an application that didn't even make it to launch day because of the severe lack of quality from the development firm.

Use reference calls to talk about quality, particularly if you're able to speak with clients who have been using and maintaining an application built by the development firm years ago.





SUMMARY

THE 6 CRITERIA TO EVALUATE

The six main criteria when hiring a software development firm are:

1. Find a mutual fit at the right level of uncertainty.
2. Find a team and leaders you can trust.
3. Find a mutual fit in price and contract type, appropriate to the level of uncertainty.
4. Evaluate reputation.
5. Evaluate maturity of development processes.
6. Evaluate quality processes.

This approach will help you avoid immature or low-quality firms, and avoid bad fits with good firms who simply aren't built for your project and organization.

The good news is there are a ton of really great development firms out there, so if you commit to a mutual, eyes-open process, you can build amazing things together.



thank you



jberbaum@highlandsolutions.com | [linkedin.com/in/jonberbaum](https://www.linkedin.com/in/jonberbaum)