

The **MOST** complete Guide to Finding the **Best Developer for your App.**

This is our promise!
Let our expert Guide Lead You Through the Hiring Process.



2 Bonus **GIFTS** included:

Automated Developer Evaluation Matrix.

Page 16

Score every developer and hire the best one!

Developers Comparison Template.

Page 16

Most important decision-making information organized in one place

Is this just another guide doubling as a marketing brochure?

Nope. If you do an internet search on "How To Hire a Software Developer?" There's lots of informational marketing material disguised as a guide put out by software development agencies. JetRuby doesn't want to be one of those.

We have lots of experience hiring and firing other developers and other development firms. We have worked with over 100 companies over the past 11 years - and before that, we were a startup. So, finding the right developer or firm for a project has been part of our business. And now, we want to share our wisdom on that process with you.

Why are we sharing this guide with you?

Because we hope you use it on our team when you consider us for a project. Please send our score, and any feedback to business@jetruby.com

Follow Our Process

We promise. It'll be good.

- 1 Read the guide.
- 2 Contact developers, put their info into the Google spreadsheet provided.
→ [Comparison Template](#)
- 3 Use the matrix, ask the questions, and calculate the score.
→ [Scoring Matrix](#)
- 4 Choose the developer with the highest score.
- 5 Build your app, release it, and grow your business.

Table of Contents

Technology Stack Overview	4
Current Development Technologies	6
Freelancers Versus Development Agencies: The Pros and Cons	9
Developers: How and Where do I find one?	10
How to tell if you are hiring a professional developer?	12
Bonus GIFTS	16

Technology Stack Overview

What's a stack?

Well, we aren't talking about pancakes. So let's get that out of the way first.

A "stack" refers to a technology stack which is the set of technologies a developer will use to build something. If you are reading this article, we assume you want to develop a mobile, desktop, web, or cross-platform app. There are different sets of technologies for developing each one of those specific types of application.

The Back-end and the Front-end

Technology stacks are made up of the front-end and the back-end segments of a technology stack. The front-end part of a stack is for developing the part of an application that humans interact with, like the webpage or an application's Graphical User Interface (GUI). Developers use back-end technologies for creating the parts of the application that communicate and transfer data between the client - the computer the application is installed on - and the server - the computer that stores data on a database for the application to retrieve remotely.

Libraries and Frameworks

If you drill down one lower level of detail of what comprises a tech stack, it's typically composed of frameworks and libraries. Both frameworks and libraries are reusable code written by someone else that developers use to build and create functionality for an application.

A framework is the skeleton of an application similar to the framing in a house. There are different frameworks for different kinds of applications. A framework makes creating the application easier because the developer doesn't have to start from scratch. Using a framework also creates consistency in the code which makes testing the application easier and the security of the code better.

A library is a collection of code syntax used in building specific functionality or parts of an application. Just like a framework is like the framing of a house, the library is like a hardware or furniture store you go to and buy things to put in or add to the house—that way, the homeowner doesn't have to build those things from scratch.

→ Technology Stack Overview

Why is it important to know the technology stack a developer is going to be using?

It's essential to look at the technologies and languages the developer or agency you want to hire uses. Every tech stack has its strengths and weaknesses. Some are geared toward building mobile apps or web applications, and others are geared toward data science applications. What stack you decide to use in your project can also determine how easy it will be to upgrade and maintain in the future and scale your digital product as your business grows.

Current Development Technologies

Checking out industry reports such as the [Stack Overflow Developer Survey](#) is an excellent place to understand what technologies and languages are currently being used in application development.

Here are some of the most popular technology stacks used by developers today in 2021. Of course, these are just brief overviews of the technologies, and we encourage you to do more in-depth research if one of these stacks looks like a good fit for your project.

Ruby On Rails Stack

Ruby On Rails (RoR) is a rapid development web application framework for quickly going from an initial idea to a Minimum Viable Product (MVP) web application. Since RoR has a host of ready-made, out-of-the-box, plug-ins and modules, the framework is very time efficient, consistent, secure, scalable, and cost-effective. Customers can expect to save 30-40% in development costs using RoR. RoR is flexible too. Developers can use RoR along with other front-end frameworks like Angular.JS or React.JS. With its own built-in testing suite, RoR promotes test and behavioral-driven development. Finally, because of the default security features already built into the stack, RoR makes following the industry-standard Secure Development Lifecycle security assurance process easy to follow for developers.

MEAN Stack

The MEAN stack is one of the best stacks for developers to use on projects that involve cloud hosting, calendars, mapping, and content aggregation sites. The stack consists of MongoDB, Express.JS, Angular.JS, and NODE.JS. The stack is open source, so it's free and has a well-supported developer community. The MEAN Stack is a one-language, end-to-end solution. Meaning, a developer only uses one language for all the different technologies in the stack. That also makes the stack very efficient because it's easier for developers to reuse code. In addition, the database is easily scalable to respond to the growth and success of your new app.

MEVN Stack

This cousin to the MEAN stack uses Vue.js instead of Angular.JS for a front-end framework. Vue.js offers a basic, out-of-the-box, lightweight functionality that can use third-party services to extend its capabilities. In addition, Vue.js combines the best aspects of Angular.js and React.js to offer developers a complete toolbox of options.

MERN Stack

Another closely related stack to the MEAN stack, the MERN stack, replaces Angular.JS with React.JS. Developers use React.JS more for creating single, one-page applications with more elaborate User Interfaces (UIs). The downside to using React is that it doesn't have a lot of core functionality, and therefore developers have to rely on more third-party services. However, React.JS is known for making changes in the code easier and is a better tool for seamlessly connecting to other application components.

Serverless Stacks

A serverless stack provides the tools and a set of technologies that allow a developer to create cloud-based applications. Cloud-based apps generally make scaling an application much more effortless because all of the work that involves managing and configuring the server infrastructure for your app is handled by the cloud provider.

Serverless stacks include Google Cloud Platform, AWS Lambda, IBM Cloud Functions, Azure Stack, Cloudflare Workers, and Computer@Edge. The other thing to consider when working with a serverless technology stack is that most are proprietary. They have licensing fees, and you must pay for the amount of computing time your app uses by the cloud provider.

LAMP, WAMP, MAMP Stack

LAMP is the old workhorse of the software development world. It is a reliable and stable set of technologies that is still very popular today. It consists of the Linux (Windows, or MAC) Operating Systems, Apache, MySQL, and PHP.

Since it's open-source, it's free, and you can more easily tailor the components in the stack to your business needs. PHP can be switched out for other options like Python or PERL.

Flutter for Web

Flutter for Web is a new contender to the web development space and a competitor to React Native. Flutter is ideally suited for the development of cross-platform applications. In addition, it makes for easy deployment so developers can focus on fixing bugs and adding features.

Freelancers Versus Development Agencies: The Pros and Cons

Use the table to evaluate which path is best for your project.

When to use an agency:	When to use a freelancer:
<p>You have a new idea for an app but no details on the scope or deliverables.</p>	<p>You have a shorter-term project (10-20 days) with a clear scope and well-defined deliverables.</p>
<p>You have a long-term project that requires a scalable team.</p>	<p>You have a project that requires minor changes within existing software that has quality documentation and version control.</p>
<p>You have a project that requires a Minimum Viable Product (MVP) development stage.</p>	<p>You want to develop an interactive prototype for your project.</p>
<p>You have a complex project that requires a team of specialists.</p>	<p>You have a project that requires no or low-code development tools like the website builders Wix, Squarespace, and Weebly, or like the app builders Air-Table, Bubble, and Adalo.</p>

Developers: How and Where do I find one?

This goes without almost needing to be said. First, ask your friends and colleagues if they know anyone. Personal recommendations from your trusted inner circle are the best way to find another trusted business partner. But, if you've run out of people to ask, here's a table of some of the more popular resources to help you in your search for your next developer.

Resource	Best For Finding Freelancers	Best for Finding an Agency
Clutch.co		●
BusinessOfApps		●
Goodfirms.co		●
DesignRush		●
Upwork	●	
Fiverr	●	
Toptal	●	
Wired		●
GitHub	●	●
Hashnode	●	
Dribbble	●	●
Behance	●	●
People Per Hour	●	
Guru	●	●

INSIDER TIP:

These websites usually have robust vetting and ranking systems in place for their candidates. However, it is common practice for these websites to offer "sponsorships", which allow companies to buy a place higher in the sites' company rankings. The "sponsorships" can go for as much as \$25,000 a month, and their cost is usually passed along to the customer in some form or another.

So, being open to considering agencies on these sites ranked in the 20th - 40th rankings can save you money without compromising on your project's quality.

Freelancer Websites

Upwork, Fiverr, Toptal, PeoplePerHour, and Guru provide transparent prices. Most freelance sites offer or do a thorough vetting of their candidates. Toptal only accepts 3% of applicants.

Tech Blogs

Blogs like TechCrunch can be a great place to read articles about software development companies that you want to consider hiring. Other blogs like Techmeme and ReadWrite can also be good places to read up on current trends and technology in the software development industry. A great place to look for tech blogs are blog aggregators like FeedSpot. There is also Hashnode - a blog aggregator for software developers. There you can find blog posts about various industry topics by software developers.

Portfolio Websites

These types of sites include Github, Dribbble, and Behance. Many of the freelance sites overlap as portfolio websites as well. For some of these sites, you may need to pay to join to view freelance portfolios. Github is the industry standard for project repositories. If you are considering a freelance software developer and they do not have an account profile on Github that should be a red flag. Most working software developers do. Behance and Dribbble mostly cater to web and UX designers.

How to tell if you are hiring a professional developer?

Professional

- The developer has a substantial, well-organized portfolio that they readily offer to you.
- The developer's website and any other content they provide to the public is well organized and the content explains technical concepts and their business practices in an easily understandable way.
- The developer can provide references. They have no dead links in their portfolio or public information. Their contact information is readily available.

Unprofessional

- The developer doesn't offer a portfolio or they don't have a large body of work in their portfolio.
- The developer's public information, like their website, either lacks the necessary information or is too confusing for you to assess if they would be suitable for your project. The technical information they provide is also difficult for you to understand.
- The developer doesn't provide any references. There are dead links in their portfolio or public information. Their contact information is not provided, or it isn't easy to get a hold of them.

- The developer asks about your business goals before seeking to understand your project's technical specifications.
- They take some time to estimate the amount of work required to complete the project before providing you with an estimate.

- The developer seems hyperfocused on your project's technical specifications without trying to understand your business goals.
- The developer quickly offers an estimate for your project without spending any time learning about the project's work or your business goals.

- The developer considers your customers and how the project will impact them now and in the future.

- The developer is more focused on coding than how their work will influence your business or customers.

Professional**Unprofessional**

- The developer focuses on the project's scope and the work required to meet your app's business requirements.

- The developer is very rigid about the tasks to complete and will not deviate from them.

- The developer is patient and takes time to educate you about any aspects of the project you don't understand. Then, you notice yourself starting to ask questions about your project that you didn't think of before.

- You feel like you are the one who has to take the initiative to communicate with the developer, and you don't feel any connection with them.

- The developer takes the time to gently engage you about software development and how the trajectory of a project works. They also make an effort to adapt to your needs.

- The developer takes a rigid stance even on minor issues that you bring up.

- The developer requests your project specifications and offers to help you create them if you do not have them already.

- The developer doesn't work with specifications.

- The developer asks for a small retainer to begin your project and correlates future compensation to when they complete project milestones.

- The developer wants the entire payment upfront before starting the project.
- They also change their estimate after starting the project.
- They do not offer to stagger their payments in correlation to meeting project milestones.

Professional	Unprofessional
<ul style="list-style-type: none"> You and the developer create a clear understanding of their role, project responsibilities, and project deliverables. 	<ul style="list-style-type: none"> The developer does not offer to engage you in creating a clear plan for contributing to your project. You are confused about their role, their responsibilities to the project, and their project deliverables.

Skill Testing Developers Before Hiring

It's one thing to write a good resume but the proper validation in someone's skill set is their ability to follow through and do what they say they can do.

Here is a list of programming skill check sites to test candidates you are considering for a project:

- HackerRank
- ChallengeRocket
- Codility
- Codeassess
- Coderbyte
- Coderpad
- Qualified
- CodeinGame

Other Vetting Techniques

- Hosting a Hackathon
- Paired Programming Interview

Legal Checklist

It probably goes without saying to make sure you have a good lawyer look over any business agreement before you sign on a dotted line or give anyone any money. That being said, make sure you ask and look over the items on this checklist before doing business with any developers:

- Have you looked at all of the software company's legal documentation?
- Does your development team need a certified specialist to meet any industry legal standards for your application? Does your team have any of these specialists on their teams?
- Can you find and review the company's hiring and management practices? Do they have employees reviews and human capital reports publicly available?
- Are there any IP issues in the software company's contracts?
- Does the developer's Service Agreement cover and define: poor performance, any unforeseen fees or expenses, data loss, intellectual property, confidentiality, and non-competition?
- Does the company have any history of litigation?

Bonus GIFTS

① → [Developer Evaluation Matrix](#)

Google Sheets with Auto Calculated Scoring Matrix

Score	Outcome
0 - 26	Run Away
27 - 52	Consider
53 - 78	Hire

A list of questions we created to evaluate developers. Each answer has a score between 0-3 points, the maximum is 78. The score is calculated automatically based on the marked answers.

② → [Developers Comparison Template](#)

Google spreadsheet for Developers Comparison

The template provides you with space to take notes on the developer's scores, project estimates - or anything else for that matter.

Thank you for reading THE MOST complete Guide to Finding the Best Developer for your App.

We always appreciate any feedback about this guide. Please feel free to contact us and let us know how this guide worked for you.

✉ business@jetruby.com

☎ +1 415-508-7706

🌐 jetruby.com

The JetRuby team wishes you good luck with your project.