

How can Arkency help you?

When we join your team, our mission is to **help you grow the business into the next level**. Not only we want to understand the application and its code, but we mainly want to understand the business itself and how **you can use our unique skills to make it more profitable**, bigger and more automated. That's why businesses with millions of dollars of yearly gross income trust us.

Our experienced developers always have your business in mind. They don't just deliver code and finish tasks. They also help you refine the vision for upcoming features and changes. They are always looking if and how the requirements can be slightly changed to reduce the scope of a task, finish it faster and bring 80% of the value with 20% of the time. That is if you agree with the proposed direction that could save you time and budget.

Who's on our team? The people behind one of the most popular Ruby on Rails blog and the authors of 8 programming books regarding Rails, React.js, DDD, and Async/Remote cooperation. We are not ninjas, and gurus. But, we are experienced and mindful developers with a wide skillset. We can easily work on frontend and backend tasks. That gives us the ability to understand the whole application faster and introduce changes on all necessary layers. Which means we can ship whole features instead of disconnected tasks. We love to work towards business objectives.



Q & A

How am I going to know what you work on today?

We practice a technique called over-communication which **keeps you always in the communication loop**. Every day, we share with you via email or chat: [what we did](#), how we managed to do it, what obstacles we met and the progress towards the business objective. That means you are informed and don't need to waste your time on micro-managing our developers.

How can we keep deadlines together?

We actively help you prioritize the most important tasks and to keep them at the top of the backlog. All the extracted sub-tasks are also prioritized and done in the proper order starting from most important and bringing the most value to the client. We strive to ship our work to production on daily and sometimes even hourly basis. That means **when it's done, it's really done and you or your customers can use it**. We often rely on techniques such as continuous deployment and feature toggles to make that happen.

How is the first week of our cooperation going to look like?

You hire us to work on your idea, on your application. To implement features, smash bugs and to help you achieve your goals. From the very first day, we are going to try to do exactly that. We take a task you decided to be the most important, we try to figure out the code, we ask questions and hopefully, on day one, we commit changes which bring us closer to finishing the task. That's our goal when we kick-start our cooperation: **to ship code and close tasks on the first day**. The next days are going to be similar. In fact, that's how we work every day: take the most important task (if there is some doubt, we try to help by asking questions and bringing more clarity), split it into smaller units, finish it, ship it to production, rinse and repeat.

How can we organize the cooperation between you and my current team?

We are going to use all the tools that you and your team are already using for project management, keeping track of tasks, progress, priorities or daily communication. That way all our progress, status updates, discussions will be fully visible and transparent to you and your team. They can see for themselves how we work, observe our practices, learn our methods and explain to us the missing gaps, fill in the blanks. The ultimate goal is to blend into a single team and drop the *us vs they* distinction, it's only *we*.

Is my team compatible with your practices?

When we join a project, **we start by following the practices that are already established in the code**. We want to start smoothly. We don't come in to tell you to re-write everything.

Over-time we will evaluate the code based on our experience and we may start suggesting changes that we believe will benefit all parties. Whenever we suggest any bigger changes to the code or the process, we will provide links to books, articles or videos that explain the idea behind our proposed direction.

Are you going to leave me for a better-paying client or your own product after 4 months?

That's very **unlikely**. We are proud of maintaining years-long lasting and fruitful relationships with our clients. We don't just jump between projects on every possible occasion. We understand that knowing your business is going to take us time and we are willing to spend that time on learning the people, the process, and the code. Leaving your project after a few

months would be a waste of the invested time and effort on both sides. So we prefer a stable, long-term cooperation. **Some of our clients have been with us for 7 years.**

What if our code-base is a mess?

We can handle it. It's going to be harder for us but, **we can handle it.** Whenever we try to implement a new feature, handle a new requirement or change the code in any way, we will try to improve it a bit as well. We follow the *Boy Scouts* principle: *Always leave the campground cleaner than you found it.*

How can I be sure I get your best developers only?

Our developers have 5-10 years experience in Rails application. We simply don't hire juniors because we specialize in handling complex Rails applications. We help established, successful projects scale into bigger companies.

What's the best way to proceed now?

If you are interested in cooperating with us, please [schedule a meeting with Andrzej, our CEO](#). You can ask any questions regarding our skills, practices, experience, techniques, availability and financial aspects.

