

Approach to the development of startups on the example of cooperation with the startup "SureTask"

For startup founders without a technical background the process of developing a startup is not something transparent. They can hardly comprehend the development of complex web services or mobile applications if they haven't done it before. Besides, most developers gave different approach to development process, and even if you the corresponding experience in the past, it is not a fact that you were satisfied with both the process and the result. In this article I describe all stages of development and approach to cooperation on the example of cooperation with one of my customers. By the time you finish reading this article, you will have a comprehensive picture of how the technical implementation of a startup works and you will be able to make an informed decision on whether this approach suits you. Even if it turns out that you do not like this approach, you will be able to clearly define your requirements for the development process and find a really matching contractor.

Here I tell you about one of the latest customised projects I have designed. SureTask (<https://suretask.com>) is a platform for the UK market that allows the customers and contractors to find each other. Let's say your faucet is broken. You can go to SureTask, post a "Faucet Broken" ad, provide a description and wait for the performers to respond to your ad. If you are a provider of services, you can find a work on SureTask. The executors are charged for each response.

Negotiations

The customer wrote to me after giving read my article on a technical resource, in which I announced my personal project. Two young men acted as the customer. One of them lives in Estonia and the other one- in England. The customer verbally outlined the idea of the project, clarified the approximate time frame and the cost.

Considering that the technical assignment was not provided, and the description of the idea sounded very superficial -"We want to create an analogue of YouDo (a similar service operating in the CIS countries) for England", I could only give an approximate timeframe, estimating the development time for the first version at about 6 months. We agreed that the customer would pay not for the finished product, but on a monthly basis, and we would cooperate until we decide to stop the development. According to the plan, I was supposed to accomplish the initial stages on my own, and then, when the development tasks could be parallelized- to attract a second developer.

It is possible to evaluate the development of a product in its entirety only when there is a rendered design of all screens, because without it there exist a huge number of possible implementations, and all of them require expense of different amounts of resources. Besides, by estimation of the cost of development for the entire product, I, as a contractor, must include risks in the cost the possible risks, which in essence will increase the price,

provided that everything goes smoothly and the risks would not happen. Therefore, we were satisfied with the idea of monthly payment for the work performed.

The conceived product already had competitors operating in the British market. The concept of the product did not imply any fundamental differences from the products of competitors. In general, the launch of such product it is quite a risky idea.

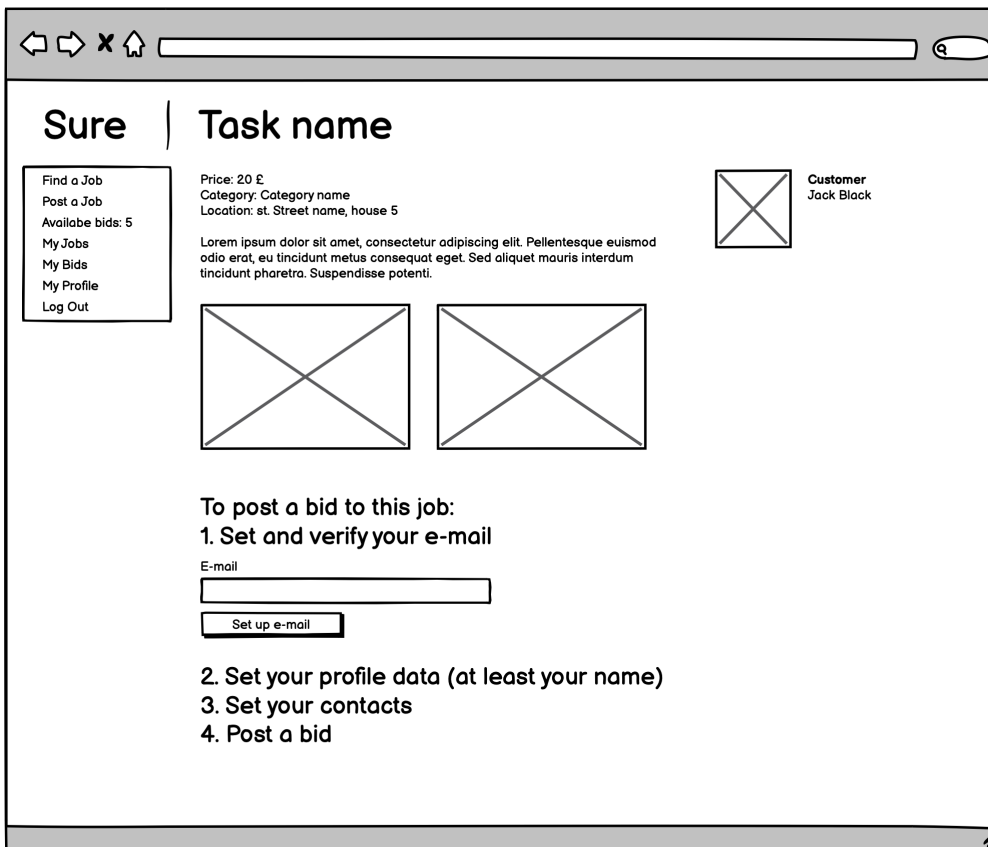
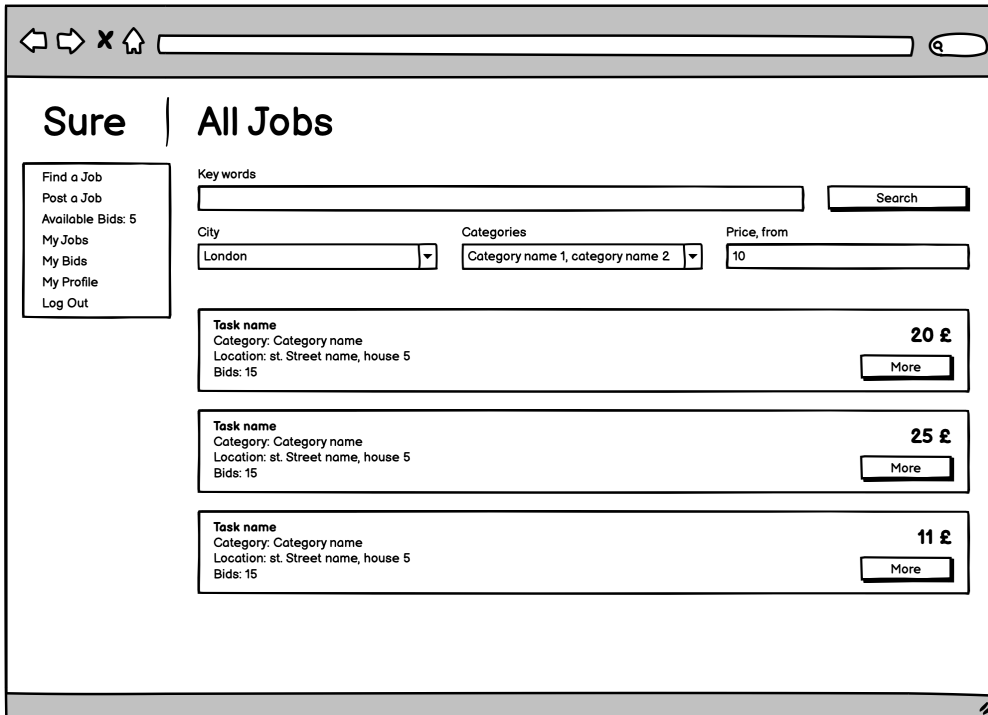
The decision on the admissibility of certain risks lies entirely on the shoulders of the founders. I have my own IT projects, where I bear these risks on my own. In the projects of customers I act as CTO, and I take care of the technical implementation and usability of the product, working closely with designers and trying to make the product as user friendly as possible. I completely trust the customer in making of business decisions, with the expectation that I am collaborating with a person who understands what he is doing. I will also be glad in the course of cooperation to learn from the customer his approach to doing business. If the customer is interested in adopting my experience in launching of projects, I would willingly share it and will help in making of decisions. However, consulting on launching of startups is not my principle service, but rather an attractive bonus, attached to mutually advantageous cooperation.

Thus, we agreed on areas of responsibility and the terms of cooperation, shook hands and got to work.

Start of cooperation

In order to start development one needs a clear understanding of what the final result should look like. However, we only had the following superficial description: "We want to create an analogue of YouDo for England." On the one hand, I already had an idea of what should come out in the end based on this formulation. It is logical that the customer had this understanding either. However, there was no guarantee that we represented the end result in the same way. Usually the developers require detailed technical assignment before start of works. I do not do that. I understand that compilation of a high-quality technical specification is a very complex and voluminous task, difficult for a non-technical founder of a startup. Instead, I suggested drawing up a graphical prototype of the future service in order to align our vision.

In the first week of cooperation, I created a prototype in Balsamiq Mockups, which suited all of us, and which could be taken into development. The prototype is not about how the future service will look like, but about how it will work. Here are some sample screens:



After accomplishment of prototype I immediately started developing the web service. I also prefer to immediately hand over the prototype to designer, whom I usually involve in the project from among the verified performers. However, the customer wanted to involve the designer independently, and it happened that due to internal reasons we decided to do it later.

Collaboration process before launch

In the first weeks of development, the architecture of the future service was created and the basic features were implemented, such as authorization and editing of a personal profile. I used the following technology stack: Node.js, React, MongoDB, GraphQL. I realized projects using various technologies, and subsequently came to the conclusion that this stack of technologies is best suited for quick development and comfortable scaling and maintaining the code in the future. I also have a lot of practices on this stack of technologies that I reuse in new projects, which also allows me to create the products faster.

Every week in the same day at the same time I demonstrated the achieved results of work and agreed on plan of activities for the next week. After the demonstration of works the customer had a link to the current version of the product so that he could dive deeper into operation of the web service at any time, or show it to future users, investors or someone else.

We handled all project tasks in Trello. We had a column called "Backlog", where all the tasks of the project were detailed and prioritized. The tasks in the backlog were also divided into groups of tasks, where each task group was supposed to be completed in one week. Thus, at each moment of time we had an idea of how much time was left before the launch of the first version. It should be noted that more features were thought out than needed to launch the first version. We planned to do all the necessary tasks for the launch, launch the project, and then continue to roll in the features into the already operating product.

The list of tasks by itself was formed based on how the graphic prototype. However, during the development of the product and the actual interaction with the resulting interface, we made decisions to either add something that was not originally included in the prototype, or to do something from the planned not as it was planned at the beginning. That is, we allowed ourselves to deviate from the original plan, and it was convenient. After all, from the very beginning it is impossible to think over everything for sure, you can only set the right vector. At the same time, the architecture of the code was arranged in such a way that we could painlessly add and change functionality without redoing everything from scratch.

At the beginning of work, it is difficult to parallelize anything, because each subsequent task depends on the previous one, so at first I worked alone. A little later, I took on a second developer, a full-stack one just like me, in order to speed up the development process.

We attracted a designer after the first version was almost ready. The designer was attracted by the customer. Previously, this designer created the design of sales pages for the main business of the customer, and the customer was satisfied with it.

However, design of a web-service is a much more subtle task, requiring several other skills than those needed to design the sales pages. As a result, the designer simply "painted" the elements of the developed web service, without paying attention to micro-interactions and to UX as such. The service began to look decent, but this work could have been done better. After this experience, I try to insist that, firstly, a designer would be involved at the very beginning of works, and secondly, it would be a designer from among the UX designers that I have previously verified.

After completion of design we implemented it on the service within about a month and then we were ready to launch the first version of the product.

Collaboration process after the launch

After launch of the project we began to receive direct feedback from users. Accordingly, now our plan consisted not only of what we had initially planned, but of the ideas that came to us when we saw how users actually use our service. I also installed a special analytics system that allowed us to track errors that occur in the service in case of some exceptional situations by our users. This allowed us to detect and fix errors that we could not identify during an independent check of the service's performance. There were not many errors, and this should be strived for, but it is impossible to make a product in which there will be no errors at all, you just need to be able to fix them thoroughly.

Besides, after the launch we decided to remove the second developer from the team in order to transfer the saved money to the marketing.

When the functionality of the web service settled down, we started developing a mobile application. Here I used the same technology stack, but ReactNative was used instead of React. This made it possible to reuse the code of web version, and, accordingly, reduce the duration of development. ReactNative also allows writing one single code for creation of both an iOS and an Android application. However, the peculiarity of the English market is that 86% of users there use iPhones, so we decided to release only the iOS application. As per the rest, the approach to work has not changed- I completed the tasks every week, demonstrated them, and agreed on a plan for the next week.

Final result

At some point in time the customer decided to stop the development of the product, because we had done everything that, in his opinion, needed to be done. Thus we decided that if something else would be required, then we will evaluate it separately and finish it privately.

During the cooperation we created a web service and a mobile application. The customers were utterly satisfied with the cooperation with me, as I met all their expectations and delivered a quality work in a timely manner. The product still exists and keeps growing.

In general, during this cooperation everything went well, but I still made a few conclusions for myself. Firstly, I will no longer let the choice of designer take its course and I will either insist on the involvement of specialists I have verified, or I will more carefully estimate the designer attracted by the client. Secondly, I realized that I was not comfortable working alone, except for the first month of work, when the architecture is built, because there was simply no one to rely on, no one to discuss the technical solutions with, and the development itself runs much slower.

One of the customers after our cooperation ordered me the development of another his startup. I offered another client to become a partner in one of my personal projects because of his business skills. He agreed and we are still working together on our common project.

If you have any questions about how the technical implementation of a startup is implemented, or you are interested to know whether there will be any differences in the development approach for your particular project, write me by the contacts stated below or choose a convenient time for you to call in the Calendly service:

<https://calendly.com/iserdmi/30min> During the conversation we will discuss your project, I will answer all your questions, and at the end of the call I will offer a specific plan for the technical implementation of your project.

Also I have prepared for you the answers to 9 questions that you need to ask yourself before starting developing a startup:

1. How should one choose a technology stack?
2. Should you hire a full-stack developer or a front-end and a back-end engineers in separate?
3. Should you you work with Agile or Waterfall?
4. How much should you pay the developers?
5. Is it worth hiring a developer for a share in the project?
6. How can you verify the competence of developer?
7. What should you provide to developer before start of works?
8. How can you control the developers?
9. How will the startup development process be structured?

To receive a file with questions and answers, add me as a friend on Facebook

<https://www.facebook.com/iserdmi> or LinkedIn <https://linkedin.com/in/iserdmi> and write the code phrase "9 questions" in the private messages.

Contacts:

<https://10.com.ru> — on my website you can find out more about me and my team, learn about the benefits of working with us, look through the examples of work, the terms of cooperation, the time frames and the prices for development
iserdmi@gmail.com

Telegram: <https://t.me/iserdmi>

Facebook: <https://www.facebook.com/iserdmi>

Instagram: <https://www.instagram.com/iserdmi>

LinkedIn: <https://linkedin.com/in/iserdmi>