**Human factor in software development**

Have you ever thought about the role of human factor in the life cycle of the project? Unfortunately, software development companies don’t give as much attention to the point as it should be.

**IT and Psychology**

Quite often we come across a rather one-sided view on development in IT. The customers’ point of view is limited by three main desires: high quality, fast result, get-more-pay-less. But these three factors sometimes contradict each other. Sometimes, tight deadlines have immediate positive influence on high quality of the product as it stimulates mental processes and job productivity. But in most cases this strategy leads to conflicts between a customer and a performer. There are situations, where badly analyzed architecture or fast software development leads to technical debt (code debt). Technical debt appears when a project needs to be updated, in particular, by other performers, a lot of bugs appear and it becomes problematic to add new features. The most widely-spread reason of technical debt is tight deadlines and lack of customer needs understanding.

The above mentioned factors don’t show us the root of the problem. If we probe deeper, we will see a real problem. We gained insight in customers’ desires. And what desires does a performer have? In his turn, he wants to earn a bunch of money from the project and continue collaboration with good customers. In one word: to be well-paid and always have a job. In this case there appears the main stumbling block in human to human relationship, well known as the conflict of interests. New rules of life demand: “get the biggest part of pie and don’t think about the rest”. To escape this scenario people should be honest with each other or make concessions to build win-to-win relationships. Hopefully, there are some approaches to satisfy both performer’s and customer’s needs.

**Building win-to-win relationship**

First of all, human 2 human relationships must be based on mutual respect. Customer must realize, that project is developed by humans, not robots. Some mistakes should be forgiven, as human can’t exist without making mistakes. Also, there are mistakes that are made because of performer’s skills level. If you employ a junior developer, don’t demand from him senior developer skills.

As honesty is a very unsteady thing in our time, we should invest in building up good relations with a customer. A performer must realize that a customer is a dummy. Probably, this is the golden rule in IT sphere: Customers can’t imagine software development life cycle of his own product and all complexity of its features. So, to escape conflict, in every deal a performer must shower his questions about the task on the customer. Every item from the terms of reference must be discussed with a customer. After building a clear realizing of the task, a performer or a team of performers may start working.

The well-known fact is that developers are very lazy, whether they are freelancers or work in software development company. All of them want to complete the task without much effort. This approach has an effect on a low quality work results. Surely, this is not as terrible as it can appear, but customers all the same remain dissatisfied. To prevent this sad situation, the relation to the project should be more responsible than it is in modern world. All steps of work must be clear to all developers, who take part in the work. To start work, a performer should choose the frame of his own work. Developers all over the world most often use agile software development. In short, this frame allows you to minimize risks by dividing the life cycle of the project into little parts and conform to new conditions. ASD-approaches help to speed up development process, allowing to estimate work and project’s status. Also, agile software development makes communications with a customer easier: performers can report about executed work very often. Doing so, they make a good opinion about themselves.

The structured code and presence of comments have a powerful influence on quality of a project. Maybe, the performed project will be updated by other developers. They wait that the code will be clear and they can fix bugs and add new features fast. But writing a readable code is really a hard task. To evolve skills of creation a readable code experienced developers recommend to read Bob Martins book “Clean Code: A Handbook of Agile Software Craftsmanship”. Read the book, and both customers and developers that work with you or with your code will appreciate cooperation with you.

One of the main things in relations between a customer and a performer lies in permanent connection. Performer must communicate with a customer to discuss all appeared questions. In this situation performer shouldn’t be shy and ask all questions in which he is interested. Also, a performer must announce about all the tasks that were completed. For this open repositories or special programs that track time and allow a customer to see what task the developer is doing right now are made.

**Making conclusions**

Wise men say that in every situation you must stay a human. There are many sad stories, when a customer didn’t pay money to the performer, or vice versa, performer took the money but didn’t finish the project. Responsibility and respect to your business partner will offer you a high and permanent profit.

If you are a performer, imagine that you are a wizard or a gin, who helps human dreams to come true. Or if you are a customer, imagine that you are a powerful lord who make anyone’s dreams come true. This imaginary game will help you to create strong relations with your partner.