

What Is The One Word That Causes Projects To Fail



Video Transcript:

Paul: Hi. It's Paul Clifford from [Disruptware](https://www.disruptware.com/) and I want to tell you about **one word that stands between you and a software disaster**. That word is communication. Now, essentially, let's look at just four examples throughout the development process where software projects go wrong and it's all down to this one word.

Firstly, in the **design**, what a lot of people do is to submit a design to a freelancer or [oDesk](https://oDesk.com/) or something like that and they select a developer and off the project goes. They're surprised when they get something back, which is not really what they wanted. So the first thing that went wrong there is that the developer didn't communicate back to you or basically the questions. You must have questions coming back, because no developer is going to understand

your design and just go off and do it. **The more questions you get back, then the more that validates that the developer really understands your design.**

Secondly, when you're actually **selecting someone for your implementation**, make sure you interview them. It's all too easy just to do everything via your email, but doing interview at least via [Skype](#) chat so you can actually **establish your rapport**.

Now, [establish a set routines](#) of when you will communicate and ensure that they are understanding what you're saying, because it's all too easy especially in the global economy when you have people in India, China, Russia, wherever for **miscommunications to happen and that's where a lot of this goes wrong**. So make sure you interview the developer before you select them.

When you're in your project and you have a conversation either a Skype chat, or anything like that, often these chats can go on a long time and what is essential is that you **record that either through voice or at least make a bullet-pointed list of what's been agreed**. And ideally, you should be using some sort of project tracking or task management system like [Trello](#) at the very basic level and put all the bullet-points on what's been agreed in there. So that you both have a really good understanding of what was discussed and what the next steps were.

Number four, **ensure that you are hearing from your developer on a daily or weekly basis**. Obviously, it depends on the length of the project. If it's many, many months, then weekly basis might be fine, but if it's shorter project or shorter module, then **make sure you're talking to them on almost daily** basis or a minimum every other day, because as soon as that breaks down, then things start going wrong, and then three weeks later you try and communicate with your developer and you find out he hasn't started it, or he's been sitting there, or he's on some other project, or he's disappeared altogether. **You need regular consistent communication with them** and that way then you can be assured that he is working hard

moving your project forward and you're getting any questions, you're able to deal with any problems really straight away.

The fifth thing, is just to **understand why developers sometimes don't communicate back to you**, and often it's because, in a way their embarrassed, they have a problem and they don't have the solution and the thing is you know, you could probably work around any problem that comes up. And between you - **you can find the solution**, but when the developer has it on their own, then they can really struggle and instead of coming back to you, and this is often the cultural thing, instead of coming back to you and saying, "Hey, I got a problem. I can't solve this." Often, they'll just go quiet and you can lose weeks of time that way.

That's why it's important to **be on top of it**, hear from them every day, check on progress and see your code and your project developing really, really nicely.

I hope you found that useful. That's Paul Clifford from [Disruptware](http://www.disruptware.com).

Recommended Resources:

1. oDesk - <https://www.odesk.com/>
2. Skype - <http://www.skype.com/>
3. Trello - <https://trello.com/>

Are you ready to become a software entrepreneur?

Get everything you need at www.disruptware.com/.