

MODULE 02 - 021: Requirement Elicitation Phase - Performing an effective system for building applications.

This guide walks through a practical system for building a set of requirements for a new application feature.

Imagine for a minute that you're a freelance developer who was handed a new feature to build by a client.

Then picture yourself building an elegant feature, all of the code working perfectly.

You follow best practices and ensure that all of the potential edge case scenarios are covered.

Now imagine that you're demoing the bright and shiny new feature to the client.

But instead of telling you that you're the best developer in the world that they're going to name their 1st child after you.

They look at the application confused, because what you built didn't match what they had in their mind at all.

This is a scenario that is played out all too often in the freelance development world.

And in many cases it's due to a poor requirement elicitation process.

The story I just mentioned is not a made up parable, it happened to me recently.

And when I say recently I mean yesterday (at the time I wrote this).

My Name is Jordan and I Wrote a Poor Requirements Doc... "Hi Jordan..."

So what did I do wrong?

The issue was caused by me rushing through the requirement elicitation phase.

I have worked for this specific client for over 5 years and I got lazy confirming the exact set of requirements needed for the feature.

Freelance Requirement Elicitation

Let's walk through what happened so you can avoid the same embarrassment and wasted time.

How it Started

A few weeks ago the client contacted me and said that an application I built for them needed a new feature.

The application is an invoicing system that their drivers utilize to generate invoices for clients.

In an email, the client attached this spreadsheet.

He said that the application had to generate this invoice to give to the customer.

The Build

After receiving the email I spent a few days modeling the new feature. I put a list of all of the messages that would be passed between modules.

I built UML diagrams to ensure the data was modeled properly.

After careful planning, I spent two weeks building the new feature and it came out perfectly.

To be 100% honest I was very proud of the work that I did.

The feature was flawless and completely bug-free.

It also fit in perfectly with the rest of the application.

I deployed the code to the staging server and I waited for the client to start showering me with praise...**but the praise never happened.**

The Problem

I emailed the client and gave a video demo of the feature.

A few hours later I received an email from the client that said:

“I’m confused, what exactly is all of this? In my email, I just meant that we need the invoices to be formatted like this spreadsheet.”

So it turned out that the client didn’t want a new module built into the application at all. I

nstead, they simply wanted an additional format option for their invoices.

Who Was at Fault?

So who exactly was at fault?

It may seem natural to put the blame on the client since they didn’t make their request clear at all.

And I was tempted to get upset and blame them (especially for the first 10-20 seconds of my fury).

But then I realized that this issue was completely within my control.

As freelancers,** it’s our job to manage each stage of a project**.

If we rush through the requirement elicitation phase, anything that happens after that stage will fall on us.

A Better Way

So how could this have been avoided?

Let’s walk through the process I should have followed and which would have lead to a better outcome.

For myself and the client.

Step 1

Right after getting the email I should have responded back to the client with clarification questions.

Examples might have been:

- ***Do you want this to be on a new page of the application?***

This is better than saying something like: Do you want this to be a new module? **Because a non-technical client isn’t going to know** what a module is.

But they will understand what a new page on the site is.

- ***How will this interact with other parts of ...the website?***

This question would have instantly given me the feedback to know that this spreadsheet was simply meant to be a different invoice formatting option.

- ***Can you describe the flow of ... how this will be generated?***

This is one of my favorite questions to ask because **it forces the client to be explicit with how a new feature should work.**

Many times I’ll ask a client to create a PowerPoint slide deck showing the flow they want from a feature.

Step 2

After asking clarification questions I should have followed up with a prototype.

I could use a tool like InVision or even a simple PowerPoint deck where each slide held a different page of the proposed new feature.

Examples would be:

- Starting with slide 1 – *Here is where you can click on a button to get to the new page.*
- On slide 2 – *I’d show the form page where the user would enter the information.*
- Lastly on slide 3 –* *I would show the invoice that was generated by the new feature*.*

A Better Ending

If I would have followed these two steps it would have taken me anywhere from a few minutes to hours to establish what feature was actually needed.

As you can imagine this is a much better option compared with wasting weeks of development time.

Summary

I hope that this has been a helpful guide to freelance requirement elicitation.

And that you'll be able to learn from my mistake and apply it to your own business.