

Working With "Big Idea" People

Posted At : April 3, 2007 11:08 AM | Posted By : Mark Kruger

Related Categories: Project Management

CF Webtools still does quite bit of work for small to medium size businesses. Unlike large companies where the point of contact is usually a CTO or a project manager, you often talk to the owner or main stakeholder in a small company. These owners are often visionary entrepreneurs with a knack for marketing. That seems to be what it takes to build a company - especially one that is based on a product for sale. Now let me say at the outset that I greatly admire these risk taking geniuses. Such men and women often go through 3 or 4 business failures before they succeed. I call them "Big Idea" people and their confidence and self-assurance are remarkable. They can see past the details and envision the final chapter in the long road to business success. Working with such folks can be frustrating however.

They often have an aversion to dealing with the minute details that are necessary to get an IT project done. They are so full of ideas that they tack on requirements without even knowing they are doing so. I often have meetings with such brightly burning souls. As I draw them to the specification we are *currently* working on, their eyes glaze over. But if I mention what our plans are for "Phase II" and they perk right up. They are far out into the future making millions - while I'm trying to figure out how to build a viable "no-frills" application that is within the budget for "phase I".

"Big Idea" people consistently lack interest in any form of documentation. We create carefully worded specifications that include all the features we think they have communicated to us. In many cases the stakeholder will skip to the bottom line and check out the billing or timeline estimates and gloss over the *actual* requirements. We often get signed and approved specifications for many thousands of dollars where I am convinced the user has not read it at all. We make the point over and over again - "If it is not in the specification then it will not be in the application". Yet we still get surprised looks when it comes time to test the application. "Where is feature A" or "Where is Item B?" At which point we take them back to the specification and ask, "Where is feature A?"

Why is it so hard for such folks to grapple with written requirements? I believe it is because they are used to *telling their story* and *communicating their vision*. Their job has often been in raising money and getting folks to "buy in" to their big idea. This communication is an infectious verbal siren-song that they have fine tuned with years of practice. Unfortunately, communicating the *big picture* doesn't do much to help the requirements building process. To translate the big picture into requirements requires painstaking, detailed and (some might say) tedious thought. You have to think about individual events within the application. You have to make lists of behaviors and use cases. You have to *stop* thinking about all the users you will have and all the money you will make and think about a single, individual user and his or her individual experience.

Some Tips on Working With a "Big Idea" Person

Here are some tips I have for getting through to the visionary:

- **Do not accept requirements, change requests, or enhancements verbally - either over the phone or in a meeting.** Even if the customer is a *phone person*

ask them to write it down in an email or a document and then send it to you. "Big Idea" people are used to the power of their voice. They know how to "sell". But getting you to buy into their application has little to do with actually building it. If they call you with an idea make sure you insist that they send it to you in written form. The phone is perhaps the least attractive way to communicate requirements. It simply does not work. A requirement or a change request is not like ordering a magazine subscription. It's more like balancing your checkbook. It requires precise details. You can't balance your check book over the phone (not easily anyway), and you can't get accurate requirements over the phone either.

- **Insist on a Feedback Process.** When I write specifications I usually include numerous questions highlighted in bold, garish yellow. Getting answers to these questions are a prerequisite to finalizing the document. More than anything I am looking to engage the customer in a back and forth dialogue till we both know what he or she wants.
- **Don't Pitch and Don't Sell.** Instead, warn and qualify. I often tell customers, "I'm sure I got some of this wrong". It gives them a reason to read it (ha). I tell them that I only hear about 70 to 80 percent of what they intend. I write specs with caveats and addendums. I describe possible hidden costs or costs that may be lurking downstream after launch. In short, I see my job as a consultant trying to help them bring their idea to life, not as a salesman trying to extract fees from them. This makes the entrepreneur feel that they have found a collaborator and a partner. Remember, they know what a pitch sounds like.
- **Include revision costs in your specification.** We include a section called "debug and revision" that is at least 15% of the cost of any application. This amount is to account for the change requests that we *know* will come as we progress.