enLumen
Leadership Services

That's Not What I Wanted…

Getting the Results You're Looking For

I spent the early part of my career as a software developer. The nice thing about software is that it is highly predictable. Barring hardware problems, software does exactly what the developer tells it to do. Even bugs aren't the software's fault – it's just doing exactly what some developer (either of the application, the compiler, or underlying operating system) told it to do. Not necessarily what the developer wanted, but what it was told to do. The developer's job is to provide thorough and accurate instructions so the software behaves correctly.

If only human interactions were that straight-forward. Software doesn't require motivation, just accurate instructions. Perfect instructions to humans can still go south if the human lacks the motivation to do the job well.

```
    'replace_interests' => false,
    'send_welcome'      => false,
  ));

(in_array("error", $result)) {
$arrResult = array ('response'=>'error','message'
} {
$arrResult = array ('response'=>'success');

json_encode($arrResult);
```

To complicate it further, two humans can hear the exact same instructions and interpret them differently. So a leader seeking to give "perfect instructions" to humans needs to take into account the prior training and experiences of each recipient to anticipate how they will interpret the instructions.

How does a leader give instructions in a way that delivers the desired results? Here are a few tips…

1. **Don't treat people like software.** Software doesn't ask for clarification, it just executes. Instructing a human effectively should be a two-way communication. Create a culture where your followers know you expect them to ask questions until they understand your expectations clearly. Be an effective listener as well as an effective talker to assure you have a shared understanding.

2. **Know your followers.** Motivation matters, and what motivates one person might not motivate another. Know them well enough to connect the importance of the task to things they care about (beyond a paycheck). See each assignment in the light of how it will help develop your follower's abilities. Understand their development level and calibrate how you interact with them accordingly. I guess there is a similarity between leaders and software developers: It's the software developer's job to understand how the software works and the leader's job to understand how his/her followers work.

3. **If you don't get the desired results, assume it was your fault.** As a leader, you are responsible for communicating expectations clearly. Clarity in your own mind doesn't count. Did *they* understand your expectations clearly? If you wanted something in sky blue and they delivered royal blue, whose fault was it? Did you specify the shade of blue and validate that they understood your expectation? Just because you said it doesn't mean they got it. *Communication clarity is measured by what is received, not by what is transmitted.* Only when you know you've executed your responsibility well do you have solid ground to hold them accountable for their performance.

Clear communication is hard work, and as a leader it is your responsibility. Own it. Commit to doing it well. Admit it when you don't.

***enLumen Leadership Services**: illuminating your path to better leadership.*