What You Can Learn from the *Healthcare.gov* Debacle

October 25, 2013

If you sat through the televised hearings conducted by the House Energy and Commerce Committee, in which Congressional Representatives ask questions they do not understand, and the senior executives of four contracting companies attempt to answer detailed questions without direct knowledge of those details, you could learn something about information technology projects. It wasn't easy, but it was instructive.

The overall picture reveals a project that failed, not because of the complexity or size of the solution, but because the project leaders failed to execute basic project management techniques – techniques that most information technology professionals would consider "Project Management 101". Consider the following project characteristics revealed by the Committee:

1. <u>No single contractor was assigned overall responsibility for the project</u>. According to the testimony, fifty-five (55) contractors were engaged on the project, but no one sat at the head of the table.   An old project management axiom says, "When everyone is responsible, no one is *really* responsible".   In this environment, it would have been impossible to escalate problems and resolve them in a timely manner. As one can see in the testimony, the participants were in a perpetual finger-pointing loop.

2. <u>The system specifications, or blueprint defining what the system would do and how it would work, were not delivered to the construction team until this summer - just a few short months before launch.</u> This did not allow sufficient time to build the system and test it to the degree it required. Not only were the specifications delivered too late in the cycle, they continued to change right up to the launch day. Therefore, the single biggest IT system ever launched by the US government needed to be built and tested in less than six months! Information technology journals and global consultants have measured the impact of incomplete or ill-defined specifications and found that "Bad Requirements" is the most frequently cited factor in system development failures.   *Healthcare.gov* confirms that research.

3. <u>The testing phase was woefully inadequate.</u> According to the witnesses, no end-to-end test was actually conducted. Several witnesses stated that they expected the integration test to begin three weeks prior to launch. However, three weeks of testing is far too short for a system this large. If that was really the plan, it should have been a bright "yellow flag" for the success of the project.

4. <u>The responsibility for most of the primary testing was placed on the user community within CMS.</u> While user testing should always be part of the acceptance process, having the users perform the primary testing is a recipe for disaster. Primary testing must be the responsibility of specialists close to the construction team, before it is passed to the users. Apparently, this approach was not used.

5. <u>When the project began, no one conducted a project "kickoff meeting" in which all project stakeholders could convene and discuss project boundaries and lines of authority.</u> Contracts were awarded and the contracting companies were left to their own

devices. This was very apparent in the witness statements as they described a very fragmented work environment.

During the testimony that the author observed, the Representatives spent the majority of their time probing the lack of testing. However, that phase of the project was far too late in the cycle to remedy the _management_ issues that doomed the project. More time should have been spent exploring the initiation of the project because that is where the problems began and the outcome determined. What we have now is a system that is being "designed during the implementation" never a good idea.

Most enterprise-level software development projects tend to be highly visible, expensive, and risky. This is true in both the public and private sectors. When these projects fail, they fail spectacularly. If your business or organization is about to launch a major software development initiative, you can reduce the expense and mitigate the risks by avoiding the mistakes outlined above.

About the author. _Al Moore is one of the four founders of Marathon Consulting, LLC, located in Virginia Beach, Virginia. The firm offers a range of Information Technology services including Custom Application Development, Mobile App Development, Data Integration and Database Programming, Internet Marketing and E-Commerce, and IT Advisory Services. Al can be reached at_ amoore@marathonus.com _._