Successful Project Management in the Digital World

Managing people, projects, and processes has always been a challenge. In the digital age, it's an even more complex challenge.

In order to grasp the issue, we have to examine how we were taught to manage. We've been taught to make a plan and stick to it. We accomplished this by breaking the plan into segments, establishing budgets and dates for these segments, and then measuring the results. If we were extremely formal, we made a nodal view that included relationships between items, such as finish to start (FS), start to start (SS), finish to finish (FF).

Then, perhaps, we made Gantt charts depicting a critical path. Depending on the user's sophistication, some inputs could be pulled from other systems, outputs could be tied to other systems. The number of nodes in these schedules could be hundreds or tens of thousands. Does this system work effectively today? The answer is no. In truth, it only worked well in situations where the process was predefined and largely unchanging for the duration. When using this method, variation creates havoc.

Furthermore, the most ineffective part of the above methodology is that individual participants typically work on only one to three items at a time; the rest of the information is just noise to them. So how do we deal with the noise? Usually, we add one or more levels of management to filter it out. Not very efficient or cost effective.

How can we approach these situations more efficiently, more flexibly? The answer: make managerial procedure more integrated, more helpful to the participant, not an obstacle. What does this new system look like and how do we implement it?

First, set the plan's highest level using milestones; don't worry about the details of achieving them. Next, the groups involved in bringing milestones to fruition should define their own method and granular detail in getting things accomplished. Then in the execution stage, establish sprints that can last from a day to two weeks. The participants define the sprint by the task to be accomplished and the method by which it happens. The team then huddles every morning to discuss the plan for the day as well as any issues that have arisen and how to resolve them, similar to a scrum. In defining this, two issues arise.

One, what if this is a very detailed program, like building a Boeing 787, can we depend only on high-level plans? Obviously not. The gist of the system is to let smaller groups detail their own methods of accomplishing tasks within a framework *and* having a system that facilitates the view and management of their final process. This method empowers teams to work in their own familiar framework as it facilitates innovation by reducing both management overhead and extraneous information that doesn't pertain to that group's task.

Two, how do we get accurate status from our new system? The illusion of accurate status using the old methods was just that: an illusion. When we present users with complex ways of doing things that are not in alignment with their own methods, three things occur: A) gaming of the system, B) working outside the system, and C) errors due to unfamiliarity with the new methodology. It has been shown that when you change methods, the number of mistakes a participant makes increase dramatically. Traditionally, we would think that more training is appropriate, but this doesn't reduce mistakes. A person who does a task has his or her own established method for finding mistakes in their work, call them rules of thumb, i.e. "It doesn't look right to me." With an unfamiliar system, it may never look right to them, so the workers learn to coast on blind trust. If the process moves quickly, this results in very costly mistakes, delays, cost overruns, and unhappy clients.

Bottom line: We need only to see what is pertinent to us, within a flexible framework.

How can we achieve this? The only way is to create a collaborative environment that's both transactional and contextual. Once we create this environment, participants can work freely in accomplishing tasks without the burden of a system bogged down by additional processes. The transactional framework lets participants see who is doing

what, find what has been done, as well as track errors and triumphs. The system does the heavy lifting, not the user. This philosophy forms the basis of the design for Qikspace: understanding that process doesn't merely include nodes in a Gantt chart, it encompasses the meetings, tasks, and workspaces integral to accomplishing your project, while contextualizing the method via communications, notes, files, and other content. The simultaneous ability to know what is going on, what has gone before, and who is involved defines the success of the process.

About Us

Qikspace (www.qikspace.com) specializes in social collaboration software with a personal relationship management (PRM) component. Qikspace was started as a research project in 2011 the emphasis was the analysis of contextual relevance in relation to human interactions. The result of this research became the platform that is being developed and enhanced today. Our unique philosophical and technical approach has allowed us to create a solution to the complex world of online human interactions and the consequent collaborations.

About the Author

David Smith is the founder of Qikspace and has an MBA from Columbia University, is a Professional Engineer, and studied Electrical Engineering at the University of Washington.

Qikspace ™

Contact Information:

w. www.qikspace.com

e. david@qikspace.com

c. +1 732-673-2370

P.O. Box 24043

Seattle, WA 98124

