RAPID. FORWARD. FAST. VISION. MANAGEMENT.

written by Manoj Khanna | February 1, 2003

There is a great deal of progress in the past two decades in Software Development Management, in terms of improving software quality, development productivity and overall development process. The progress over the last two decades specified what to do with the programs like Sourcecode Configuration Management (ClearCase, CVS, etc.) and Defect Tracking System. On the other hand, the techniques learned are impossible to put into a program.

“VISIBILITY” is the most important element of project, and so is its “ABILITY TO CHANGE DIRECTIONS”. A highly visible project tells it’s leader where it is at all times.

In a “VISIBLE” scenario, the project instruments everything with tests. Not regressing the bench version, and making every feature available in bench version a deliverable. Thus making the status of the project always visible.

To be visible, a project must write Programmer Tests (sometimes called Unit Tests). Done right, they reduce the odds of defects to so low that the few bugs you actually get will simply go into your requirements tracking system.

For direction changes the project must have a configuration tool (tools such as CVS, ClearCase, etc.) and the prescence of copius tests. This enables the leader to request features in the most important order, and any feature which is not requested cause no waste code and eventually no waste of work.

Thus this signifies all the tests that ensure visibility also permit steering. Tests will tell you what’s done and what isn’t.

In other cases, when there is no code produced then a record of document reviews is a good indicator of project status. Also its important to track what's going on with change requests and defect reports coming into the project. A

good risk register is also important.

A good quality system can go a long way to improve the development process, though it is up to you to perform the implementation. There are several documents which define a quality system. Two of those which are well-known are:

– IEEE Standard 1298

– ISO Standards 9001:1994 and 9000-3:1997 (or the later versions).

There isn't any "canned" software development system that works across all types of software development projects. In fact, some software gurus adhere to a "contingent" approach to software development. Why? Because there are a broad variety of types of software development projects, there is, thus, a need to use a broad variety of software development approaches, depending on the type of project involved.

© Manoj Khanna 2003 – 2012.