



Stone Software Development

Sometimes I feel like a procedureless child, a long way from home. Then I remember this marvelous story that my great-grandfather never told me.

Once upon a time in the land of Need-It-Right-Away, there was a great dearth of well-written software. Software customers, greatly desirous of obtaining applications that did exactly what they wanted – yesterday – and at the very least possible cost, had rendered all of their in-house developers into beaten, sniveling hackers, who cowered within fabric-covered holes, ate fat mixed with refined flour and sugars, updated their resumes, and dreamed of better times. Then one day a lone developer from a neighboring province rode into town, strode through the swinging doors of the largest conference room, hung up his spurs, and began to speak with the local project managers as if he planned to stay for a few accounting cycles.

“There's not a charge number to be had in any of our kingdoms,” he was told. “Our needs are too urgent, we can't afford to impact any of our schedules by bringing you on board. Besides, unless you can provide a product before COB that is exactly what we are imagining at this very moment for less than minimum wage, why, you're just wasting our time.”

“Ah, no problemo” the lone developer replied. “In fact, I was thinking of creating an application to share with all of you based on the stone development method.” He pulled an old tempest-hardened laptop from his saddlebag and booted it up. Then he removed a small, smooth, bluish-tinted

stone from his pocket and carefully placed it next to his machine.

By now, a flurry of e-mails and flash notes with rumors of a new development method had drawn many customers and in-house developers to the conference room. They scurried for chairs, popped open containers of carbonated caffeine, and brushed the crumbs of deep-fried artificially flavored foods from their sweatshirts. The lone developer closed his eyes, stretched ergonomically, assumed an enigmatic expression, and then wiggled his fingers over the little blue stone by his keyboard while all of the customers in the room beamed their concept of an ideal system at him via mental telepathy.

“Ahhh,” he said after several deep cleansing breaths, “I do so much enjoy stone development. Of course,” opening one eye to peek at the customers, “stone development with requirements – that's hard to beat.”

The local developers gasped, but after a moment one of the customers admitted that he did have, somewhere, a list of specific and fairly well documented requirements that identified the greater portion of what he imagined the ideal application should provide. “Outta sight!” the lone developer exclaimed as he leafed through the pages and placed them next to his stone. “You know, I once worked on a stone development project with requirements and a few plans, and it was simply incredible. After all,” he said with a wink at the project managers, “just asking for something by COB is a little vague, don't you think?”

One of the managers looked at his customer, and between the two of them came up with basic information for a schedule and simple quality assurance and configuration management plans. Encouraged by this, several in-house developers began to interact directly with the customer, while the manager used the simple metrics identified to track progress, risks, and costs. Meanwhile the lone developer's fingers flew over the keys of his laptop, swiftly integrating the flood of information that began to pour forth. Design elements, reviews, testing, and acceptance criteria, all clearly traced in a matrix soon resulted in a secure, relational, Section 508-compliant, real time, Web-enabled, embedded application that was a marvel to everyone who used it. It wasn't quite free or completed by COB the first day, but no one seemed to notice.

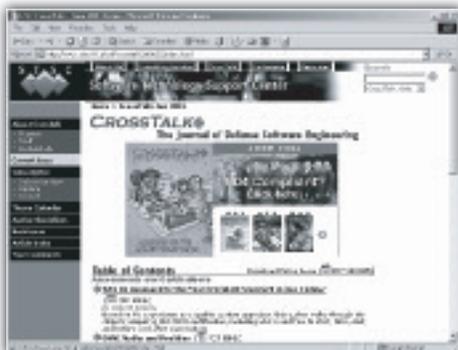
The customers and managers of the land of Need-It-Right-Away offered the lone developer a prestigious title, a black belt, and a great deal of money for his little blue stone, but he graciously declined and eventually rode off into the sunset. However, history records that from that time forth, there was no longer a lack of well-written software within any of the kingdoms of the land.

And all of the in-house developers, who had taken very careful notes, began to eat better and spend more time with their families.

The End.

— Robert K. Smith
rkensmith@earthlink.net

CROSSTALK ARCHIVES



Visit CROSSTALK online and access 11 years of software-related articles in a fully searchable database that makes finding relevant and timely information painless. Make CROSSTALK your first stop for all your software research needs.

www.stsc.hill.af.mil/crosstalk