



## **Case Study – Agile Development Breakdown**

Robert Dolezal, Partner, Consultiq

Spring 2008

© 2008 Consultiq™, All Rights Reserved

Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

# Agile Software Development Breakdown Scenario with Situation Analysis

## 1. **Client:** Database Publishing Company.

The client is a database publisher of insurance information products sold as newsletters (news) and commercial insurance expiration data (data) via traditional printed and online products.

## 2. **Situation and Approach:**

In January, 2007, client tasked its internal IT development team with creating a unified proprietary subscriber database that could communicate through an API and web service with both intra-company functions such as customer service and extra-company functions such as an off-the-shelf Software-as-a-Service CRM sales-tracking and customer maintenance system. The database operated in a Microsoft SQL 2005 instance with ASP.net framework via Visual Studio and C# programming. The IT team was allowed to use Agile Development methodologies based on an iterative series of steps leading to a tasking statement. Management and the IT team agreed that the development could be expected to take between two and three months. The IT team chose not to use structured product management tools like Product Requirements Documents (PRD) and Market Requirements Documents (MRD).

The IT team proposed architecture and database definition inputs based on user object definitions provided by management and the user team. In each case, the IT team diligently reviewed the inputs and developed trial screens and flowcharts of the database and their proposed API. In each case, management and users proposed alterations. This cycle continued until the three groups reached a consensus on an acceptable approach. An elapsed two months were used in these iterations.

During build, the IT team encountered a series of unforeseen obstacles to their planned approach, requiring changes to the design, including changes made by Microsoft to Visual Studio and SQL. During reporting of these changes and attempts to reach a new consensus, additional iterative cycles consumed time and staffing resources, and delayed the project by an additional two months. By this time, the project was still only an estimated 25% complete, 200% of budget, and two months overdue. The project manager was fired as a result of the project failure.

A month was consumed in searching for a new project manager. Simultaneously, management decided to use Salesforce.com as its CRM. A site license was purchased and a Salesforce consultant was retained to help with the API integration. The consultant did an exhaustive user

study of the internal customers (customer service and sales team) and detailed the requirements of the API. Additional iterations were required, and another six weeks were consumed in the analysis and architecture development, which required discarding most of the work done previously and instituting a new architecture. At the conclusion, management retained significant doubts about the suitability of the proposed system that would have relied on Salesforce.com's external database as a repository for the company's proprietary customer data without an internal backup database.

Nevertheless, because of the high visibility of the project and investment to date, the IT team began to build the API and test integration with the Salesforce.com API. A project review at one month estimated that 50% of the development was complete. Management reviewed the project and found that the database on Salesforce did not contain key data objects required by the company, and once again architecture was adjusted to create a SILO database in SQL that was independent but linked to the Salesforce database. This decision required two weeks of full IT staff to accommodate and alter the prior development plan. The project was judged to be 33% complete at the end of this review cycle.

In all, seven months had elapsed since project initiation. The IT team had neglected other mission-vital development, was demoralized, and felt that they should keep going regardless of whether there was management agreement that their plan would result in a workable system. The project had cost 5 developers and two managers' time, was 400% over budget, and was 800% over schedule.

### **3. Interfering Factors:**

Our consultancy was retained to provide interim operating management for the company until a new CTO could be identified and hired, inheriting the IT project. After initial fact-finding, we decided with management's approval to halt the project and return to traditional MRD/PRD development standards. In two weeks, decisions were made to abandon the Salesforce.com system, simplify the requirements, keep the database entirely in-house, and use the inputs from the Salesforce consultant's user case studies to define a new database architecture and approach. Management concurred with these decisions as a potential path to bring the project to a resolution.

The primary breakdowns in the agile development procedure occurred through lack of maintaining a single point of control for the requirements of the project and through dividing

decision-making authority equally between the development team's project leader and the senior manager driving the development. Other interfering factors were determined to be:

- a. Failure to define the project in execution stages with clear benchmarks wasted time on false starts.
- b. Failure to maintain documentation of the system's role, function, and purpose meant that there was no central clearinghouse suitable for testing development theories before they were implemented.
- c. No time estimates, variances, or other quality control functions were in place to provide measurements or determine whether schedules were reasonable.
- d. Changes to requirements during development rendered prior decisions obsolete.
- e. Failure to assess the weaknesses of Salesforce which could have prevented selection of the system for implementation.
- f. Morale degradation limited the potential for programmer and analyst feedback that could have identified false paths and approaches.
- g. Management blame placing focused on the individuals caught in the process rather than identifying the process itself as the cause of the failures.
- h. Management distraction by other issues lost effective control of the process.

#### **4. Economic and Other Consequences:**

Failure to complete the project on schedule and in budget directly cost more than \$1 million, 11 months for a 6-person IT team, plus an estimated 30% of the time utilization of three senior managers.

Indirect costs were estimated to cost more than \$3 million, including:

- a. Revenue product releases were delayed by six months because no sales-support or CSM, with resultant lost-opportunity costs.
- b. Undue staff turnover occurred and productivity was negatively impacted.
- c. Other IT projects were delayed or eliminated.
- d. Future revenue-generating product development was postponed or eliminated.
- e. Funds spent on license-fees for Salesforce.com were wasted.
- f. Mission-critical production revisions were postponed or delayed due to management attention to the agile development project.

The impact on the organization was so heavy that management seriously entertained the entire discontinuance of operations due to the costs experienced and the lack of confidence in the company's ability to develop other internal software applications necessary to its growth and operations.

#### **5. Resolution:**

Following the two-week assessment, an additional seven-day period was used for a joint project definition meeting facilitated by our consultancy. Participants included the entire IT team and the relevant managers. Our lead consultant assumed the project manager's role. A market-requirements analysis resulted in major changes to the development assumptions, adoption of a web-service and API structure accessing a single SQL database, and a simple set of web-based screens for entry, editing, and recording of all customer subscription data into the system. Future hooks were defined for the CRM that would allow sales team members to access critical data, as were data structures for forthcoming products that previously were overlooked in the systems design. A simple PRD allowed the program team to modularly divide the project according to expertise. The customer service managers were brought in prior to start of development and identified several key items that had been overlooked, which were incorporated. With management concurrence, the project initiated and was completed in four weeks from acceptance of the PRD, including all management-mandated requirements for the first release. Two additional revision cycles were planned over a one-month period to add desirable functionality. Because the span of scope and requirements included future product releases, they were incorporated into the system with the two revision releases, eliminating the need for future IT development. Morale was restored, and management regained faith that IT could develop its software internally. A CTO was hired to standardize the development process with traditional MRD/PRD product lifecycle process structures.