**HEALTHCARE.GOV WEBSITE FAILURE**

**TEACHING NOTE**

*This teaching note was prepared by the authors and is intended to be used as a basis for class discussion. The views presented here are those of the authors based on their professional judgment and do not necessarily reflect the views of the Society for Case Research. The information in this case is totally based on publicly available news sources. Copyright © 2014 by the Society for Case Research and the authors. No part of this work may be reproduced or used in any form or by any means without the written permission of the Society for Case Research.*

**Critical Incident Overview**

This critical incident is a **descriptive** case covering the oft-reported failure of Healthcare.gov website. The case provides a broad overview of the desired functionality of the site, its construction and testing and it ultimate implementation problems. The incident focuses on project management, leadership and procurement aspects of the website’s failure.

Instructors can use this incident primarily in information system related courses, especially those in information systems management, project management and systems development. Public Administration students may also examine this incident to see the impact of technology implementations on public policy. Students in undergraduate and graduate programs can benefit from this critical incident.

**Research Methods**

The authors wrote this critical incident based on publicly available news reports.

**Learning Objectives (LO)**

The learning objectives of this critical incident are for students to:

1. Analyze the factors that lead to complexity in large scale G2C system development and acquisition
2. Evaluate system testing approaches for a large scale web-based system
3. Evaluate leadership’s response to testing results and the lessons leaders can learn about system implementation from this incident.

**Questions**

1. **What factors made Healthcare.gov a complex system to implement? What effect does complexity have on the likelihood of project success? (LO1)**
2. **Evaluate the system testing approach taken in Healthcare.gov. What improvements would you recommend? (LO2)**
3. **What should leadership have done upon receiving the CGI Federal memo (p. 2, paragraph 5) identifying implementation risks based on inadequate testing prior to the October 1 rollout? (LO3)**

**Answers to Questions**

1. **What factors made Healthcare.gov a complex system to implement? What effect does complexity have on the likelihood of project success (LO1)**

Student answers to these questions will vary. As for factors that made Healthcare.gov complex, one source (Gallaher, 2013) identified the “seven deadly sins of Healthcare.gov”. The first four speak to complexity:

1. Hyper-Complexity. The HealthCare.gov project was an amalgam of three major contracts, each with its own contractor and set of deliverables: a new e-commerce site, a new information middleware infrastructure, and a hosted data center integration project.
2. Dependency issues. In addition, the whole thing was dependent on data provided by Experian—a data source that neither the government nor the other contractors could do any sort of data quality work on. Without a way to handle exceptions in Experian's data—such as a mismatch between street addresses due to a misspelling in Experian's system or just old data—the site experienced many early headaches.
3. All new construction. Many government IT projects, particularly ones that are created as the result of specific legislation, require the construction of an entirely new infrastructure. HealthCare.gov had the complexity multiplier of being based in software and systems—the "data hub" middleware that tied the site to the systems of insurance providers in particular—that had never been used live before.
4. Rolling requirements. The specifications for the project were delayed repeatedly then changed frequently up to within a month of the target release date. The tweaking resulted in design changes.

Faculty may also wish to point out the “right to left” scheduling challenge faced in this project. The due date of the project (right hand end) was immoveable as October 1, 2013. The starting date (left hand end) kept being delayed by changing requirements.

Another dimension that faculty may wish to speak to is “scope creep”. In Healcare.gov, “rolling requirements” (described in d. above) is the source of “creep”.

The impact of complexity on system success is well documented and long standing. As early as 1996, Baccarini wrote in the project management literature of the impact of complexity on project outcomes. In 1997, Ewusi identified complexity in his work on critical issues in abandoned systems. Subsequent literature supports these findings.

Part of the complexity is due to the number of vendors. A good review of the vendors involved in the case can be seen at:

<http://www.npr.org/blogs/alltechconsidered/2013/10/25/240532575/a-diagram-of-healthcare-gov-based-on-the-people-who-built-it>

1. **Evaluate the system testing approach taken in Healthcare.gov. What improvements would you recommend? (LO2)**

The failed initial implementation of Healthcare.gov is an obvious indictment of system testing. Three of Gallaher’s “seven deadly sins of Healthcare.gov” (2013) speak to testing related issues:

1. Anti-testing. Since the requirements kept changing up until the last minute, there was no way to do full site testing until mere weeks before the release date. It's not clear if any real-world data was used in testing the customer validation piece of the site, since that would have required hitting actual Experian credit data. There was no limited release of the system for "beta testing" among a select audience, aside from the demo done by President Obama. All this despite the fact that nearly every component of the site was brand new and unproven against any real-world load.
2. Release late and once. Instead of doing a rolling release of features starting with information on what people could expect in terms of subsidies, the government committed to an all-or-nothing release date. There was no way to test the site's performance under full load as a result, and the feds couldn't gradually scale up infrastructure based on experience and testing either.
3. Anti-bugfixing. There was, based on statements from the government, no effective way to manage bug tracking across the multiple components of the site. This meant no way to identify root causes of issues and prioritize fixes at the time of launch. Instead, the contractors implemented all this after the launch. And while the data center provider had been certified as complying with government security requirements, it's not clear that there was ever any realistic capacity planning done because everything was so new.

Students should recommend the antithesis of each of these points. On “anti-testing”, Health and Human Services (HHS) should have frozen requirements and participated in full system testing with a realistic load (perhaps with simulation). Student should also point to the need for capacity planning based on testing results from full load testing. On “release late and once”, HHS should have done a rolling release with essential features on October 1 and then add features as time went on. Additionally, following an agile development process that focused on incrementally released core features would have improved the ‘final’ project compared to a “release once” approach. On “anti-bugfixing”, students should point to the need for a common bug tracking across all vendors.

1. **What should leadership have done upon receiving the CGI Federal memo (p. 2. paragraph 5) identifying implementation risks based on inadequate testing prior to the October 1 rollout? (LO3)**

This question should spark student discussion and does not have a definitive answer. Political pressures undoubtedly led the administration to continue implementation. When the system failed to perform, there were widespread calls, especially from the GOP, that “someone ought to get fired” for the debacle. Others (Kimberly, 2013) argue that firing someone is not essential, “just fix the mess”.

In this incident there is a clear “reverse halo effect” (or “devil effect”) wherein the failure of the Healthcare.gov website influenced people’s evaluation of other aspects of the Affordable Care Act.

Project management and IT professionals undoubtedly would recommend that the project should have been delayed to allow for further testing and, perhaps, phased implementation. Political pressures may not have allowed for this, although politicians may not have been fully aware of the potential fallout from failure.

**General Discussion**

Students should find this incident to be timely and one that they can generally relate. Faculty can use this case to enforce the point that system project like Healthcare.gov are so large and complex that they are extremely difficult to manage. The number of contractors and databases adds to this complexity. They can further point to the need to manage and mitigate risk on large projects.

**Epilogue**

As of the writing of this note, over 8 million citizens have signed up with Healthcare.gov, exceeding the initial goal of 7 million. The web site is functioning effectively. HHS has delayed some aspects of the law including the employer mandate by one year. The political damage to the Obama administration and the Democratic party’s mid-term election results, however, have been less favorable. As a result of the implementation problems associated with the rollout of Healthcare.gov, Secretary Sebilius resigned as Secretary of Health and Human Services at the beginning of April, 2014 (Shear, 2014).

**References**

Obamacare: After Obama’s apology, talk of solutions and -- still -- the website. (n.d.). *CNN*. Retrieved November 8, 2013, from <http://www.cnn.com/2013/11/08/politics/obama-obamacare-apology/index.html>

Baccarini, D. (1996). The concept of project complexity—a review. *International Journal of Project Management*, *14*(4), 201–204. doi:10.1016/0263-7863(95)00093-3

Ewusi-Mensah, K. (1997). Critical issues in abandoned information systems development projects. *Communications of the. ACM*, *40*(9), 74–80. doi:10.1145/260750.260775

Gallaher, S. (2013, October 29). The seven deadly sins of HealthCare.gov. *Ars Technica*. Retrieved November 8, 2013, from <http://arstechnica.com/information-technology/2013/10/the-seven-deadly-sins-of-healthcare-gov/>

Health agency chief refuses to disclose ObamaCare enrollment numbers. (2013, October 29). *FoxNews.com*. Text.Article. Retrieved November 8, 2013, from <http://www.foxnews.com/politics/2013/10/29/health-agency-chief-apologizes-for-healthcaregov-failures/>

Howard, A. (2013, June 28). Healthcare.gov: Code Developed by the People and for the People, Released Back to the People. *The Atlantic*. Retrieved November 8, 2013, from <http://www.theatlantic.com/technology/archive/2013/06/healthcaregov-code-developed-by-the-people-and-for-the-people-released-back-to-the-people/277295/>

Hu, E. (2013, October 25). A Diagram Of HealthCare.gov, Based On The People Who Built It. *NPR.org*. Retrieved November 8, 2013, from <http://www.npr.org/blogs/alltechconsidered/2013/10/25/240532575/a-diagram-of-healthcare-gov-based-on-the-people-who-built-it>

Kessler, G. (n.d.). How much did HealthCare.gov cost? *Washington Post*. Retrieved November 8, 2013, from <http://www.washingtonpost.com/blogs/fact-checker/wp/2013/10/24/how-much-did-healthcare-gov-cost/>

Kimberly, John. Don’t fire Sebelius; just fix the mess. (n.d.). *CNN*. Retrieved November 8, 2013, from <http://www.cnn.com/2013/10/31/opinion/kimberly-firing-sebelius/index.html>

Lafraniere, R. P., Sharon, & Austen, I. (2013, October 12). From the Start, Signs of Trouble at Health Portal. *The New York Times*. Retrieved from <http://www.nytimes.com/2013/10/13/us/politics/from-the-start-signs-of-trouble-at-health-portal.html>

Neuman, S. (2013, October 30). Obama Vows Healthcare.Gov Problems Will Be Fixed. *NPR.org*. Retrieved November 8, 2013, from <http://www.npr.org/blogs/thetwo-way/2013/10/30/241923064/obama-vows-healthcare-gov-problems-will-be-fixed-asap>

Patient Protection and Affordable Care Act: Status of Federal and State Efforts to Establish Health Insurance Exchanges for Small Businesses. (n.d.). Retrieved November 8, 2013, from <http://www.gao.gov/products/GAO-13-614>

PPACA Implementation Failures: Answers from HHS: Hearings before the House Energy and commerce Committee, 113th Congress (testimony of Kathleen Sebelius, October 30, 2013).

Shear, M, D, (2014, April 10). Sebelius Resigns After Troubles Over Health Site. New York Times. Retrieved from <http://www.nytimes.com/2014/04/11/us/politics/sebelius-resigning-as-health-secretary.html?_r=0>

Why did Healthcare.gov’s source code mysteriously vanish from public view? (n.d.). *The Verge*. Retrieved November 8, 2013, from <http://www.theverge.com/2013/10/14/4838100/why-did-healthcare-govs-source-code-mysteriously-vanish-from-view>

Writer, G. A., & News, N. B. C. (2013, November 1). Open-source advocates to government: Let us help you fix healthcare.gov. *NBC News*. Retrieved November 8, 2013, from http://investigations.nbcnews.com/\_news/2013/11/01/21268571-open-source-advocates-to-government-let-us-help-you-fix-healthcaregov