A grounded theory analysis of modern web applications: knowledge, skills, and abilities for DevOps

Article · October 2013

DOI: 10.1145/2512209.2512229

4 authors, including:

Sam Chung
Southern Illinois University Carbondale
45 PUBLICATIONS **366** CITATIONS
SEE PROFILE

Marc Dupuis
University of Washington Bothell
33 PUBLICATIONS **146** CITATIONS
SEE PROFILE

A Grounded Theory Analysis of Modern Web Applications - Knowledge, Skills, and Abilities for DevOps

Soon K. Bang, Sam Chung, Young Choh, Marc Dupuis
Information Technology & Systems, Institute of Technology, UW Tacoma
Box 358426, 1900 Commerce St, Tacoma, WA 98402
1-253-692-5886
{imageup2, chungsa, ychoh, marcjd}@uw.edu

ABSTRACT
Since 2009, DevOps, the combination of development and operation, has been adopted within organizations in industry, such as Netflix, Flickr, and Fotopedia. Configuration management tools have been used to support DevOps. However, in this paper we investigate which Knowledge, Skills, and Abilities (KSA) have been employed in developing and deploying modern web applications and how these KSAs support DevOps. By applying a qualitative analysis approach, namely grounded theory, to three web application development projects, we discover that the KSAs for both Software Development and IT Operator practitioners support the four perspectives of DevOps: collaboration culture, automation, measurement, and sharing.

Categories and Subject Descriptors
K.6.3 [**Management of Computing and Information Systems**]: Software Management - *Software development*

General Terms
Management, Documentation, Design, Experimentation, Security, Human Factors

Keywords
Grounded Theory, Knowledge, Skills and Abilities, DevOps

1. INTRODUCTION
The arrival of cloud computing allowed us to merge together software development, deployment, and operation in what is known as DevOps. DevOps is a combination of development and operations [1]. Consequently, diverse stakeholders are involved in DevOps, including business analysts, software developers, software testers, and quality assurance personnel for development, and system administrators, database administrators, network administrators, web masters, and security officers for operations. Because of these diverse roles, DevOps has four perspectives: 1) a culture of collaboration between all team members, 2) automation of build, deployment, and testing, 3) measurement of process, value, cost, and technical metrics, and 4) sharing of knowledge and tools [2].

IT practitioners have used configuration management tools such as Chef[1], Puppet[2], Salt[3], Amazon OpsWorks[4], etc., to support DevOps. However, the current DevOps approach is limited to the usage of proprietary configuration tools focused on automation of operations.

Instead, we propose a new approach to support DevOps - Knowledge, Skills, and Abilities (KSA) for DevOps. KSA has been used by the Department of Defense to identify the better candidates from a group of people qualified for a position [3]. For this purpose, we first describe grounded theory [4] in the context of several KSAs that have been used to develop and deploy modern web applications and apply it to the artifacts of modern web application development projects. Based upon this analysis, we then discover how the KSAs can support the four perspectives of DevOps and argue that a new theory - modern web application development with KSAs approach support the four perspectives of DevOps.

2. GROUNDED THEORY
Grounded theory is an inductive qualitative analysis approach which involves generating a theory from given source data [4]. Grounded theory consists of five steps. First, all data are collected from the test cases. Second, discrete codes are assigned to the collected data. Third, after assigning discrete codes to the artifacts of the three web applications, these codes are then grouped into concepts. Next, the relationships between concepts are examined and classified into categories. Finally, based upon these categories, a new theory is proposed.

We applied grounded theory to modern web application development. Modern web applications consist of many different artifacts in addition to source code, such as design documents in Unified Modeling Language (UML), Burn Down Charts (BDCs) and logs, Data Flow Diagram (DFD), Dependency Injection (DI), many cloud configuration files, etc. We first assign the names of the tools as the discrete codes to the data since the abilities to generate the data are proved by showing how the tools are used. For example, if we use a scrum management tool, Microsoft Team Foundation Service, we are able to generate BDCs. Then, we group the discrete codes into specific concepts by identifying which tools are employed to support which skills. For example, both Microsoft Team Foundation Service and Google Spreadsheet template are grouped into Scrum project management skills. Based upon the grouped concepts, we classify the concepts into categories – knowledge that can be earned through exercising the

SIGITE/RIIT'13, Oct 10-12 2013, Orlando, FL, USA
ACM 978-1-4503-2494-6/13/10.
http://dx.doi.org/10.1145/2512209.2512229

[1] Opscode chef, http://www.opscode.com/
[2] Puppet Labs, https://puppetlabs.com/
[3] SaltStack, http://saltstack.com/
[4] AWS OpsWorks, http://aws.amazon.com/opsworks/

skills and a perspective of DevOps that the knowledge can support. Then, we propose a new theory which modern web application development with KSAs approach support four perspectives of DevOps from the relationship between KSAs and DevOps.

3. ANALYSIS

We will examine data from three real life cases: a Point of Delivery (POD) Web Application for an Aerospace Manufacturer, a Membership Management Web Application for a Smart and Secure Computing Research Group (SSCRG), and an Online Ordering Web Application for Asia Ginger Teriyaki Restaurant.

To perform a grounded theory analysis, four steps are performed. During the first step, we collect data from the three case studies. Many diverse data are collected.

At step 2, we assign discrete codes to the collected data of each web application. UML diagrams can be created by using Sparx's Enterprise Architect v10.0 CASE Tool or MS Visual Studio Architecture Edition. MVC directories are automatically generated if MS ASP .NET MVC 4 template is chosen. MS Threat Modeling Toolkit is used to draw a DFD based upon the given use case and misuse case. All used tools are identified and assigned to the data.

At step 3, we group the discrete codes into concepts. For example, MS Team Foundation service is used to manage the web application project with Scrum [5] concept or skill. Enterprise Architect CASE tool is used to document software architecture of each application. To use the CASE tool, a set of guidelines for documentation is required. 5W1H Re-Doc [6] is used.

Table 4: Concepts and Categories

	Two-Phase Categorization	
Concepts	**Knowledge**	**Perspectives**
• Scrum	• Agile development methodology	• Collaboration • Automation • Measurement • Sharing
• 5W1H Re-Doc	• Architecture Documentation	• Collaboration • Sharing • Measurement
• MVC Architectural Pattern	• Architectural pattern • Separation of concerns	• Collaboration • Automation
• Dependency Injection	• Separation of concerns	• Collaboration • Automation
• Unit Testing	• Behavior-Driven Development	• Automation
• Mock Objects	• Behavior-Driven Development	• Automation
• Cloud – SaaS	• Architecture Documentation • Agile development methodology	• Collaboration • Sharing
• Cloud - PaaS	• Cloud computing	• Automation
• MS Threat Modeling and Risk Analysis	• Threat modeling and risk analysis	• Automation • Measurement

At step 4, we construct categories that highlight the relationship between concepts. We use a two-phase categorization by first categorizing the concepts to knowledge that software developers and IT operators have learned. For example, Scrum is one of the agile development methodologies, which is a software development process with agility on requirement changes. The categories for each concept are shown in Table 4.

Then, the knowledge is categorized to the four properties of DevOps at the second categorization phase, which is shown in Table 4. For example, the Agile software development methodology supports the four perspectives of DevOps: 1) *collaboration* between all team members by means of daily scrum meetings, sprint retrospectives, reviews, and planning meetings; 2) *automation* of build, deployment, and testing; 3) *measurement* of process and technical metrics by scrum burn down chart, and 4) *sharing* of knowledge by product and sprint backlogs.

4. DISCUSSION

Finally, we propose a theory from Table 4: ***KSAs support the four perspectives of DevOps***. Through the four steps of grounded theory analysis, we first discover what tools are used in creating the artifacts of the modern web application development projects, which skills are exercised to use the tools, what knowledge are studied to understand the concepts of the skills, and then how the knowledge are related to the four properties of DevOps.

Based upon this analysis, we propose a new theory that modern web application development with the Knowledge, Skills, and Abilities approach supports the four perspectives of DevOps. Future work is needed to discover how we can educate and train software developers and operators to be equipped with these KSAs to support DevOps.

5. REFERENCES

[1] Michael Hüttermann, September 2012, *DevOps for Developers*, Apress.

[2] Jez Humble & Joanne Molesky, August 2011, *Why Enterprise Adopt Devops to Enable Continuous Delivery* Cutter IT Journal, Vol. 24, No. 8.

[3] Department of Homeland Security, *Common Terms on Job Opportunity Announcements*, DOI= http://www.dhs.gov/common-terms-job-opportunity-announcements.

[4] Wellons, J., J. Johnson, 2011 *A Grounded Theory Analysis of Introductory Computer Science Pedagogy*, Journal on Systemics, Cybernetics and Informatics, vol.8 2011.

[5] Deemer, Pete, et al., 2010, *The Scrum Primer* DOI=http://assets. scrumtraininginstitute.com/downloads/1/scrumprimer121.pdf

[6] Sam Chung, Daehee Won, Baeg, M.H., Sangdeok Park. Jan. 2009, *Service-oriented reverse reengineering: 5W1H model-driven re-documentation and candidate services identification*. Service-Oriented Computing and Applications (SOCA), 2009 IEEE International Conference

[7] A. Freeman, 2012, *Pro ASP.NET MVC 4, (4th Ed.)*, Apress, New York

[8] Bellware, Scott. June 2008, *Behavior-Driven Development*. Code Magazine.

[9] Peter Mell, Timothy Grance. September 2011, *The NIST Definition of Cloud Computing*. National Institute of Standards and Technology Special Publication 800-145, 7 pages.

[10] MSDN, *Threat Modeling*, DOI= http://msdn.microsoft.com/en-us/library/ff648644.aspx