Teaching DevOps and Cloud based Software Engineering in University Curricula

[1]Yuri Demchenko, [1]Zhiming Zhao, [2]Jayachander Surbiryala, [1]Spiros Koulouzis, [1]Zeshun Shi, [1]Xiaofeng Liao,
[3]Jelena Gordiyenko
[1]University of Amsterdam
System and Networking Lab
Amsterdam, The Netherlands
e-mail: {y.demchenko, z.zhao, S.Koulouzis, z.shi2, X.Liao}@uva.nl

[2] University of Stavanger
Department of Electrical Engineering and Computer Science, Stavanger, Norway
e-mail: jayachander.surbiryala@uis.no
[3] Agile Telecom
e-mail: jg4787@gmail.com

***Abstract*—** This paper presents recommendations on the design and pilot implementation of the DevOps and Cloud based Software Development curricula for Computer Science and Software Engineering masters. The central part of proposed approach is the Body of Knowledge in the DevOps technologies for Software Engineering (DevOpsSE BoK) that defines a set Knowledge Areas and Knowledge Units required for SE professionals to work efficiently as DevOps engineer or application developer. Defining DevOpsSE-BoK provides a basis for defining required professional competences and skills and allows consistent curricula structuring and profiling. The paper also reports on the experience of the first course run on 2018/2019 academic year at the University of Amsterdam. The paper presents the structure of the course and explains what instructional methodologies have been used for course development, such as project based learning that facilitates the students' team based skills both in mastering Agile development process and skills sharing. The paper provides a short summary of the generally used DevOps definitions, concepts, models and tools, specifically focusing on the cloud based DevOps tools for software development, deployment and operation that allows the main DevOps principle of continuous development and continuous improvement which are critical for modern agile data driven companies.

Keywords-* *DevOps and Cloud Based Software Engineering; DevOps Software Engieering Body of Knowledge; Education; Instructional methodology; Project Based Learning.

I. INTRODUCTION

Rapid technologies change and growing adoption of agile services and applications development models by modern organisation facilitates interest to and implementation of the DevOps methods and technologies that allow organisations to adopt the principles of Continuous Development – Continuous Integration – Continuous Delivery in their IT infrastructure and customer facing services which are foundation principles of DevOps [1, 2].

Although DevOps and Agile software development are known for a while, adoption and use of DevOps became really wide and common with use of the cloud powered software development [3] that benefits from high level automation of most of the processes in the applications development and operation lifecycle. Cloud Computing and related services enables the Infrastructure as Code (IaCode) model for integrating, deploying, managing cloud based services. Existing cloud platforms (both public such as Amazon Web Services (AWS) and Microsoft Azure, and Open Source such as OpenStack or CloudStack) provide rich functionality for applications deployment automation and cloud resources monitoring and billing. Clouds provide a native environment for deploying and operating Big Data applications providing also a basis for implementing the agile Data Science applications development currently powered agile data driven businesses.

DevOps and its cloud based implementation require advanced knowledge, competences and professional skills. In this context DevOps is often referred to as more culture than just collection of technologies. There is a growing demand for specialists with strong technical background and deep knowledge in DevOps Cloud Computing technologies. Despite strong demand from industry and hot job market, there is no established university curricula or programmes at the moment. DevOps training is provided by professional training organisations or big technology companies which in most cases also offer DevOps certification.

This paper presents recommendations on the design and implementation of the DevOps and Cloud based Software Development curricula for Computer Science and Software Engineering masters. The paper also reports on the experience of the first course run on 2018/2019 academic year at the University of Amsterdam. The central part of proposed approach is the Body of Knowledge in the DevOps technologies for Software Engineering (DevOpsSE-BoK) that defines a set of Knowledge Areas and Knowledge Units required for Software Engineering professionals to work efficiently as DevOps engineer or application developer.

The proposed BoK and model curriculum intends to provide an overview and analysis of the existing DevOps and cloud based software development models, approaches and best practices. The curriculum design outlined here presents a complete course on DevOps for Computer Science and Software Engineering masters. The proposed DevOpsSE-BoK is efficiently uses and extends the IEEE SWEBOK [4] and ACM/IEEE Computer Science BoK [5]. The presented model curriculum is structured in such a way that defines, in a form of lectures or learning modules, the main concepts and tools in DevOps, SE and Cloud Computing. Practical course configuration and content can be composed of the basic modules to target specific learners' groups or utilise existing resources.

The remainder of the paper is organized as follows. Section II explains need for an agile software development in modern companies and consequently growing demand for DevOps related knowledge and skills. Section III provides overview of existing DevOps definitions which are given by DevOps practitioners and expert. Section IV provides suggestions about defining DevOps and Software Engineering Body of Knowledge and its mapping to cloud based development and operational environment. Section V provides example of the DevOps and SE course by authors in the University of Amsterdam, and the paper concludes with the summary and future developments in section VI.

II. Demand for DevOps Competences and Skills and Education Gap

DevOps is a methodology for agile software development in an operational environment. DevOps became a major technology, approach and culture in applications and services development for modern agile data-driven companies. Modern technologies, particularly cloud and Big Data analytics, embrace DevOps methods. DevOps is often referred to the quick adoption of emerging technology and subsequent integration into production.

DevOps proposes a new emerging software and applications development model that realises the continuous integration (CI), continuous delivery (CD) and continuous improvement of the services and applications, that is deeply based on the cloud environment, on demand services deployment and highly granular services monitoring currently available on the major cloud platforms. Organisations such as Netflix, Facebook, and Google are revolutionising their IT infrastructure by using DevOps and moving services deployment towards clouds.

Demand for DevOps competences and skills is currently satisfied by professional training and certification provided by professional training organisations and big technology companies and software houses such as Microsoft Learn [7, 8] or AWS DevOps [9, 10]. There is limited offer from Coursera [11] and Udacity [12] that are focused primarily on some specific topics or platforms and don't provide consistent materials for the academic degree. Courses on DevOps and related software engineering and cloud technologies and tools are well presented at Linkedin Learn [13], which can be recommended for both students and teaching staff.

DevOps community is very committed and features number of expert and community supported websites, blogs and online forums. We can admit two valuable initiatives and services that offer training and certification on DevOps and also contribute to the definitions of the DevOps competences and Body of Knowledge:

- DevOps Institute: The Association for DevOps Professionals [14]
- DevOps Agile Skills Association (DASA) [15]

The DASA DevOps Fundamentals course provide an example of well developed course that can be a starting point for anyone involved in an Agile and/or DevOps team. The course is also supported with the DASA competence model and DevOps certifications scheme.

At the time of writing this paper, only one university course has been discovered at the John Hopkins Engineering part-time master program 605.609 - DevOps Software Development, which has been established in 2016 and is primarily focused on the DevOps concepts and tools overview [16]. The course lacks cloud technologies and tools orientation what is essential in modern DevOps. The course also less focused on practical skills development such as for example can be achieved with the project based learning to master both agile development process and team based development during the group project.

III. Definitions: What is DevOps?

A. *DevOps according to DevOps experts and practitioners*

DevOps term and concepts has been actively used since 2015 [17] and 2016 [18] when it introduced as extension of the Agile software development for modern Internet companies that besides agile software development need also continuously improve and adopt their services to continuously changing technology and market environment.

Formal and "live" DevOps definition is given at the community maintained website [1]: "DevOps is the practice of operations and development engineers participating together through the entire service lifecycle; from the design and development process all the way to production support. DevOps is also characterized by operations staff making use of many of the same techniques as developers for their systems work."

The following DevOps features are admitted [1]:

- DevOps has strong affinities with Agile and Lean approaches.
- DevOps can be interpreted as an outgrowth of Agile software development prescribes close collaboration of customers, product management, developers, and (sometimes) QA to fill in the gaps and rapidly iterate towards a better product
- DevOps is extending Agile principles beyond the boundaries of "the code" to the entire delivered service

B. *Relationship to Agile and Continuous Delivery [2]*

Agile and DevOps are closely related, however agile model is mostly applied for software development and represents a change in thinking and practice of the software development teams, while DevOps involves the whole lifecycle/process of the solution or service delivery and requires organizational and culture changes [2]:

- The need for DevOps was born from the increasing popularity of agile software development, as that tends to lead to an increased number of releases.
- One goal of DevOps is to establish an environment where releasing more reliable applications, faster and more frequently, can occur.

Continuous delivery and DevOps are similar in their meanings but they are two different concepts. DevOps has a broader scope, and centers around:

- Organizational change: specifically, to support greater collaboration between the various types of workers involved in software delivery:
- Automating the processes in software delivery what allows to make delivery of new application versions faster and more frequently, however preserving service continuity.

IV. Defining Body of Knowledge In Software Engineering and DevOps

Body of Knowledge (BoK) refers to several domains or operational categories into which the domain specific theory and practices breaks down. It is like a blueprint or framework which students and practitioners can see how things fit together.

When defining an initial set of BoK elements, also can be referred to as Knowledge Units (KU), that can be grouped into Knowledge Areas (KA) to indicate linking to similar concepts in CS-BoK [4], SWEBOK [5], or Data Science Engineering BoK (DS-BoK) [7] developed by the authors.

The following main BoK elements are defined for DevOpsSE BoK:

1. DevOps fundamentals, Continuous Integration, Continuous Delivery, Continuous Testing. Relation to other agile development technologies Lean, CAMS, and ITSM.
2. Organisational impact of DevOps, Digital Transformation and DevOps; DevOps Team structure and operation, Leadership, collaboration, and structured problem solving; success factors and key performance indicators.
3. Agile software development: Scrum, Kanban, Kaizen; Agile Scrum process and team management; Role of multi-disciplinary feature teams. Agile Manifesto.
4. DevOps Tools and Processes: CI/CD pipeline; DevOps Toolchain; Coding, versioning, collaboration and testing. Versioning and team based development, Git, Automated testing.
5. DevOps Practices and Platforms: Software packaging, Containerization. Container technologies, Kubernetes, Tools
6. Cloud Computing Architectures, service and deployment models, Cloud IaaS and Infrastructure as Code, Cloud economics
7. Cloud powered software development: AWS example and tools
8. Cloud monitoring tools and Operate for Development concept.
9. Cloud Automation Overview: Cloud based tools (e.g. AWS CloudFormation, Azure ARM); multicloud tools Chef, Puppet, Ansible, Terraform, others.
10. Popular cloud automation tools: Chef, Puppet, Ansible (depending on available platform)
11. Cloud Security Architecture and Models, Cloud compliance, CSA Consensus Assessment Initiative Questionnaire and Certification.
12. DevSecOps; Secure software development lifecycle; cloud based tools for secure Software development.

Figure 1 shows the relations between the DevOps pipeline components and the cloud based services and resources used for deploying and operating customer/user applications. The diagram also provides examples of the DevOps tools used at each pipeline stage.

V. Implementation: DevOps and Cloud based Software Engineering at the University of Amsterdam

A. General information and motivation

The DevOps and Cloud based software development course is taught as a part of the Master Software Engineering and counts for 6 credits. The course runs 2 months and has 8 contact hours per week of which 4 hours are lectures and 4 hours practice and labs.

This course discusses the basic concepts and technologies of DevOps, including its philosophy, workflow, monitoring methods and tools. The course gathers the recent development in cloud based services and applications development, deployment and operations based on the best industry practices. Students have opportunity to apply these concepts to understand how they can be best implemented to automate development, test, and release practices.

The course also includes the final projects to design cloud based application or service to support a selected business process or scientific workflow using one of the available cloud platforms: local cloud testbed, Amazon AWS, or Microsoft Azure. The project groups consist of 4-5 members and are required to function as small Agile teams using Agile Scrum model (optionally Kanban or Kaizen, if motivated by the project team).

B. Learning outcomes

The DevOps course defines the following Intended Learning Outcomes (ILO)

- Understand how to build cloud based applications and use cloud automation tools in the DevOps process.
- Understand the theoretical background of DevOps and cloud automation techniques.
- Demonstrate the application of these techniques applied to a range of applications and software development scenarios/processes.
- Understand functionalities offered by the popular DevOps and cloud automations tools such as Chef, Puppet, Ansible, SlipStream
- Have hands on experience with one of such tools as a result of a project completion.

Not formal course goal is to prepare the student to be ready to take one of DevOps related certifications exams.

The lab assignments include the basic hands-on tasks to learn the cloud platforms services and existing DevOps tools and services including cloud monitoring, cloud automation. The practice can be done on one of two major cloud platforms by Amazon AWS and Microsoft Azure, what together provides a basis for the practical project realization. Selection of public clouds is essential for the students' future workplace alignment as majority of software development companies are using public cloud platforms.

C. *Lectures*

The course includes two general types of lectures. First type is technology introduction lectures that provide extended overview of the basic technologies used in modern DevOps environment and practice, including DevOps concept and models, cloud technologies and services, CI/CD tools, cloud automation and monitoring tools, security, access control, compliance. Second type of lectures is more focused on use cases, practices, and case studies, where real life projects and development are used as examples.

Technology oriented lectures are provided at the beginning of the course and typically supported by practice and labs. The students are required to submit reports on practice, which contributes to final grade on course. Lectures on practices and case studies are more interactive and require students' involvement into case studies discussion.

The presented below example of the lecture topics reflects such approach. Guest lectures by practitioners have specific practical orientation and offered at the end of course.

Lecture 1: Introduction DevOps: Core concepts and models in DevOps, Continuous Delivery (CD), Continuous Integration (CI), Continuous Deployment

Lecture 2: DevOps and Agile development process: DevOps relation to Agile, Scrum, Lean, ITSM, Best practices

Lecture 3. Cloud service and Cloud Economics. Cloud resources management, monitoring and software assessment, applications deployment, accounts management

Lecture 4: DevOps Tools and Processes: Service oriented application development. Web service architecture, Micro services, Standards, Tools. Coding, versioning, collaboration and testing. Versioning and team based development, GIT, Automated testing.

Lecture 5: DevOps Practices and Platforms: Software packaging, Containerization. Container technologies, Technologies, Tools

Lecture 6: Cloud powered Software Engineering. DevOps and software development on major cloud platforms: AWS/CloudFormation, Microsoft Azure/Azure Resource Manager (ARM)

Lecture 7: Cloud Automation: Overview Chef, Puppet, Ansible, SlipStream

Lecture 8: Deep look into Ansible: Organisation, playbook and applications management process. Configuration management, Environment setups in AWS.

Lecture 9: Software Delivery Lifecycle in an agile DevOps organisation: QA in cloud based software development systems. Software security and Security Development Lifecycle management (SDLM) in clouds.

Lecture 10: DevOps for Time critical cloud applications: DevOps tools for infrastructure planning, Critical path algorithm, Infrastructure planning; Case studies.

Lecture 11: DevOps for Time critical cloud applications: Time critical provisioning and deployment; Deployment algorithms, time critical algorithms; Monitoring, adaptation, Scaling out/up.

The course reserves time for guest lectures that are given by cooperating industry partners and experts from local or international companies. Below is example of guest lectures from the 2018-2019 term:

Guest Lecture 1: Agile technologies, Scrum practical exercise (given by practicing Scrum Master)

Guest Lecture 2: Microsoft Azure DevOps and tools (Microsoft Azure DevOps Team)

Guest Lecture 3: DevOps practices and Enterprise Infrastructure cloud migration (Sogeti Cloud Integrator company)

Guest Lecture 4: SlipStream Cloud Automation Platform and Data Containers (SixSq, Switzerland) [17]

D. *Practice*

Practice in this course is organised by weekly topics to allow students sufficient time to master the technology and do reporting. The presented below topics are selected to gradually provide necessary knowledge and experience for the students to start the project development which is the main reporting material of the course.

Practice 01: Getting started with Cloud: AWS console and CLI, VM deployment, Accounts and security configuration.

Practice 02: Continuous Integration and Continuous Deployment process and tools on the example REST Web Services project

Practice 03: AWS Monitoring tools: Configure monitoring for deployed service.

Practice 04: Use AWS Cloud Formation to create simple template for deployed service: Adding AWS monitoring to CloudFormation template

Practice 05: Ansible with AWS: Create Ansible playbook

Depending on available resources, platforms and local preference the practice can include other cloud based DevOps platform such as Azure DevOps, or cloud automation tools such as Chef, Puppet of Slipstream.

E. *Self-study and seminars*

The course is built in such a way as to facilitate the students deeper study/research into the subject of DevOps and related technologies and tools, improve their skill in working with professional and scientific literature, facilitate collaborative learning and actively use their knowledge in the discussion.

To achieve this the course employs two types of self-study activities: (1) discussion of the selected papers in the classroom which is organised as a seminar at the second part of course when the student acquired sufficient knowledge to critically assess the papers content; and (2) literature based research on the topics related to the group project, the report on the literature study is included in the final project report, and contribute to the final course grade. The requirements to the literature study are described in the course guidelines.

F. Project development

The final project, including literature study, is the key components to achieve higher level of knowledge in the DevOps and Software Engineering. The project topics are selected by the students themselves and course guidelines specify criteria to the project content and complexity criteria. In particular, the project should demonstrate the CI/CD process and provide facility for the service or application monitoring. Such topics as, for example, Content Management System (CMS) typically used for website content management is not sufficient.

VI. Conclusion, Lessons Learned and Future Developments

This paper presents the authors experience on defining and teaching the DevOps and Software Engineering course as a part of the Computer Science or Software Engineering masters [19]. The paper explains major aspects of DevOps in modern software development practice and how they can be implemented in the university environment. The presented course is one the pioneering course, it is not a common practice yet to provide such strongly technology oriented and technology dependent courses in university curricula that could catch the fast development of both theory and practice of the software development and continuously evolving development and operational/production platforms facilitated by cloud technologies that allows programmability of all components of modern IT infrastructure.

The course development and implementation have re-used the authors' previous experience in developing effective curricula for Cloud Computing [20] and Big Data [21] that at the time of their development and implementation faced similar challenges of the new emerging technologies that had limited academic implementation.

The successful first run of the course is a result of contribution of the whole course management team that consisted of teachers, researcher and PhD students with actually complementary experience. Interesting to mention that the course management team has also implemented an Agile Kanban model as contrary to Scrum, because the main focus of the team was developing, testing and updating each next weekly set of lectures and practical or lab assignments and activities. The fact that course actively used AWS Educate Class Environment that was not sufficiently document, created a number of issues that need to be adopted in the process of course delivery.

We expect that after the second run, the course will be ready for wide sharing and profiling for other technology and science domain that all face need for agile development and extensive use of cloud based technologies and platforms, first of Data Science that is moving to DataOps agile model for development and operation of data driven applications and business models. Wider course acceptance can be also facilitated by the formal definition of the DevOps Software Engineering Body of Knowledge as a part of the Computer Science or Software Engineering BoK, what is one of the authors focus in the further course development.

References

[1] What is DevOps? The Agile Admin [online] https://theagileadmin.com/what-is-devops/

[2] DevOps, Wikipedia [online] https://en.wikipedia.org/wiki/DevOps

[3] Louis Columbus, Jinesh Varia, Building Powerful Web Applications in the AWS Cloud [online] https://d1.awsstatic.com/whitepapers/AWS_Cloud_Best_Practices.pdf

[4] Software Engineering Body of Knowledge (SWEBOK) [online] https://www.computer.org/web/swebok/v3

[5] ACM and IEEE Computer Science Curricula 2013 (CS2013) [online] http://dx.doi.org/10.1145/2534860

[6] Data Science Body of Knowledge (DS-BoK). EDSF Community Initiative [online] https://github.com/EDISONcommunity/EDSF/tree/master/data-science-body-of-knowledge

[7] Microsoft Learn [online] https://docs.microsoft.com/en-gb/learn/

[8] Azure DevOps [online] https://azure.microsoft.com/en-gb/services/devops/

[9] AWS Training [online] https://www.aws.training/

[10] AWS DevOps [online] https://aws.amazon.com/devops/

[11] DevOps at Coursera [online] https://www.coursera.org/courses?query=devops

[12] DevOps at Udacity [online] https://eu.udacity.com/courses/all?keyword=devops

[13] Linkedin Learning [online] https://www.linkedin.com/learning/

[14] DevOps Institute: The Association for DevOps Professionals [online] https://devopsinstitute.com/

[15] DevOps Agile Skills Association (DASA) [online] https://www.devopsagileskills.org/certifications/dasa-devops-fundamentals/

[16] John Hopkins Engineering part-time master program 605.609 - DevOps Software Development [online] https://ep.jhu.edu/programs-and-courses/605.609-devops-software-development

[17] Ernest Mueller, DevOps State of the Union 2015, Slideshare [online] https://www.slideshare.net/mxyzplk/devops-state-of-the-union-2015

[18] Jason Bloomberg, The Agile Architecture Revolution: How Cloud Computing, REST-Based SOA, and Mobile Computing Are Changing Enterprise IT [online] https://intellyx.com/author/jbloomberg/

[19] DevOps and Cloud Based Software, Software Engineering at UvA [online] http://studiegids.uva.nl/xmlpages/page/2018-2019/zoek-vak/vak/63471

[20] Demchenko, Yuri, David Bernstein, Adam Belloum, Ana Oprescu, Tomasz W. Wlodarczyk, Cees de Laat, New Instructional Models for Building Effective Curricula on Cloud Computing Technologies and Engineering. Proc. The 5th IEEE International Conference and Workshops on Cloud Computing Technology and Science (CloudCom2013), 2-5 December 2013, Bristol, UK.

[21] Demchenko, Yuri, Emanuel Gruengard, Sander Klous, Instructional Model for Building effective Big Data Curricula for Online and Campus Education. 1st IEEE STC CC and RDA Workshop on Curricula and Teaching Methods in Cloud Computing, Big Data, and Data Science, in Proc. The 6th IEEE International Conference and Workshops on Cloud Computing Technology and Science (CloudCom2014), 15-18 December 2014, Singapore.