

# Towards Utilization of Lean Canvas in the DevOps Software

Padmaraj Nidagundi, Leonids Novickis

Division of Software Engineering, Riga Technical University. Address: Riga, Latvia.

**Abstract.** *The growth of technology made human to depend more on the software applications in his daily life and nowadays software companies focused more on building robust error free software to end customers in very short time. Software development companies facing one side growth of technological complexity and another side build the products fast to win a competition in business. In recent years growth of a DevOps given lot of new growth opportunity for the software companies.*

*DevOps basic principles focused on the collaboration and communication as a key in between software development information technology professional. It is concentrated on the automating the most of the routine tasks such as development, delivery, infrastructure, support, software testing in software development process. DevOps also emphasize on the building, testing and releasing the software more quickly and in a reliable way.*

**Keywords:** *devops testing, software testing canvas model design, software testing.*

## I. INTRODUCTION

In daily life, growing dependency of human on software embedded devices made a human more flexible to manage daily tasks. In recent years software development tools, technology and process changed a lot, it affected directly or indirectly traditional software development process. From the last few years, DevOps become buzzwords in the software development industry and most of the companies gradually focusing their projects adopting towards the DevOps to build the software fast and reduce the cost [1]. DevOps is an approach based on the lean and agile principles, practices in which teams are working in a collaborative way and communicate [1] more constantly to deliver the product or services in a continuous manner to get fast customer feedback and fulfill their needs. Considering DevOps toolchain [9], it involves the develop a code, build, test, package, release, configure, monitor. DevOps concepts are more emphasized more on the communication, collaboration and integration between all teams [2]. This entire process bridges the gap in between development, operation and quality assurance teams contribute to one platform to speed up software development and delivery process with the help of cutting edge tools and automation process.

The core lean process [3] focus on the removing unwanted things in process, tools and improve the effectiveness of the development process, this can impact straight on the process deliver speed and help to reduce the development cost. Lean can influence directly to traditional development and documentation methods in a process and more than

that, it can bring new ideas for the customer / stakeholder-centric software development. A short time of delivering the software and cost saving for the overall project makes “lean” as the best strategy for software development.

In recent years, the word “lean” not only limited manufacturing it now also used in software companies to get rid of the waste in process and tools. The same lean concept also used by Ash Maurya for the lean canvas design for the business. Using the lean canvas for the business, anyone identifies the or create their own business model, business plan, possible solutions, key metrics.

In DevOps software development software validation and verification play a key role to fulfill the customer expectation with the error free software. The develops focused more on using the automation in testing using possible cutting edge tools, in this process for improving the testing process, we can adopt the lean canvas board to overcome many problems.

## II. THE PROBLEM STATEMENT

In DevOps process, software verification and validation play a key role to ensure that the software is working according to the requirements. Quality assurance, tester role can be merged in either development or operation teams, most of the time they merged with the development team so they can work in parallel [4] [15].

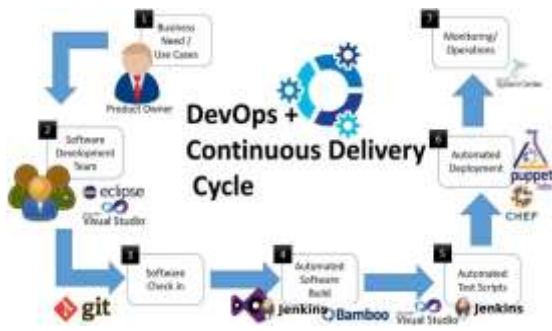


Fig. 1. Old SDLC and New DevOps SDLC overview

The most common DevOps challenges [11], [14], [15] related to software testing are:

Continuous integration: tester may have lesser knowledge about tools such as SVN, GIT, Docker,

Testing in Agile	Testing in DevOps
Test as early and as often as possible	Test continuously
Automate testing as much as possible	Automate almost everything
Continuous integration and testing is a step forward	Continuous integration and testing is mandatory
Potentially shippable code at the end of a sprint	Potentially shippable code following every integration

Fig. 2. DevOps continuous delivery cycle overview

Jenkins for code management and unaware of continuous integration practices.

Continuous testing: tester may find himself not educated with 100% automation for testing and at the end user notice still software bugs found after continuous testing using automated tools and process such as Test-driven development (TDD) & Acceptance test driven development (ATDD).

Configuration management: tester may not be able to use continues integration (CI) tools such as Chef, Puppet or Ansible to create the run-time environment according to different requirements.

Continuous delivery [10] or continuous deployment: tester can notice many repeated bugs due to continuous delivery or continuous deployment in the code and face difficulties with managing test automation frameworks.

Cultural change: tester needs to have many, mixed skillset and utilization of several tools such as Vagrant as a virtualization platform.

Test strategy: hard to define the test strategy from beginning of the project.

Test estimation: human resource estimation & testing time estimation for the project may get wrong.

Test planning: creating simple test plans in complex testing environment.

The biggest challenge company need to address in two main areas [14], [15], first developers need to look over more on production system and operation

team need to learn and implement to program the system.

### III. SCIENTIFIC NOVELTY OF PAPER

The business lean canvas evaluates the business model, it's a simple white board blocks design with several segmentations on it that help to identify the business plan to marketing strategy to end customers. Lean business canvas is a one-page document with several blocks. It is a template that helps to describe the business and its overall strategy and nowadays it's most of the time used in stats up, business case design and in restructuring the existing products.

Old SDLC	New SDLC
Directive	Adaptive
Task-oriented	Goal-directed
Specialized roles	Empowered teams
Resistant to change	Optimally responsive
Outsourced	Automated
Project optimization	Portfolio optimization

Fig. 3. DevOps vs Agile overview

The blocks of lean canvas start with problem definition, key metrics, customer segments and unique value proposition.

Use the lean canvas model as a base our core contribution is to focus on the finding the test metrics and investigating how we can adopt the lean canvas models in develops software test process.

Specifically:

Identifying and finding the suitable lean canvas porotype in DevOps.

Improving the DevOps test planning with help of lean canvas.

Simplifying the DevOps software test strategy.

Utilization of lean principles and lean canvas as the base model for the design of the new DevOps compatible lean canvas.

Find the appropriate lean canvas blocks in the design of the lean board.

#### A. Related research

Alexander Osterwalder and his co-authors in early 2000 published the business model canvas, and it is very attractive for the plan of activities and strategy management for any business [6].

The life cycle of the lean canvas is starting with the idea, followed by building, product, measure, data and learn this continues the process.

#### B. Lean canvas core principles

The core principles are used to identify the risk part of the created plan for the specific project.

Developing a document for your project idea/plan.

Analyze/Measure and remove the waste process in your project plan.

Continuous repeating test cycles for idea/plan.

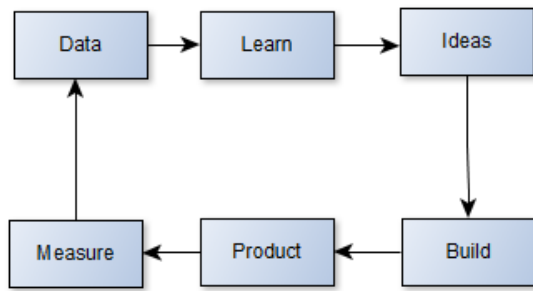


Fig. 4. Lean canvas life cycle

As above showed in fig 4, lean canvas life cycle, following steps - Ideas, build, product, measure and learn, we can recognize and generate the lean canvas model for the business. Now from gathering all core life cycle and lean principals able to generate the similar teams for the DevOps testing.

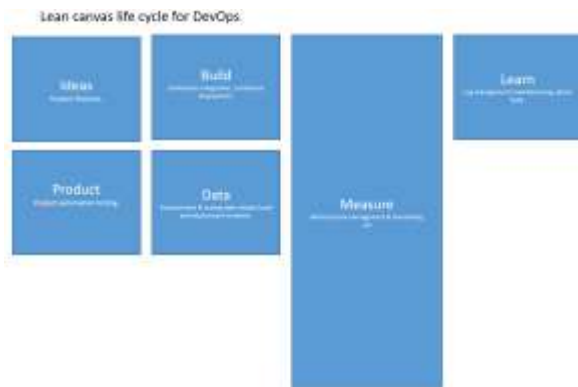


Fig. 5. Prototype of lean canvas life cycle for DevOps

- Ideas - product features
- Build - continuous integration, continuous deployment.
- Product - automation testing.
- Measure - infrastructure management & monitoring.
- Data - development & testing data related build and deployment incidents.
- Learn - log management, load balancing, about tools.

#### IV. DEVOPS SOFTWARE DEVELOPMENT LIFE CYCLES & LEAN TEST LIFE CYCLE

In DevOps, testing quality engineering is closely coupled [8] with software engineering and operations teams. In such environment software testing is not at the end of the release cycle it merged with beginning of the development. Development team and system engineers team make the code available in the appropriate environment and testing team start validating the software.

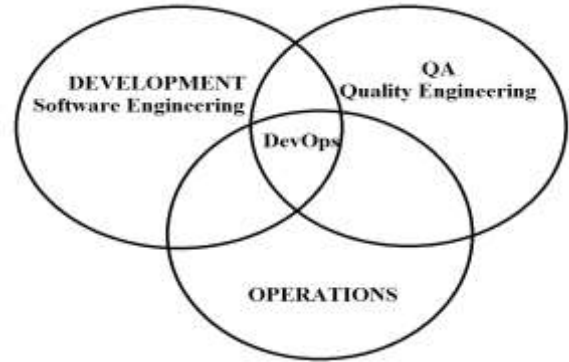


Fig. 6. DevOps venn diagram

Testing team needs to work on action bases testing to align the test design, test automation and test case



Fig. 7. Lean canvas life cycle

development and need to fix the code changes to make sure do not break the product test cases.

All seven lean principles [5], we try to classify different metrics regarding DevOps.

**Transport** – In the DevOps testing process, the tester needs to stay in an ideal mode when nothing to test and not considering for to take other tasks. DevOps say nothing about what a person can do in his free time within the project.

**Inventory** – If the test automation [11] is failed or under testing any item, then tester not able to confirm about software quality.

**Motion** – In DevOps test team human resources are frequently changing due to any reason, then replacement of knowledgeable team member required additional processing.

**Waiting** – Changing testing scope with reference to project development requirements and agile environment, tester need to wait for software delivery for the test.

**Overproduction** – In DevOps tester can finish his task early then as before planned.

**Over processing** – It is possible may be due to the wrong estimation of the test planning or selecting the not appropriate tools for the testing.

Defects – In DevOps lack of having appropriate test environments and failing test cases in automation with no reason can raise more software logs and bugs.

*A. Identifying Relevant Test Metrics for Lean Canvas Board*

Considering above lean principles, we able to recognize different possible test metrics in the DevOps software testing. In DevOps, testing test metrics can be further utilized as an indicator for the lean canvas white board design. Although identified key metrics name titles are not standard [7], in further, we can use rename and use according to the subject.

Test metrics raw data come mainly from process, tools or documentation part and these will be further improved with lean core principles [16] [17].

**V. DEVOPS TESTING PROCESS WASTE IDENTIFICATION USING LEAN PRINCIPLES TO DETERMINE RIGHT TEST METRICS**

Software quality is key component to make software product successful and bridge the gap product and user satisfaction. In DevOps, testing quality engineering is closely coupled with software engineering and operations teams.

The study explains the possibilities for the utilization of the lean canvas design in the DevOps for software testing purpose this can impact on the software testing process and improve the software quality. The figure 7 show us what are the possibilities, in the input phase considering all related terms where testing involved in develops lifecycle will extract the needed basic test metrics. Transformation

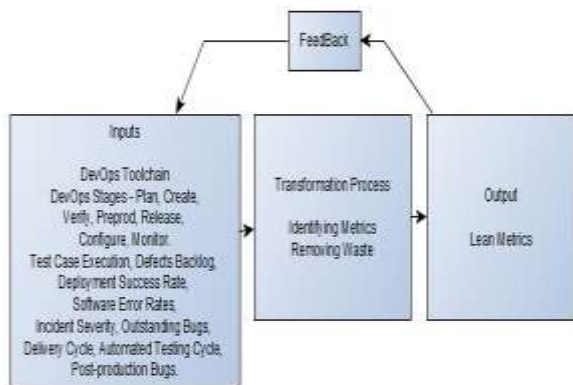


Fig. 8. DevOps test metrics identification.

phase will optimize metrics removing the waste from it with core lean principle. Third stage give us lean metrics those are more meaning full for further use for design of the lean canvas. In this process feedback, will be always there to improve metrics with each process.

Example test metrics [12] are test case creation, test case execution, defects backlog, deployment

success rate, software error rates, incident severity, outstanding bugs, delivery cycle, automated testing cycle, post-production bugs, toolchain, DevOps stages - plan, create, verify, preprod, release, configure, monitor. Such metrics can be extracted from availability of data and these metrics will be used to build the lean canvas board [13]. Collective and individual metrics will give detailed information in each step DevOps.

**VI. CONCLUSION AND FURTHER RESEARCH**

To continue the design testing lean canvas for the DevOps, it is necessary to accomplish the following research activities.

Need to carry out an experiment in DevOps for identifying the lean metrics in a testing process.

Identified lean metrics need to be tested to define the criteria.

Need to find out more about transformation models.

Design appropriate prototype for the DevOps lean canvas.

Need to develop algorithms to optimize the lean metrics.

Need to investigate with practical wok test metrics fit with input considerations.

Need to develop the framework and tool these can be used for development of the lean canvas board according to DevOps testing process.

Need to analyse and build the road map for one page

DevOps software testing canvas model design from the experiment.

The new idea and approaches proposed in this article emphasize on rethinking about traditional approach about long test documentation method in software development life cycle. The lean canvas one page documentation design strategy can save time and money with software testing.

**REFERENCES**

- [1] Diel L., Marczak S., Daniela S. Cruzes. Communication Challenges and Strategies in Distributed DevOps. E, IEEE 11th International Conference on Global Software Engineering. (ICGSE), 2016, pp: 24-28.
- [2] M. Rajkumar., Pole AN., Adige V., Mahanta P. DevOps culture and its impact on cloud delivery and software development. International Conference on Advances in Computing, Communication & Automation (ICACCA), 2016, pp: 1-6.
- [3] Janes A. A guide to lean software development in action, Software Testing, Verification and Validation Workshops (ICSTW), 2015 IEEE Eighth International Conference on, 2015, pp 1-2.
- [4] Soni M. End to End Automation on Cloud with Build Pipeline: The Case for DevOps in Insurance Industry, Continuous Integration, Continuous Testing, and Continuous Delivery. IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), 2015, pp 5-89.
- [5] Nidagundi P., Novickis L. Introduction to Lean Canvas Transformation Models and Metrics in Software Testing. Applied Computer Systems. Nr.19, 2016, pp 30-36.
- [6] Mohammad A., An Empirical Study on Lean and Agile Methods in Global Software Development. IEEE 11th

- International Conference on Global Software Engineering Workshops (ICGSEW), 2016, pp 61-64.
- [7] Nidagundi P., Novickis L. Introducing Lean Canvas Model Adaptation in the Scrum Software Testing. ICTE 2016, Riga Technical University. Applied Computer Systems. 2016, pp97–103.
- [8] Justin Rohrman "How Testing Fits Into DevOps, Because It's Here to Stay" (2015).
- [9] Richard Seroter "Exploring the ENTIRE DevOps Toolchain for (Cloud) Teams" (2014).
- [10] Cameron Cosgrove "Can you build software faster, cheaper and better?" (2016).
- [11] "World Quality Report 2015-2016 - QUEST" (2016).
- [12] Asami Novak "Measuring DevOps: Metrics That Define Software Quality" (2016).
- [13] Nidagundi P., Novickis L. Towards Utilization of a Lean Canvas in the Testing Extra-Functional Properties. 2017, pp-349.
- [14] Challenges of Implementing DevOps Testing & How To Deal With Them (2016).
- [15] "Jumpstarting DevOps with Continuous Testing – Cognizant" (2015).
- [16] Nidagundi P., Novickis L. Towards Utilization of a Lean Canvas in the Biometric Software Testing. IIOAB Journal - Institute of Integrative Omics and Applied Biotechnology. 2017, pp32–36.
- [17] Lincke, R., Lundberg, J. & Löwe, W. (2008). Comparing software metrics tools, Proceedings of the 2008 international symposium on Software testing and analysis, ACM, pp131–142.