

Innovative Corporate Structure DevOps real case implementation

SEMESTER PROJECT
ENTREPRENEURIAL ENGINEERING
AALBORG UNIVERSITY
03/06/2019



**Eighth Semester at School
of Engineering and Science**

Entrepreneurial Engineering

Fibigerstræde 16

9220 Aalborg Øst

<https://www.aau.dk>

Project title:

Innovative Corporate Structure

DevOps real case implementation

Project:

Master Thesis

Project Period:

01. Feb 2019 - 04. Jun 2019

Author:

Krzysztof Tadeusz Kruk

Supervisor:

Frank Gertsen

Co-examiner:

Michael Vaag

Pages: 61

Appendices: A, B

Submission Date: 03-06-2019

Synopsis:

DevOps becomes more and more popular organizational structure. Its values are applied in a growing number of companies in a software development industry. However, its proper implementation is dependent on many aspects and should be carefully observed. This master thesis has on its purpose an investigation of the impact of this structure application and identification of a possible ways of its further supervision. Surveys among the employees working with DevOps were done in order to gain a quantitative data to investigate a results and a quality of DevOps implementation in ILS Sp.z.o.o.

Acknowledgements

This project was enabled thanks to the collaboration with Intelligent Logistic Solutions Sp.z.o.o. I would like to express my gratitude to the company executives who used their time and knowledge, which enabled me to bring this value to the project. Therefore, thanks to:

- Robert Pławiak, CEO of ILS Sp.z.o.o.
- Katarzyna Żabicka, Business Line Director in ILS Sp.z.o.o.

Furthermore I would like to express a special thanks to my supervisor, Frank Gertsen who helped me to navigate through a conceptual frames and fill theoretical requirements of this thesis.

Preface

During three first semesters at Aalborg University I had an opportunity to learn various topics related to entrepreneurship. I had an ability to get better picture on how does the entrepreneurship works. This knowledge let me to look differently on how does an innovation drives company's development and give it a competitive strength. Very often innovation is being identified with a product of the company. But what if a product itself is already innovative because of its nature, like in the case of software development? Probably an innovation in that market that is hidden in the business processes of a company and in its structure will enable to improve a company's market position. With this master thesis I decided to research the topic of an influence of an organizational structure on a performance of a company.

Executive Summary

This master thesis project included a research of the topic of DevOps influence as a organizational structure on a company's performance. This project was based on three activities.

Theoretical part included a review of a literature and theories related to the topic of DevOps. This research included its techniques, values and methodologies.

Thanks to the collaboration of the author with the ILS Sp.z.o.o. there was possible a practical part of the thesis were observations of DevOps influence on the performance of the company were done. In order to explore this influence the author made a surveys among the ILS employees in order to investigate their impressions about the impact of a new structure on their work.

During the project the author led discussions with the executives in order to define a proper tool for tracking the changes in their company from the time of a new structure implementation. After discussions it was decided that Key Performance Indicators (KPI) will enable that functionality. Therefore a practical part of this thesis included a design of a customized set of KPI that will enable to the executives to track changes in their company.

Table of content

Acknowledgements	iii
Preface	v
Executive Summary	vii
1 Introduction	1
2 Research Question	3
3 Methodology	4
3.1 Action Research	4
3.1.1 Benefits of Action Research	7
3.2 Literature Review (LR)	7
4 DevOps research	9
4.1 Introduction to DevOps	9
4.2 DevOps definition	9
4.2.1 Values	10
4.2.2 Principles	11
4.2.3 Processes	11
4.2.4 Methods	12
4.2.5 Practices	12
4.2.6 Tools	12
4.3 Methodology of DevOps	13
4.3.1 Scrum	13
4.3.2 Kanban	13
4.3.3 Visable Ops	14
4.3.4 Incident Command System	14
4.3.5 Agile Software Development	15
4.3.6 Lean Thinking	16
4.4 Most frequently improved indicators	18
4.4.1 Deployment frequency	18
4.4.2 Speed of deployment	18

4.4.3	Deployment failure rate	18
4.4.4	Mean time to detect (MTTD)	18
4.4.5	Mean time to repair (MTTR)	19
4.4.6	Customer satisfaction	19
4.5	Importance of DevOps	19
5	ILS as a company	21
5.1	ILS as part of Pelion S.A.	21
5.2	Intelligent Logistic Solutions S.A.	22
5.3	Mission of ILS	22
5.4	Values of ILS	23
5.5	ILS field of expertise	23
5.6	ILS structure	24
6	DevOps Implementation - ILS case	26
6.1	Implementation process	26
6.2	DevOps team structure	28
6.3	Expected improvements	29
6.4	Key Performance Indicators (KPI)	31
6.4.1	Initial intention of the project	31
6.4.2	KPI introduction	31
6.4.3	ILS custom KPI set	32
6.5	KPIs description	35
6.5.1	Time To Market (TTM)	35
6.5.2	Number of Deployments	36
6.5.3	Mean Time To Repair (MTTR)	37
6.5.4	Number of failures	38
6.5.5	Customer satisfaction	39
6.5.6	Number of opened projects	39
6.5.7	Estimated time deviation (ETD)	40
6.6	Technical implementation execution	41
6.7	Map of KPIs relations	41
7	Empirical Data Collection	45
7.1	Survey for all employess	45
7.2	Survey for the management team	47
7.3	Expected results	48

8 Empirical Results	49
8.1 Survey to all employees	49
8.1.1 <i>Whats your age?</i>	49
8.1.2 <i>What is your role in the company?</i>	50
8.1.3 <i>Are you aware of DevOps structure in your organization?</i> . . .	51
8.1.4 <i>How do you rate DevOps influence on the communication in your team?</i>	52
8.1.5 <i>How do you rate a DevOps influence on the speed of your work?</i>	53
8.1.6 <i>Do you feel that DevOps improves your work? How?</i>	54
8.1.7 <i>Explain briefly in your own words, what is DevOps about (how DevOps works)</i>	54
8.1.8 Conclusion	55
8.2 Survey to the management team	55
8.2.1 <i>How do you rate an impact of devops on the business results of your team?</i>	55
8.2.2 <i>How do you rate a devops influence on the communication in your team?</i>	56
8.2.3 <i>How do you rate a devops influence on the work speed of your team?</i>	57
8.2.4 <i>How do you rate a DevOps influence on the communication with your clients?</i>	58
8.2.5 Conclusion	59
9 Conclusion	60
Bibliography	62
A Survey to all employees	65
B Survey to the management team	69

Introduction

1

Many code production companies struggles with inefficient processes. A competition on a market of software development is extremely high. According to Jon Jenkins Amazon deploys their application codes 11,6 seconds a day average [Jenkins, 2011]. In order to keep a position on such high competitive market apps must be developed quickly and with a certain quality level.

In order to be able to increase an efficiency of code production which means to increase a speed of work of employees while keeping a quality of a code, a business processes must be improved. Business processes are needed in every company who wants to formalize their work and efforts an to have control on its efficiency. A design of those processes should be aligned to the individual needs of an entrepreneurship. However processes are not a standalone factor that impacts an efficiency.

The basis for these processes is an organizational structure of a company. For some time already, more and more companies implements DevOps as a structure of their organization as well as their values. DevOps is about shortening the distances between people in the company. This term is differently interpreted by various sources but DevOps in general is an organizational culture with a teams structure in its center. This structure should positively affect a communication among the employees and shorten an information flow in order to accelerate the speed of work. It is also about the aims alignment of two different teams: Dev (software developers) and Ops (Operations).

In Łódź, Poland, there is a company Intelligent Logistic Solutions Sp.z.o.o. which is a software support for their clients in a pharmacy industry. When their clients grew, and with them their needs, ILS noticed that they have problems with organizing their work efficiently. Therefore they decided to implement DevOps as an improvement to their work. The author of this thesis decided to investigate an influence of the new structure on the company's performance. He will investigate how did the most important factors, which should be improved after DevOps implementation, changed. He will also attempt to find an effective way for a constant

tracking of this influence on the company's performance.

The results of this project will support an identification of problems that are still ahead of ILS as well as to assess a usefulness of DevOps in a processes improvement initiatives.

Research Question 2

In the fast-paced environment of code production, companies face challenges to find sufficient flexibility in their organizational structure and responsibility delegation.

In order to find a method in those efforts, coding companies created many modern approaches in team building. That approaches and methodologies require validation to create a real foundation for navigation between them.

This master thesis is going to investigate the influence of the teams configuration on the business results of the company. It seeks to provide the understanding of the gains that can be achieved and the challenges to be overcome on the way through such journey.

Taking into consideration that goals, the following question is going to be investigated:

How does the new DevOps structure affect ILS business performance?

Methodology 3

3.1 Action Research

Action research is a set of research methods, where their common goal is to understand and accept the abilities and competencies of participants of particular society instead of imposing opinions. According to the Institute for the Study of Inquiry in Education [Sagor, 2000]:

Action Research is a disciplined process of inquiry conducted by and for those taking the action. The primary reason for engaging in action research is to assist the “actor” in improving and/or refining his or her actions.

In general the Action Research was developed as a tool for teachers in order to improve their learning techniques. However it can be also applied in the academic research, where there is a need of understanding the organizational processes.

In this methodology there is much higher importance given to an actor of a research instead of to a researcher. Action research is a process and it contains several steps, which together combines into a cycle (Figure 3.1).

An important attribute of action research is also that it is performed not only in order to describe a particular phenomenon, but also to solve its problems. Its purpose is a change, not only a research. That is a reason why it is called "action" research.

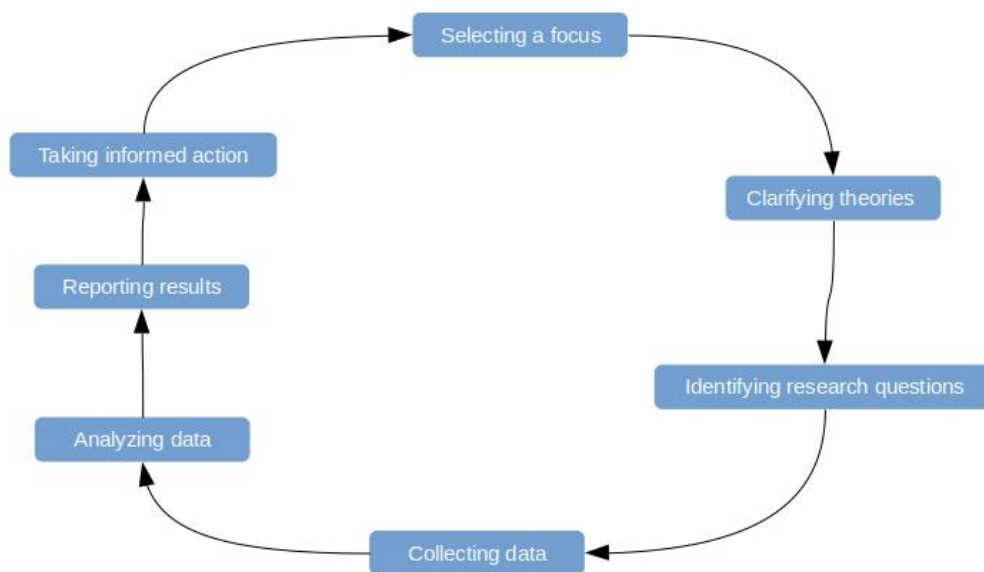


Figure 3.1: Action research process cycle

Selecting a Focus

The action research process begins with the important topic identification. It is an important step, because a wrong problem identification can result in a waste of time for a research. It begins with the question:

What element(s) of our practice do we wish to investigate?

In this research the author investigates the influence of the DevOps implementation to the business results of the company. That challenge requires an investigation of the practices that are performed in the company during that research.

Clarifying Theories

In the second step the researcher identifies the values, beliefs and theoretical perspectives that author holds in relation to their focus. It is important to clarify which theories are going to be investigated, and how.

In this research the author expects that DevOps can actually improve business

results of the company. He is going to investigate where lies the essence of this success (if any).

Identifying Research Questions

It is a conclusion of previous steps when researchers already know what they want to investigate and how. Based on that knowledge a proper research question can be formed to formalize the research efforts.

Collecting Data

Data collection is an important part of an action research. It is a professional approach to base important decisions and conclusions on a reliable data. Data can be gathered in many different ways:

- Interviews
- Journals
- Field notes
- Surveys
- Logs of meetings
- Case studies
- Questionnaires

In this research data collection is performed by author based on the close collaboration with the DevOps management team of the company, where this research is being performed. Data will be gathered in a form of notes from interviews and data sets measured by a system used for monitoring the company's performance.

Analyzing Data

Once the data are gained an analysis is required. A proper data analysis should answer two questions:

- What is the story told by these data?
- Why did the story play itself out this way?

By answering these question the author is going to achieve a deeper understanding of a conclusion underlying in the results.

Reporting Results

Sharing results of a research is usually one the main purposes of performing a research. Therefor after data analysis there comes a moment when results should be reported.

Collected data and its analysis are going to be reviewed in chapters 7 and 8.

Taking Informed Action

This step include planning actions that should be performed in order to achieve the expected results. It is based on the conclusions from previous steps. In this step, a researcher may see a need for a further research.

3.1.1 Benefits of Action Research

Using an action research in this project carries several benefits in relation to its topic.

First of all, it require a researcher to be a part of the society, group or team that is a subject of a research. The author of this research takes an active participation in the daily life of the company that implements DevOps. Using an action research the author can describe the real state of the problem, and focus on solution finding.

3.2 Literature Review (LR)

In order to organize efforts in terms of a theoretical par of this research, the Literature Review (LR) will be used. It is a method used in order to address a clearly formulated questions and identify resources like books, articles, blogs etc. that may contain answer or partial answer to those questions [University of Southern California, 2019].

A development of the literature review contains four steps (Figure 3.2 that could be referred to the Action Research steps. It starts with the problem formulation. With the properly formulated problem, a researcher can start an actual literature search, which leads to a data evaluation and its analysis.

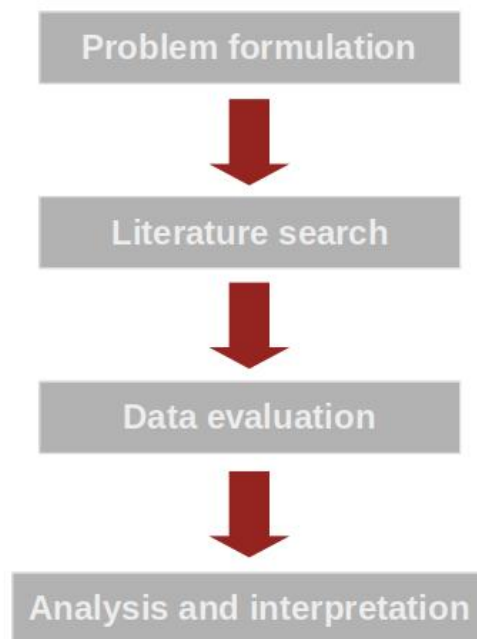


Figure 3.2: Literature Review process

This report is not a standalone Literature Review. It is used as an introductory part of a research, a methodology that enable the author to organize sources of information.

DevOps research 4

4.1 Introduction to DevOps

The name DevOps combines two words: Development and Operations. DevOps is a methodology applied in a team structure, in order to facilitate the collaboration between these two departments.

There is inconsistency in terms of goals of those teams. The development team is working in the environment of a rapid changes, they aim to build more and more of a new projects and products. They are rewarded for answering the needs of the market which is constantly changing. It is required from companies to keep changing and to introduce new technologies in order to satisfy their clients.

At the same time operations teams usually are driven by an opposite motivation. The business depends on them keeping all services and products available. Changes are the last thing they usually want. They are aiming their efforts to resist changes and sustain stability of the company's portfolio [Edwards, 2010].

Both teams are correct in their motivation. However that kind of teams structure results in inefficient working process. There is a strong need for a new team structure design, that will help to smooth out the relation between department. That need is an origin of DevOps as a new work environment.

4.2 DevOps definition

Many researchers suggest that DevOps is a wider application of Agile methodology.

According to Ernest Mueller's definition:

DevOps is the practice of operations and development engineers participating together in the entire service lifecycle, from design through the development process to production support. [Mueller, 2010]

That definition can be expanded for a six different levels of its implementation:

4.2.1 Values

DevOps values are aligned with Agile Manifesto [Kent Beck, 2001] which are:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

In DevOps case there is a higher emphasis on the phrase "Working software" and it could be even changed to "working software delivery to customer".

DevOps is deeply focused on interpersonal aspects. It depends on a collaboration and trust between the co-workers [Hüttermann, 2012]. That aspect of DevOps values are described by Michael Hüttermann:

- Respect for one another
- Commitment to shared goals
- Collective ownership
- Shared values

DevOps integrates two groups of workers, who are rewarded for achieving completely opposite goals. Therefore developers and operations teams should be treated as a one group. They should have given the same goal that they should aim to achieve. To gain those goals employees should have shared incentives as well as shared picture of quality and purpose. Thanks to those, they will be able to cooperate as a one group, not as a two fighting against each other teams.

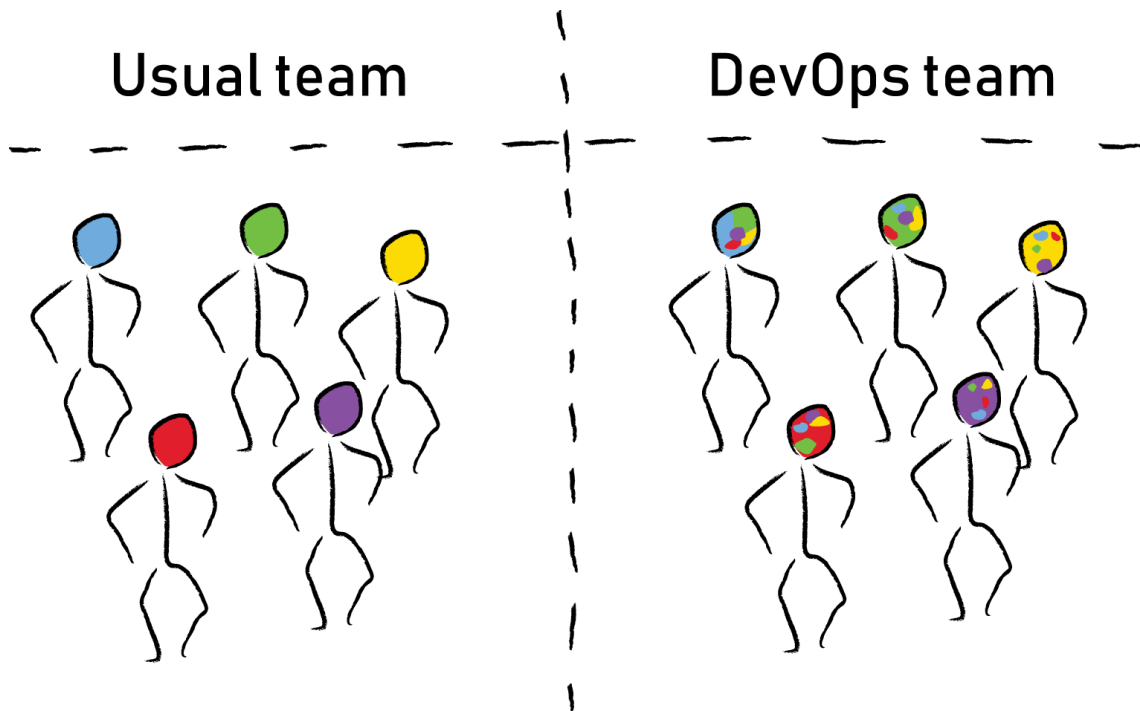


Figure 4.1: Knowledge sharing in DevOps

Team workers should also share their knowledge among each others. Naturally not everyone need to have a knowledge about each detail of a project. Each collaborator should has at least partial knowledge about the scope of a work of their colleagues from the team. They can achieve that by a collaborative problem solving. In case of a system failure all team members need to has at least general idea where the problem may lie and how to solve it.

4.2.2 Principles

There is no set of DevOps principles. As in many of these conceptual levels, it follows the principles of Agile Manifesto.

One of the most common principles in all researches is “Infrastructure as code” (IaC). It is the process of managing data centers through machine-readable file definitions, rather than physical hardware configuration or interactive configuration tools [Wittig and Wittig, 2016]. In that approach Developers are involved in the infrastructure’s development process earlier, which is aligned with DevOps values.

4.2.3 Processes

Processes defines the way the software is created and deployed. Very often processes are more important than the tools used to execute them. Using the same tools it

is possible to achieve the same goals. It is different with processes. Working in a different processes, the results may vary, and it is unlikely that two processes will end up with the same result. In DevOps, developers and operations share the same processes, because they has common goals.

4.2.4 Methods

There are many methods that can be used in DevOps implementation like:

- Scrum
- Kanban
- Visible Ops
- Incident Command System

The number of these methods is still growing. Methods like Scrum or Kanban are the same that work for Agile Methodology, but with more focus on development with operations integration.

4.2.5 Practices

There are three main practice areas in DevOps:

- **Infrastructure Automation**
That practice assume that corporation create its own systems, configuration, app deployment process in order to improve its performance.
- **Continuous Delivery**
Building apps and their deployment must be fast and automated.
- **Site Reliability Engineering**
The goals of this practice is scalability and reliability of software systems. It is a discipline that applies aspects of software engineering into operation problems.

4.2.6 Tools

There is uncounted number of tools that enable companies to apply above values, principles, methods and practices. Some of them are listed below:

- Jenkins
- Travis

- Teamcity
- Puppet
- Chef
- Ansible
- Cfengine

4.3 Methodology of DevOps

As it was mentioned before within DevOps many methods can be applied, that have their origins in other frameworks like Scrum or Kanban.

4.3.1 Scrum

Scrum is a framework for complex product development [Partogi, 2018]. It is a method applied in a project management process. It does not define how the development process should look like, and this is why it can be applied within DevOps. It relies on the Sprints, when the team do some certain amount of work. Each finished sprint initiates a next one, over and over until a product is deployed. This way, a product is being developed in a cycles, by iterations. Scrum as a project management framework is not about defining a final shape of a product at the beginning, but rather than that, it is about a product development by adding and building new features, until sum of those features will get a product that precisely fill a customer needs.

Currently scrum is almost a standard in software development. Some companies use it as a tool to organize a product's project, but the other go further and based on this framework they rely an organization of a whole company[Andreas Opelt, Boris Gloger, Wolfgang Pfarl, 2013].

4.3.2 Kanban

Kanban methodology was introduced in the late 1940s when Toyota began optimizing its engineering process. They were inspired by supermarkets which created a model for stock their shelves. Toyota applied the same model to its factory floors in order to align their inventory levels with the actual consumption of materials. They improved communication between teams by creating a "kanban" card, that was passed between teams. When some material was finished at the production line, the team sent the card to a warehouse and at the same time

warehouse sent an order to its suppliers. During the time this methodology evolved to a manufacturing process called "Just In Time" (JIT) [Radigan, 2019].

In a software development world the same JIT principles are applied. It is done in order to align a team's capacity with an amount of work in progress. Kanban improves a flow of a DevOps system. It also visualize a value flow in that system.

4.3.3 Visible Ops

Visible Ops is a methodology that helps to answer the question "where to start?" in process optimization, not only in a DevOps aspect. However values covered in this book are very aligned with DevOps values and standards. It is described in a book "Visible Ops Handbook" by Gene Kim, Kevin Behr and George Spafford. It covers implementation of controls and processes optimization. The authors propose four steps:

1. Stabilize Patient, Modify First Response
2. Catch and Release, Find Fragile Artifacts
3. Establish Repeatable Build Library
4. Enable Continuous Improvement

4.3.4 Incident Command System

It is a methodology to organize the incidents detection and reaction process. Incident Command System (ICS) is a standardized hierarchical structure, used by governments to identify issues and fix them. It can be applied in an operations side of organization. It comprises five main systems [National Service Knowledge Network, 2019]:

- **The Command Section**

It is responsible for developing and maintaining communication between different parties involved into the problem.

- **The Operations Section**

It handles tactical operations, coordinates the operation, and directs tasks and resources to the responsible sites.

- **The Planning Section**

Provides the necessary information to the Command Center to develop the action plan that will accomplish the objectives. They also collect and evaluate information as it is made available.

- **The Logistics Section**

Provides required resources like equipment, tools and personnel to support other sections.

- **The Finance Section**

Is responsible for accounting for economical resources required during operations proceeding.

4.3.5 Agile Software Development

DevOps has its origins in Agile framework. It is based on Agile Manifesto [Kent Beck, 2001] that was created in 2001. Agile Software Development can be seen from different perspectives. In a general sense it is a philosophy, where the value of providing a well working software has the highest priority. DevOps extends those values to include more emphasis to the internal communication within a company.

In the original manifesto published in 2001 [Kent Beck, 2001] the values and the principles can be found.

Values

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

Agile software development comprise leadership concepts, lean management processes, practices, techniques and tools that enable application of those values.

Agile should not be treated only as a set of tools. It is an approach, without which those tools will not represent a high value to the users. Agile as an approach is not a methodology which from the moment of its implementation will not change. This is a mindset that grows and evolves. As it is described in a book of Thomas Stober and Uwe Hansmann:

An agile approach grows and evolves over time by mixing and matching ideas contributed by practitioners.

Thomas Stober and Uwe Hansmann [2010]

4.3.6 Lean Thinking

Lean thinking is a methodology that reduces a waste of effectiveness while the processes and practices are improved [Ferreira, 2015]. By an alignment customers satisfaction with employee satisfaction, lean thinking supports innovation and agile software development. The same applies to DevOps therefore most innovative leaders should consider adding lean thinking to their set of management techniques and methodologies.

Lean thinking methodology in software development was introduced in 2003 by Mary Poppendieck, who gathered all together a set of practices that should be followed in code production companies who wants to apply lean thinking in their processes [Varma, 2015].

- **Optimize the whole** - software cration process is a chain of elements that all together generates a value. Therefore the value of a software is as high as the weakest element of it. That is why it is better to optimize the whole chain rather than its parts separately.
- **Focus on customers** - software as a product is created for customers in order to satisfy their needs. Its development is a complex process and developers easily can lose a goal of satisfying customer needs by focusing on solutions and technical issues. Developers should focus on asking the right questions in order to understand customers needs better. That approach helps to keep focus on solving the right problems.
- **Energize workers** - the most important asset of a company are people. Therefore it is important to keep them excited and energized by giving them a challenging and interesting problems to solve. Otherwise they can quickly lose their interest and energy which is needed to realize projects. Lean is not only about systems optimization. In lean thinking people are considered as the most important element of problem solving process.
- **Eliminate waste** - waste elimination has its application also in a software development.
 - **Partially done work** - every partly done work or temporary solutions generates unnecessary work and results in an unnecessary waste of time. Therefore the parts of code or a documentation that does not represent any value to the whole and final version of a software should be avoided.
 - **Extra processes** - any "paper work" and "bureaucracy" means additional work without generating any value to the customer. Those

extra processes should be eliminated in a company that applies lean thinking methodology.

- **Extra features** - there are many examples of applications, that has a variety of features, but majority of them are used rarely or never. In order to increase an effectiveness of a company, it is a good practice to narrow down a list of features to the most important ones.
- **Task switching** - multitasking may seem a productive approach, but each task switching event generates invisible costs, which summed results in a longer lead time.
- **Waiting** - most of the time spend on a software development is wasted for delays and waiting. Delays and waiting are generated mostly by too many stages of a code production process where each hand-off generate waiting.
- **Motion** - motion in terms of software development means how many interactions must a developer do in order to obtain a given information. Those interactions should be as low as possible in order to achieve the highest performance of a company.
- **Defects** - any defects in a produced software leads to a customer dissatisfaction. The longer a bug stays in a production code, the more costly it is to fix it.
- **Enhance Learning** - software development is a creative process where many knowledge discovered while code writing is being changed or even discarded. A software developer should seek solutions by short experiments and rapid validation of assumptions.
- **Increase flow** - lean software development favors a smooth flow in a code production. In order to achieve that, code production process should be divided on phases that aim to accomplish project in the shortest, easiest and most efficient way.
- **Build quality in** - before a version of code will move to the next stage of its development, it is important to identify the most important defects that can impact on a core of application. That allows to reduce a number of corrections and time waste as well as it will improve a customer satisfaction.
- **Keep getting better** - no one is perfect and the same applies to processes. In the moment when it seems that the perfect process is created and it is implemented it is a time to look at it with a critical eye and look for ways to improve it or look for other, more suitable solutions.

4.4 Most frequently improved indicators

In this project a real life performance of a company that implemented DevOps will be compared with expected results. In order to procedure this action, a specific set of measures must be defined.

In this paper the author will use Key Performance Indicators (KPI) for the purpose of business performance evaluation. To narrow down the number of KPI's that should be checked, the most frequently improved KPI's based on the DevOps implementation are required.

4.4.1 Deployment frequency

A software deployment gathers all necessary activities that make a software available for being used by customers. A deployment frequency means the number of executed deployments in a specific amount of time. It is one of the most important KPI that is being improved under DevOps influence. It directly affects the company's business results, if a company is rewarded for each application that is deployed. It is also a key competitive advantage to be able to deliver new features to customers as fast as possible.

4.4.2 Speed of deployment

A high deployment frequency depends in a major part on a speed of deployment. It is a time from the moment when a developer writes first lines of code to the time it is fully deployed. According to statistics [Earnshaw, 2013] more than 25 percent of the teams were able to accelerate time of deployment to less than a day. PayPal publicly said that it has improved their lead time to one hour or less [Brown, 2013].

4.4.3 Deployment failure rate

An increased deployment frequency may result in a lower quality of code. It is not a desired state. DevOps practices make a significant impact on a failure rate. Amazon Web Services claim to have only 0,001 percent of their deployments cause outages.

4.4.4 Mean time to detect (MTTD)

Mean time to detect refers to the average amount that takes the issue detection. It is usually measured from the beginning of the system outage or other malfunction to the moment when the team identify that issue.

It is a perfect metric for teams that need to track the effectiveness of their incident management process. It reflects the effectiveness of the operation abilities in the team [AlertOps, 2018].

4.4.5 Mean time to repair (MTTR)

Mean time to repair refers to the time it takes to resolve the problem of an outage or malfunction. It is a significant metric for DevOps structure, that measure a team's reaction time and the effectiveness of development and operations cooperation.

4.4.6 Customer satisfaction

All above KPI's has their influence on the customer satisfaction. It is probably the hardest metric to measure, and it can rely only on the customer's surveys.

4.5 Importance of DevOps

Modern companies understands an importance of improving their processes. According to the 12th Annual State of Agile Report [CollabNet, VersionOne, 2018] 71% of respondents stated that they are currently running DevOps initiative or are planning it in the next 12 months.

Respondents who already have DevOps implemented were also asked about how success is measured in their organizations. The results were as follows in the table 4.1.

Top success metrics	
Accelerated delivery speed	58%
Improved quality	51%
Increased flow of business value to users	44%
Reduce risk	41%
Increase customer satisfaction	33%
Decreased IT costs	28%
Ensure compliance/governance	17%

Table 4.1: Top measures of success for the respondents [CollabNet, VersionOne, 2018]

**Respondents were able to make multiple selections*

Those results confirm that the aspect of the code production process that is being improved mostly is a speed of delivery.

Respondents were also asked about the value of capability improvement in practise.

The most rated answer was:

Ability to measure cycle time, wait time, bottlenecks of business value flowing through delivery cycle.

ILS as a company 5

ILS Sp.z.o.o. is a Polish software production company that supplies Pelion group in a software products. ILS stands for: Intelligent Logistic Solutions.

5.1 ILS as part of Pelion S.A.

Pelion S.A.

Pelion S.A. is the biggest pharmacy group in Poland. It was started by Jacek Szwajcowski in 1990. Overall Pelion comprises over 160 co-partnerships. The goal of Pelion is to supply pharmaceuticals into all possible fields of its distribution. At the same time Pelion is also the most modern wholesaler of pharmaceuticals in Poland.

It is a parent company for three main activities and two supportive ones. Pelion supports polish health care in three ways:

- **Wholesale**

This is achieved by two companies: PGF and Pharmapoint. These companies distributes wholesales pharmaceuticals in amounts of around one billion units a day and cooperate with over 600 clients.

- **Retail**

The biggest retail brand is DOZ S.A. with other smaller brands. Thanks to the retail business line Pelion can reach direct customers as well.

- **Hospital supply and logistics**

Conducted by Urtica and Pharmalink. A hospital support run by Urtica is another opening to the new fields of drug distribution. This time new clients are hospitals. Logistic services is historically the youngest business activity of Pelion. Thanks to its expertise in a logistic processes related to the pharmacy products, as well as thanks to the knowledge of the law and regulations of these processes, Pelion can offer its clients services reliant on save and legitimate way of pharmaceuticals transportation.



Figure 5.1: Pelion's activities

Pelion itself comprises also two supportive companies: BSS which takes care of the whole HR processes in all Pelion companies and ILS - Intelligent Logistic Solutions which supports Pelion as a group with its software products.

5.2 Intelligent Logistic Solutions S.A.

Current CEO of ILS is Rovert Pławiak, who together with the vice president Paweł Cieżki and business lines directors builds a management team of the company. ILS currently goes through many structural changes, and one of its steps is implementation of a new organizational structure DevOps in its processes. ILS was originated in a company supporting Pelion's activities in terms of IT solutions. It is important to clarify, that ILS is not a reflection of IT department for the whole group anymore. It is as a software production company that has three main business lines in pharmacy industry: wholesale, retail and hospital support and logistics. ILS specialize in processes automation as well as seeking and building a software solutions for those business lines ILS works based on a close collaboration with IT departments of its clients. Thanks to such collaboration it is enabled to build an innovative agile environment and business processes optimization.

5.3 Mission of ILS

ILS has its mission statement:

LET'S TALK ABOUT IT – LET'S DO IT – IT WORKS

They achieve it by a professional IT care, providing a congeneric services, starting from a needs and requirements analysis to an operations support of freshly integrated or already existing IT systems.

5.4 Values of ILS

ILS has a set of values that determines a successful mission fulfillment.

- Reliability - they work in an accountable and ethical way, by fulfilling the statements of contracts and respectful cooperation with clients
- Teamwork - ILS works based on a trust that employees can rely on each other
- Effectiveness - by a rational resources management ILS provide the highest standard of services
- Security - ILS provides services with an appropriate standards of security, dependent on the requirements of the project
- Innovativeness - ILS seeks for an innovative solutions to provide the best services to their clients

5.5 ILS field of expertise

At the very beginning ILS was maintaining a big authorial ERP system called HAL. It was based on an Oracle Database and other Oracle products. This system was build to manage all activities of Pelion from its early stages of development. HAL is responsible for maintaining all processes including finances, logistics, HR, and stack positions.

In the early stages, Pelion did not have a clear distinction for the business lines. That resulted with no clear distinction in the HAL system as well.

While company was growing, the distinction was required due to the different business needs. That distinction appeared also in the HAL system. HAL system was build using a various Oracle products from database to the user interface.

With the growth of the company each business line had different needs regarding their products. Because of that, new applications were build using different programming languages like Java, Kotlin or JavaScript and different frameworks were implemented like SpringBoot, Hibernate or Angular.

Now ILS has a strong expertise in developing modern applications and is a provider

of database and data center services including IT project management and IT support.

5.6 ILS structure

With a growing needs and requirements from a client's side, a new structure of ILS was necessary, in order to satisfy clients. Each business line had different technological requirements and they work in a different rhythm, which is problematic and expensive to keep all the departments up ready to act in case of problem occurrence.

Business lines use various project management processes. In order to improve their services, ILS need to adapt to those processes, which helps to smooth the collaboration between parties. For example the retail business line uses sprints to manage projects, which suits their needs. On the other hand the wholesale business line finds a waterfall system more suitable to their requirements.

A waterfall project structure is based on on the project flow and contains five steps:

1. Requirements - step when all product requirements are designed
2. Design - project of a software architecture
3. Implementation - it include software production
4. Verification - phase for testing
5. Maintenance - it is a state when a software product is placed in the production environment

Each step must be finished before entering to the next one.

It was not possible to fill all the requirements without team reconfiguration, and a new structure of ILS was necessary. Since November 2019, ILS team is subdivided to three main business lines, that take care of a software development for their clients:

- **Retail Business Line**

It is responsible for supporting and supplying with a required software tools all retail companies that are part of Pelion. Big amount of activities of that team is focused on an automation of logistic process of drug supply to pharmaces.

- **Wholesale Business Line**

It supplies a PGF which is a wholesaler of Pelion with software solutions to create a modern and automated environment of PGF's employees.

- **Hospital Business Line**

Urtica's main competence is to supply hospitals with required products. This business line in ILS makes software solutions that enable easy collaboration between Urtica and their customers.

Each team in a business line comprises five positions (which will be described in the Chapter 6):

1. Leader
2. ERP Architect
3. Developers
4. Business Analysts
5. Testers
6. Ops

Beside that there are other departments. These departments support a software development team. Their existence is necessary for achieving business goals of a company. They support project management and resources organization and optimization. Thanks to their support it is possible to keep a required level of security as well as to keep coherence of ERP system architecture. These departments are:

1. LBF - rozwinięcie?
2. IT Projects
3. Strategy and Finances
4. IT Security
5. IT Architecture
6. HR
7. Compliance

DevOps Implementation -

ILS case 6

DevOps as a term is very broad and each person can interpret it a bit differently. DevOps as a structure is not a universal solution, therefore it is required to perform a deep analysis of the company's situation in order to implement it in a right way. In this chapter there will be discussed based on an ILS case.

6.1 Implementation process

Pains identification

For a long time before first discussions about DevOps, the management team suffered from many problems with finding a universal suitable solutions for all clients. Many problems were addressed to the poor knowledge sharing regarding different products. There were some employees who had some specific knowledge about certain products and that knowledge was not documented and not standardized. Other problems were connected to a different work rhythm of each client. Different working hours and different project management approaches of clients caused a lot of unnecessary problems with communication and work management on the ILS side.

This resulted in a long discussion about structural change. A more flexible but also product or business line oriented structure was needed. DevOps was a perfect choice for resolving those problems.

Initial analysis

Management team did an analysis of technological competences of all employees. They also identified who has a strategic knowledge about each product in order to perform a proper employees division. Based on those analysis they restructured the company and implemented "the teams per product" philosophy instead of "team per responsibility".

Change Management

ILS has a specific procedure that defines the rules regarding the change management. It describes how the processes and practices change should be performed. Such approach enable to standardize a way through such complex process. Is is implemented in purpose of rules definition regarding sourcing and IT systems modifications, testing, integration and new or modified product reception. This procedure applies not only to a change inside ILS. Each client who require a system change in their processes, and report a need for such change to ILS also goes through that procedure.

Change management includes organizational changes. Therefore this procedure was activated while devops implementation. In this procedure there are specified steps that must be executed while implementing a new structure, that defines the way of preparation to change, its implementation as well as the testing processes.

These steps are as following:

1. Report the need for a change using a special form for that purpose
2. Processing the reported changes
3. Agreement among the required changes
4. Project execution
5. Acceptance tests
6. Ready change handing over
7. Changes registration

There are certain gains that are a motivation for that procedure. First of all, it allows for a processes optimization and minimization of a failure occurrence. It also gives clients an opportunity for a better understanding of its own processes. By such procedure, ILS gains an image of a company based on standards and processes among the group of Pelion. It also enables ILS to optimize an usage of its own resources. Last but not least, personnel of ILS gains a contentment from a clients collaboration rules stabilization.

In order to assess the success of change management procedure implementation, there was made a survey among the employees. Purpose of that survey is to diagnose if personnel knows its rules, if they follow them and if the procedure is aligned with current business requirements and situation of ILS.

Transition to a new structure

Currently ILS is still in the process of transition to a new structure. New projects are done in a new structure already, however projects from the time before the change, cause some inconveniences. Even though, the team performed a competence and strategic knowledge analysis, some employees still has a strategic knowledge about projects from the other business line (which should be gained and transferred to the other team). This is a process which require some time to be adapted.

6.2 DevOps team structure

An implementation of DevOps included a team restructure. In the early stages ILS comprised different departments and the responsibility distribution was divided per department. That meant for example that a department of testers was only responsible for testing, and a department of developers was only responsible for development of a new product. There was no distinction which customer is supported by which employees. The communication between departments was generating many unnecessary problems, mainly based on communication.

Currently the communication flow is improved a lot by implementing DevOps. As mentioned in the previous section, now teams are build in the idea of team per product. It means that each team comprise all required specialists to develop and maintain product during its life-cycle.

New team consists of the following positions:

Leader

A leader is responsible for managing the teams tasks and sustaining the progress of the team. It is an important role because leader is responsible for the most important decisions.

ERP architect

ERP (Enterprise Resource Planning) is a software based system that enable a resource planning in the entire business process. It allows for using an integrated applications for management and automation of most of repeatable office tasks.

ERP architect is a team member who is responsible for an integration of new applications with the HAL system and provide a compatibility with the entire

system.

Developers

Developers use their competences to write a code that is compliant with requirements. They also takes care of finding the best possible solutions in technological aspect of the problem.

Business Analysts

Business Analysts keep an eye on the actions of the application in the context of the whole system and its impact on the business activities of the company. Business logic in the programming terminology is a reflection of the real world business dependencies.

Testers

Testers takes care of a validation part of code production process. No matter what kind of change developers did in their code, testers have an eye on it and run tests to find possible malfunctions of new applications. It is a significant part of an application development.

Ops

Ops, are team members who solve problems after the time when application runs on the production environment. It is essential that Ops are in a direct relation with developers, because it improves the time of finding a solution to the problem.

6.3 Expected improvements

In a more general sense, DevOps is being implemented for two purposes: communication flow, and work speed improvement. However, every company that implements DevOps, may expect slightly different results. It depends on the actual situation of a company, as well as the type of it's products, it's maturity and overall business processes inside a company.

ILS it's expectations put in a better project management. First of all ILS wants to isolate projects within the business line. That can improve few aspects. Beside having a better flexibility regarding technological requirements, employees have also

better chance to focus on one client and to understand a characteristics of a certain client.

Second of all, there is a problem that some employees has a very specific knowledge about certain products. In case of problems, the whole team rely on one person, which is an inconvenient situation, because in case when that person is absent for any reason, it takes a lot of time for other employees to identify a problem and to find a solution. It is more problem of knowledge sharing but it indirectly leads to DevOps and product oriented team structure.

Having a product oriented team structure also helps to manage projects more efficiently. It is easier to manage resources when there is a clear distinction which team is responsible for which product and project. Therefore it is expected that the new structure will improve a project management process.

ILS also expect to improve its mean time to repair. When there was a responsibility oriented structure, it took more time for operations department to identify where the problem lies and where to find source of information that can help to find a solution. Having a closer relation between operations employees and developers has an impact on the reparation time after the failure recognition. When customer reports a problem, an employee already knows which team they should contact with, and developers from that team can identify a problem straight away as they have the best knowledge about the code they have wrote.

Finally ILS expects that the new organizational structure will help to adjust the work organization process to clients requirements. Each client has different needs regarding time and project management or a technology. It became almost impossible in the previous structure to fit all the needs with one solution. It generated a lot of unnecessary problems when ILS was trying to satisfy all those needs and it was very hard to manage such messy solution. A new structure provides a high flexibility regarding the product requirements without causing many inconvenient problems.

Beside the project management and communication improvements ILS expects also to see an indirect reflection of change in a business results of the company. Based on this master thesis project, the management team will implement a tool for KPI tracking, that will enable them to get more awareness about the real effects of a change.

6.4 Key Performance Indicators (KPI)

6.4.1 Initial intention of the project

An initial intention of this master thesis project was to track KPI's during the semester, and based on a results, analyze an impact of DevOps on the business results of the company. However due to a fact that company is still in a process of change, and maturity of the DevOps structure in ILS is still in progress it was problematic to implement KPI's tracking and results could be misleading. Therefore the author decided not to track those KPI's from the beginning and spend more time on investigation of internal impact of a new structure on the employees satisfaction and their impressions. Those impressions and opinions are gathered in the surveys in the Chapter 7.

An author's KPI initiative was also well received, therefore the management team of ILS decided to use it in the company's business evaluation processes. Those KPI's with its definitions are described in the next sections.

6.4.2 KPI introduction

Key Performance Indicators (KPI) are a measurable values of a crucial aspects of an organizational performance, based on a parameters related to those processes. Together they are a tool that measures the performance of a company. It is used in order to identify how good the processes, products, activities and practices are. It is a natural choice of a tool that can measure the successful DevOps implementation. It also enable to analyze the success of a company, based on a real data. It helps to eliminate decision making process based on a rule of thumb.

Key Performance Indicators are created in order to keep business strategy aligned with a strategy. Therefore they should be tracked with a regularity in a specific time spans dependent on the company's tempo and abilities. A popular myth is that KPI should express how much money a company earns on which project. It is a wrong way, as using a financial values makes them a Result Indicators (RI). RIs are a result of a company's performance but they cannot indicate why a performance has a certain level. Therefore KPIs are nonfinancial values [Parmenter, 2015]. KPIs should be team based. A responsibility for a values of KPIs should be a applied to a minimum scale of a team. It is deep enough to indicate where the problem is.

Some of the KPI's are universal for all companies, but some are unique for an

industry [Reh, 2018]. That matter is not different in terms of software production industry. As mentioned in chapter 4, the following KPI's are most likely to be improved by using DevOps [Watson, 2017]:

- Deployment frequency
- Change volume
- Deployment time
- Lead time
- Automated test pass %
- Defect escape rate
- Availability
- Service level agreements
- Failed deployments
- Error rates
- Application usage and traffic
- Application performance
- Mean time to detection (MTTD)
- Mean time to recovery (MTTR)

However, KPI's are dependent in a major part on the characteristics of the company. Each company has different business processes and different relation between it's units and products, and what is even more important, they place their business value in different products or services. Therefore it require an analysis of the company's needs and processes to build a custom set of KPI's.

After discussions, the author and the management team of ILS designed a set of eight Key Performance Indicators that should be implemented and tracked in order to investigate the company's optimization especially in terms of current DevOps implementation.

6.4.3 ILS custom KPI set

ILS produce a wide range of software applications in terms of technology and its application to business. Clients of ILS order applications that are compatible with their already existing systems and require their maintenance at any time. Beside software development ILS provide a data center services, database management as well as the network infrastructure. ILS base its work organization on a ticket system,

irregardless of the business line. This ticket system is now implemented in a Jira system, a powerful product of Atlassian. That approach simplify a KPI tracking process and the Jira software enables to track those indicators in a reliable way.

In general KPIs designed to be tracked by ILS can be divided into three categories(Fig. 6.2):

Efficiency

This group of indicators will provide an information about how efficient the company is and how fast can teams deploy their code in relation to the team size. Based on such information the management team will be able to identify if the problem with efficiency will occur. They will be able to identify if the problem of delays lies in the inefficient processes, too big workload or any other factors.

Efficiency group comprises:

- Time To Market (TTM)
- Number of deployments

Quality

Quality indicators enables to measure the quality of the products produced by ILS. It will provide information how many malfunctions products has and what may be even more important how fast it takes to ILS to implement a solution for them. Not always lower quantity of malfunctions means higher reliability.

For example if one company has no issues with their product and after a three years a problem occurred which took 24 hours to solve, users will probably complain about it, and definitely will notice it. If the same company had a product that has three problems a day, but a company is able to solve them in approximately five minutes, then an overall customer experience is significantly higher.

What is more, it is also economically efficient to solve a high number of small mistakes as fast as possible rather than fixing a low number of a big failures late after the implementation. The cost of bug fixing rises exponentially over the time [Ambler, 2004].

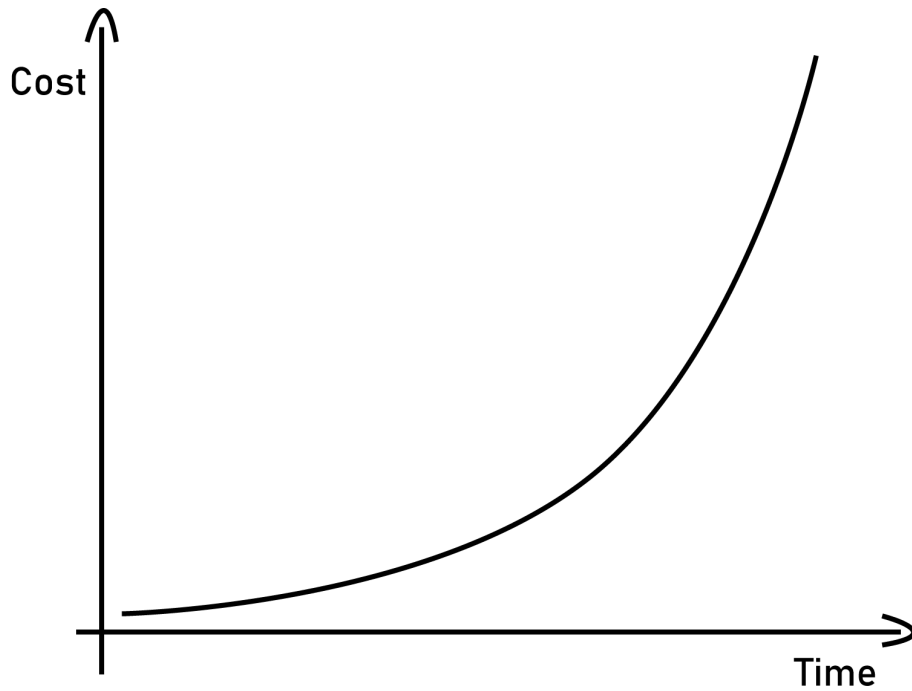


Figure 6.1: Bug fixing cost rises exponentially over time

Indicators in the quality group:

- Mean Time To Repair (MTTR)
- Number of failures
- Customer satisfaction

Capacity

A capacity group of indicators can help to identify internal needs and abilities of a company. It gives an information about the workload and about the level of the use of the resources. That kind of information is essential while negotiations with clients, and may be a decisive point in discussions.

An information about capacity of the company is also important for a future planing. If there is a high demand for the company's services but its capacity is to low it may be an opportunity that should be used for recruitment decisions.

Capacity groups three indicators:

- Number of opened projects
- Production time
- Estimated Time Deviation

Key Performance Indicators groups

Indicators group	Efficiency	Quality	Capacity
Value provided	Ability to identify the occurrence of the problems with an efficiency	Ability to provide the highest customer experience by a processes optimization	Supportive information for decisions regarding the future projects planning about existing resources
Key Performance Indicators	1) Time To Market (TTM) 2) Deployments quantity	3) Mean Time To Repair (MTTR) 4) Malfunctions quantity 5) Customer satisfaction	6) Opened projects quantity 7) Realization time 8) Estimated time deviation

Figure 6.2: KPI groups

6.5 KPIs description

Creation of a specific set of KPIs is an iterative process. The proposed KPIs are agreed with ILS management team to be implemented and tracked as a first set. Each quarter the set of KPI will be reviewed by the management team in order to evaluate its relevance to the current situation of a company.

6.5.1 Time To Market (TTM)

First of the indicators in the group of efficiency is a Time To Market (TTM) indicator. It is a time that it takes for a company to deploy a product in a client's production environment. A short TTM means a fast return of investment. Therefore usually companies want to minimize it as much as possible. However if a short TTM is a result of skipping some steps of a product development, like for example testing, then it may result in a low quality of a product. However not always it happens. Especially in agile software development a shorter TTM may be just a result of a better planning process. Each scrum cycle include all required steps before deployment, so a shorter TTM depends strongly on a lower number of cycles.

In the ILS it will measure how long it takes from opening a ticket to the moment it is resolved. That time will not include the time the ticket was waiting in the queue. This decision was made because of the fact that queue in ILS is strongly dependent on the clients side decisions. Before there is a decision to start a project, many analysis are done for preparation in order to identify if there are no issues for a project implementation and integration and very often a decision about starting a

project is not done for a long period of time. Another reason is the fact that there is a high demand for ILS services in Pelion group and currently ILS is building its personnel to increase its business capacity.

6.5.2 Number of Deployments

A number of deployments reflects a total amount of applications deployed to the clients servers in relation to the time. Such information will enable to measure a frequency of deployments. An actual number of deployments has a direct influence on the business results of the company. Naturally every company wants to maximize it as possible but it may result in a lower quality of a code or in overestimation of the available business capacity.

In the ILS this indicator will be measured by calculation of a number of solved tickets. Tickets is a basic unit while ordering a product in ILS negotiations with clients. Tickets may have different purpose. It can be a need for a completely new application, an additional feature in existing software or a bug fixing. It will be measured in a time span of month. By taking into consideration a speed of work in ILS a month is a proper period of time, when that number can be measured.

Additionally the Number of Deployments will be measured in relation to the workforce of ILS. It will be set as a separate indicator, however it will be treated as a "sub indicator" that provides additional information. This will give the management team an information about how efficiently the capacity of the company is exploited. Based on that the company can make more conscious decision, regarding whether or not the new tickets should be accepted and about the possible increase of the number of employees. However the company should not draw conclusions based this indicator too rapidly. Each decision should be preceded by a deeper analysis of a case. Calculation of the number of deployments per time depending on the number of employees goes with inaccuracy regarding the causes of it. To low value of this indicator may be a result of a wrongly managed processes and projects. The question that should be asked in that case is if there is an unexploited workforce of the company. A negative answer for that question may suggest that the problem lies in the processes of the company, which may not be enough efficient. On the other hand too high value should also be treated as a warning sign for the management team. If that happen, it may be an initiator to pay more attention to the quality of the code. If the quality decrease that may be an information about overestimation of the available business capacity.

The Number of Deployments indicator in relation to the workforce will be calculated in two versions: considering a workforce available in the measured period of time and considering a total workforce of the company. Available here means the employees that did not take a holiday in that period. Too big differences between those numbers may be a reason to investigate where the problem lies.

These indicators will be calculated by applying the following equations:

$$ND_{available} = \frac{ND}{E_{available}}$$

and

$$ND_{total} = \frac{ND}{E_{total}}$$

Where:

$ND_{available}$ – Number of deployments in relation to a number of available employees

ND_{total} – Number of deployments in relation to a total number of employees

ND – Number of deployments in a specified period

$E_{available}$ – Number of available employees

E_{total} – Total number of employees

6.5.3 Mean Time To Repair (MTTR)

Mean Time To Repair is a popular indicator to measure the success of devops implementation. Because a devops teams structure shorten the distance between Ops and Devs, most likely a solution finding process should be improved and a reparation time should be decreased.

This indicator is used to evaluate how good a failure reparation process is. A failure reparation process comprises it's diagnosis, reparation, testing and deployment of a solution. The time for this process in ILS is measure from the moment when the ticket of a problem type is reported or the moment when problem is identified by ILS to the moment when the ticked is closed and marked as solved.

It is important that the MTTR is short, it is even more important for a company to have it short, than to have a low rate of failures. It happens because of the fact that if a company had only one issue after three years, but its time to repair was a day, then definitely users would notice and most probably would complain

about it. If the same company had three issues a day, but their reparation time took only three minutes, then just a little number of users would notice it and the customer experience would be much higher. For that reason this indicator has a high importance. The management team should track it, and in case of its deviation especially in direction of increase, make a diagnosis of that event. This indicator is considered in the specific amount of time. In case of ILS it will be a time span of a week. Two values are used to calculate it: reparation time of each failure and a number of failures in a given time. Calculation of MTTR will be based on the following formula:

$$MTTR = \frac{T_r}{F_{week}}$$

Where:

$MTTR$ – Mean Time To Repair

T_r – Reparation time of each failure

F_{week} – Number of failures in a considered week

6.5.4 Number of failures

Number of failures is an another indicator from the group of quality indicators. It informs about the number of failures in a specified period. It influence the quality of a code directly therefore it is a natural desire for a company to keep it low. It may have a risky consequences. A low number of failures may result in a higher MTTR which is not a good case, as described in a section about MTTR. Number of failures will be lower if the employees who resolves the problem pay more attention to the case. Unfortunately a higher attention to details very often requires more time, to spend on one problem which results in a higher MTTR.

Because MTTR should be kept as low as possible, therefore the number of failures should be tracked always together with MTTR. The relation between those two indicators should be kept in a balance. Both should be kept as low as possible, and the company has to identify which indicator has a higher importance to their business. Only then a company can give the best customer experience.

In the ILS case the number of failures will be considered in a week period. This indicator will be calculated from the time of its implementation. Therefore it will apply to all products. It will include projects from the time before DevOps structure. By making a distinction between old projects and current ones it should be possible to see the influence of a new structure to failure rate.

6.5.5 Customer satisfaction

Customer satisfaction describes how products and services of ILS meet customers expectations. It is not a hard data metric, that can be measured using calculations. Even though it is not easy to measure, it is a factor that influence directly the demand for the ILS products and services. It determine the quality of products and therefore it belong to that group of indicators. Because of those reasons there is a high importance to measure this factor.

Devops structure directly influence the customer service thanks to a faster feedback and response during the software development as well as its support time. In devops structure products are developed more rapidly. That responsiveness and dynamics of a product development is a strong advantage of devops.

In ILS customer satisfaction will be measured using specially designed surveys. Questions will help to measure how customers experience ILS's service and may point out weaknesses that the personnel was not aware of. Due to the Pelions policy, ILS has a constant set of clients. Therefore that surveys will be conducted every six months. More frequent surveys conduction is not necessary, and may appear disruptive.

6.5.6 Number of opened projects

Its a first indicator from the group of capacity indicators. It gives an information about how busy employees are and what abilities a company has, in terms of readiness for taking the next project. Together with other indicators this information may also indicate that the personnel is overworked. A very high value of number of opened projects should be a warning sign for the management team.

Still it is natural to think that this number should be high, because that number influence the earnings of a company. In fact the studies shows [L. Dooley, G. Lupton, D. O'Sullivan, 2004] that it is not a good practice. Too many opened projects most likely result in ineffective work of teams. Nevertheless that number is needed by the management team to estimate the current business capacity of the company and keep the workload in balance. With an awareness of the current business capacity the management team is able to make decisions about the future goals of the company.

New projects in ILS are divided depending on its size. They are grouped into size levels:

- S - small projects
- M - medium projects
- L - large projects
- XL - extra large projects

A size of the project may change during its development. That change may be done for two reasons:

1. A project size was under (or over) estimated at its beginning by client or ILS
2. A client did not realized its own need at the beginning

ILS personnel is constantly improving the communication flow with clients in order to avoid too often changes in the sizes of the projects.

A number of opened projects is going to be summed up weekly for each size of a project independently.

6.5.7 Estimated time deviation (ETD)

Estimated Time Deviation is the last indicator in a group of capacity. It is not a direct indicator of capacity, however it impacts it in an important way. If this indicator is high, it may be an important information about ability of a company to estimate its own capacity.

Estimated Time Deviation indicates how much the time spent on a project execution was longer than expected time of its execution. With that information ILS can infer, how precisely they can estimate time needed to finish a project. It is also a validation on how precisely ILS can estimate its own business capacity, and estimate a project size. A project size among other factors depends on a time required to accomplish a project. If a project size was under or overestimated then reflection of that mistake will be visible in estimated time deviation.

There will be two versions of that KPI. As a first step Time Deviation will be calculated by subtraction of an estimated time of execution from a real time that was spend on that project. A negative value of this indicator will mean that the estimated time was overestimated. A positive value will mean about its underestimation. A second version will be a mean value of ETD in a given period. In this case it will be a month period. It will be a value of a total ETD in a current month divided by the number of closed projects in that time. Both versions will be calculated for a closed projects.

ETD will be calculated using the following formulas:

$$ETD = T_r - T_e$$

$$METD = \frac{ETD}{P}$$

Where

ETD – Estimated Time Deviation

METD – Mean Estimated Time Deviation

T_r – Real time of execution

T_e – Estimated time of execution

P – Number of closed projects in a given month

Estimated Time Deviation has direct impact on a customers satisfaction. The better company redeem the initial decisions, the better impressions clients has regarding its reliability.

6.6 Technical implementation execution

All indicators will be calculated based on data that is gathered in the Jira system. This system enable a customized dashboards display and organizing data presentation in a flexible way. Thanks to that the management team will be able to access the data in any moment.

Every indicator will be calculated for each business line separately as well as a total value. Such distinction enables a more precised place and cause of the problems determination. That approach allow for a better identification of a needed improvements. In case of a problem recognition, such case will require a deeper analysis of its circumstances.

6.7 Map of KPIs relations

In the description of all indicators there can be seen a strong relationship between each indicator. Usually indicators are dependent to each other, therefore it is important to identify such dependencies in order to be aware of them. Knowledge of those relations enable identification of causes of undesirable behaviour [Adela

del-Río-Ortega and Manuel Resinas, 2009]. KPI relations may be divided into proportional or inversely proportional. Most often there is one direction the influence of indicators. Such map of these relations and its directions enables to identify a hierarchy of them.

In order to ease the discussion of these relations, the author uses abbreviations instead of full KPI names:

- **TTM** - Time To Market
- **NoD** - Number of Deployments
- **MTTR** - Mean Time To Repair
- **NoF** - Number of Failures
- **CS** - Customer Satisfaction
- **NoOP** - Number of Opened Projects
- **RT** - Realization Time
- **ETD** - Estimated Deviation Time

Before a relation mapping author made a tabular way of data presentation in order to prepare data to map them graphically.

Num.	First KPI	Second KPI	Influence type
1	TTM	NoOP	inversely proportional
2	TTM	NoD	inversely proportional
3	TTM	ETD	proportional
4	TTM	NoF	inversely proportional
5	NoD	NoF	proportional
6	MTTR	NoF	inversely proportional
7	ETD	CS	inversely proportional
8	MTTR	CS	inversely proportional
9	CS	NoOP	proportional
10	NoF	CS	inversely proportional

Table 6.1: KPIs relations and dependencies

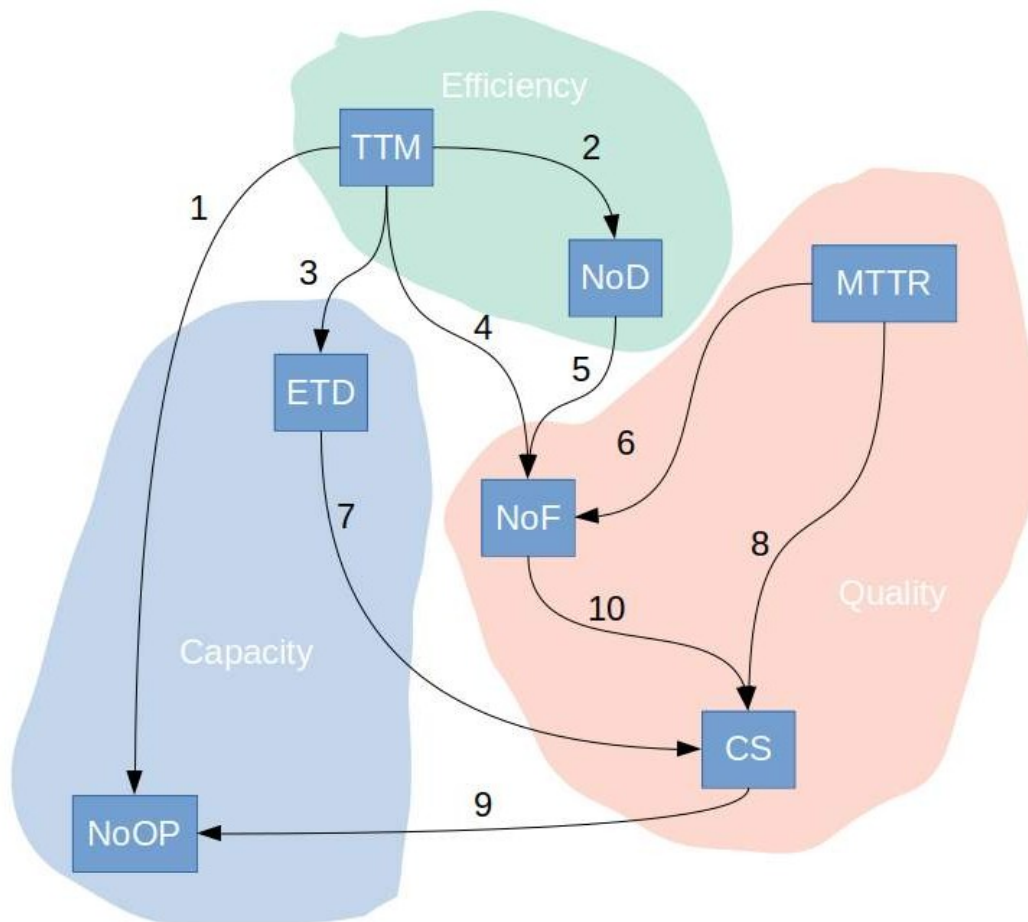


Figure 6.3: Map of KPI relations

A significant conclusion from a map of dependencies is a fact that there are two main factors that initiate an influence flow. It is a Time To Market and Mean Time To Repair. The indicators in the end of an influence flow are Number of Opened Projects and Customer Satisfaction. ILS benefits from having them at high value, because it transfers to the business results of the company. Worth to mention is a fact that a low TTM and MTTR results in a higher Customer Satisfaction as well as a higher Number of Opened Projects. This is an important dependency for the case of DevOps. One of the main purposes of DevOps implementation is improvement of a speed of work. Thanks to the map of relations it is easier to understand why DevOps in the end results in a better customer experience.

It is important to notice that the relations between these indicators described in such a simple way does not truly reflect the reality. These relations are influenced by many factors, very often connected with the current situation of a company. It is a complex problem and this map represents a simplified picture of the whole.

For instance relation between MTTR and NoF not always has to be inversely proportional. An assumption that is inversely proportional arise from the fact that by lowering a time for a failure reparation, a risk of imprecise solution of a problem may be increased. Therefore increasing a Number of Failures. However it may happen that the higher code quality can result in a lower NoF. A better code most probably will include less crucial or complex failures, and therefore the time needed for its repair can also be lower.

Empirical Data Collection

7

Initial intention of this master thesis project was to gather a KPI data from the period of a semester and interpret them afterwards in terms of a DevOps influence on a company's performance. Unfortunately ILS is at to early stage of DevOps implementation to be able to assess its influence on business results of a company. However in order to assess its influence on a company performance, the author made a survey among the employees, which purpose was to inspect what are its effects from employees point of view.

Two separate surveys were done: one for the management team and other, more general for all employees. Those surveys were designed by the author and executed using a MS Forms application. The management team filled both of them. In the survey dedicated to them there are additional questions more specifically asking about the performance differences in the scope of a team. Both surveys were made in English language with Polish translations, in respect of the employees who had less English language experience. Responses were written in English and Polish dependently on the level of English of the respondents. In the Chapter 8 the results of both surveys are discussed. These results are combination of translations of Polish responses and original English ones. Original results are placed in the appendixes.

7.1 Survey for all employess

This survey contains seven questions. Its purpose is to identify what picture of DevOps structure employees has, as well as if they feel its impact on their work.

1. What is your age?

Purpose of this question is to define an age of respondents.

2. What is your role in the company?

Depending on the position in the structure the feel of its impact may differ. It may be an important information while planning the communication of a structural changes and it can help in understanding an unexpected behavior of some employees.

3. Are you aware of DevOps structure in your organisation?

This question is asked in order to evaluate the communication of a structural changes of a company. At the stage of DevOps integration, many processes are not fully correctly executed. Thanks to that question it is possible to separate an option related to the lack of a new structure awareness among the employees.

4. How do you rate a DevOps influence on the communication in the team?

One of the most often improved elements of a team collaboration in a DevOps structure is a communication. Thanks to this question it is enabled to analyze an impact of DevOps on a team's communication.

5. How do you rate a DevOps influence on the speed of work?

Another most frequently improved factor is a speed of work. An increased speed of work is usually possible thanks to the better teams communication and integration.

6. Do you feel that DevOps improves your work? How?

This is an open question, where the author asks employees about their other impressions about DevOps influence on their work. With this question the author try's to identify other possible ways of DevOps influence on their work.

7. Explain briefly in your own words, what is DevOps about (how does DevOps works)

This open question has its purpose in identifying how does the employees understand DevOps. In order to understand the results of the surveys better, it is valuable to get better picture of how they understand the structure they are working in. It may happen that the respondents has different picture of it than the management team. It such situation happen it may be necessary to improve the communication of its

processes and values. Responses to this question may help in better interpretation of the other responses.

7.2 Survey for the management team

Questions directed to the management team has its purpose on identification of a new organizational structure influence in the scope of a team. These questions are directed mainly to the directors of business lines. Thanks to this survey a communication with clients is also being assessed.

1. How do you rate an impact of DevOps on the business results of your team?

With that question its enabled to get an information about a real reflection of DevOps in a business results of each business line. Each business line director has a direct access to that data and can empirically evaluate influence of a new structure on them.

2. How do you rate a DevOps influence on the communication in your team?

This question is equivalent to a question about the communication from the survey to all employees but its scope includes a team managed by the respondents. Answers for this question in confrontation with answers of the given team members can improve interpretation of a survey results.

3. How do you rate a DevOps influence on the work speed of your team?

This question also enables to indicate a DevOps influence on the most important performance element that is being improved while DevOps implementation. Directors of business lines has also an ability to answer to this question based on a real results of a given team. In this question a speed of work applies to the whole team. With that meaning, speed of work is a sum of a work tempo of all employees that belongs to that team.

4. How do you rate a DevOps influence on the communication with your clients?

With this question it is possible to identify if one of the most important factors impacting the code production process is being improved, which is a communication with clients. An effective communication with client is important not only because of the marketing reasons. It has also an impact on an efficient work of teams. Therefore it is necessary that this communication flows in an undisturbed way.

7.3 Expected results

It is expected that answers related to the improvement of a communication and a speed of work will be rather positive. Responses to the question number seven in the survey directed to all employees where the author asks for a brief description of DevOps with respondents own words may bring interesting results.

An exemplary response should be similar to that sentence:

DevOps is about integration of a number of people with all capabilities required to develop and deploy a product.

A general impression of the author after conversations with employees is that, they had very diverse feeling about what DevOps is and what is it about.

This is a natural situation, because there is no one correct answer what DevOps is. This term can be interpreted differently. However it is important that in the scope of the same company understanding of that term was more aligned. Otherwise it may be hard to execute processes in a new structure implemented.

Empirical Results 8

In general the results of surveys are aligned to the expected. This survey was designed and distributed using MS Forms and its original results are placed in the appendixes.

8.1 Survey to all employees

The attendance of the respondents in the survey directed to the all employees was 22%.

8.1.1 *Whats your age?*

Majority of the respondents was age of 35 - 50 years old 8.1. An average age of software developers in Poland according to *Stack Overflow Developer Results* is 26 [Stack Overflow, 2016]

This means that a majority of ILS employees is older than an average in the country of Poland. This result is related to the fact that ILS still has to support technologies, which specialist are in that age range.

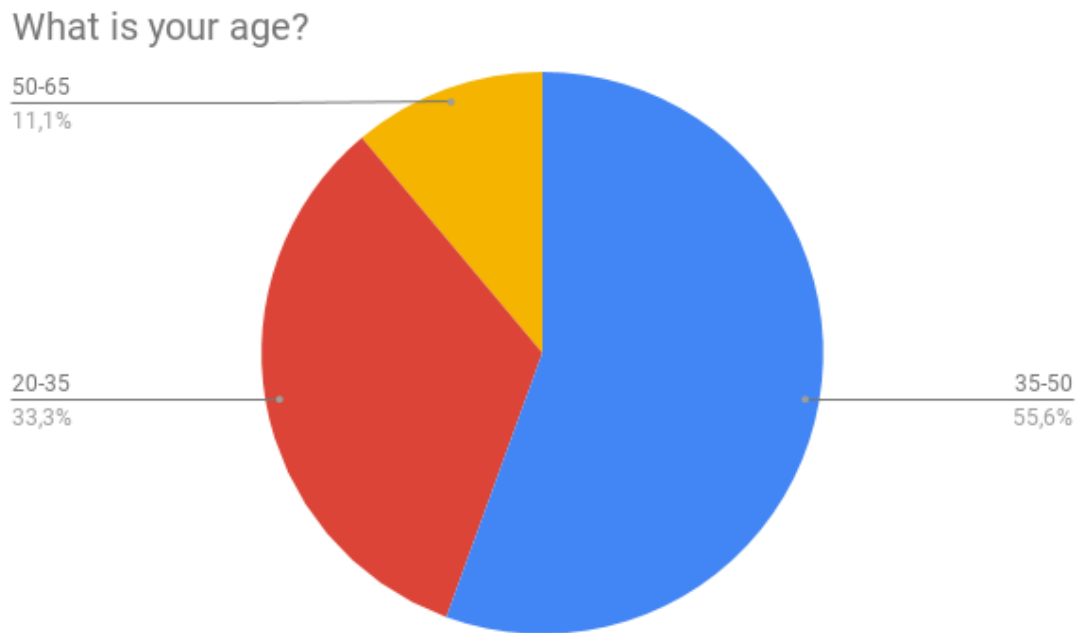


Figure 8.1: The age of the respondents group

8.1.2 *What is your role in the company?*

With this question the author asks about roles in the company of the survey respondents. Responses came from a wide range of positions but if do not take into account the level of a seniority majority is posed by Software Developers. It is natural because they are the asset of a software production company. A wide range of positions of employees who responded to the survey proclaims that survey reached many levels of the organization

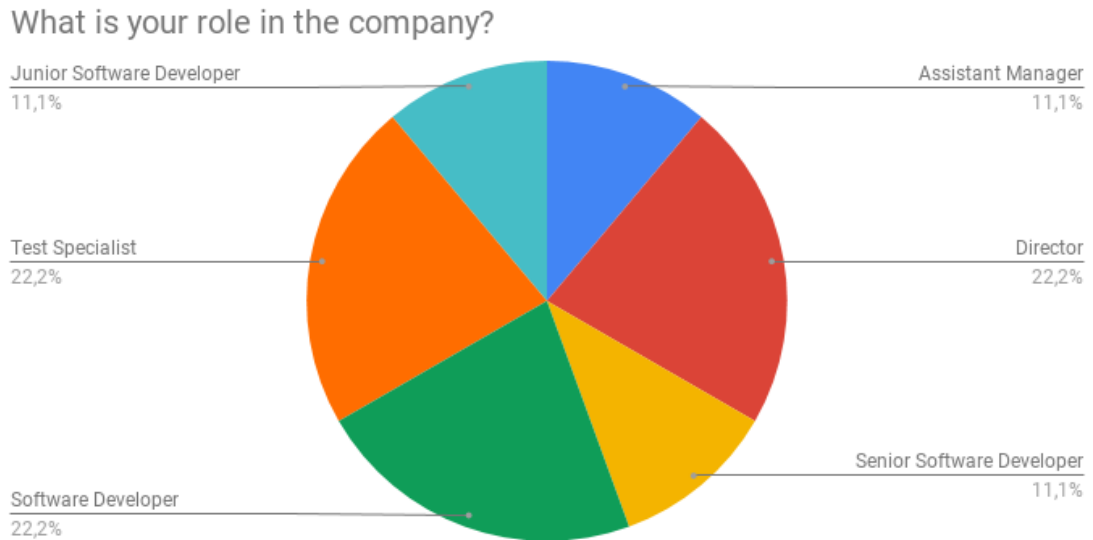


Figure 8.2: The roles of the respondents group in the ILS

8.1.3 *Are you aware of DevOps structure in your organization?*

A purpose of this question was to seclude an option that respondents are not aware of a new structure in the company. Majority of them answered affirmatively. However as it is shown on the Figure 8.3 not all of the respondents are aware of a new structure. It is an early stage of DevOps implementation in ILS. Most probably such responses are a result of ineffective communication of a new model structure. Because the respondents work in this model, probably they know how it works (which is confirmed by the answers to the last question) but they are not aware of its name. Therefore there is a risk that they do not understand its values as well.

Are you aware of devops structure in your organization?

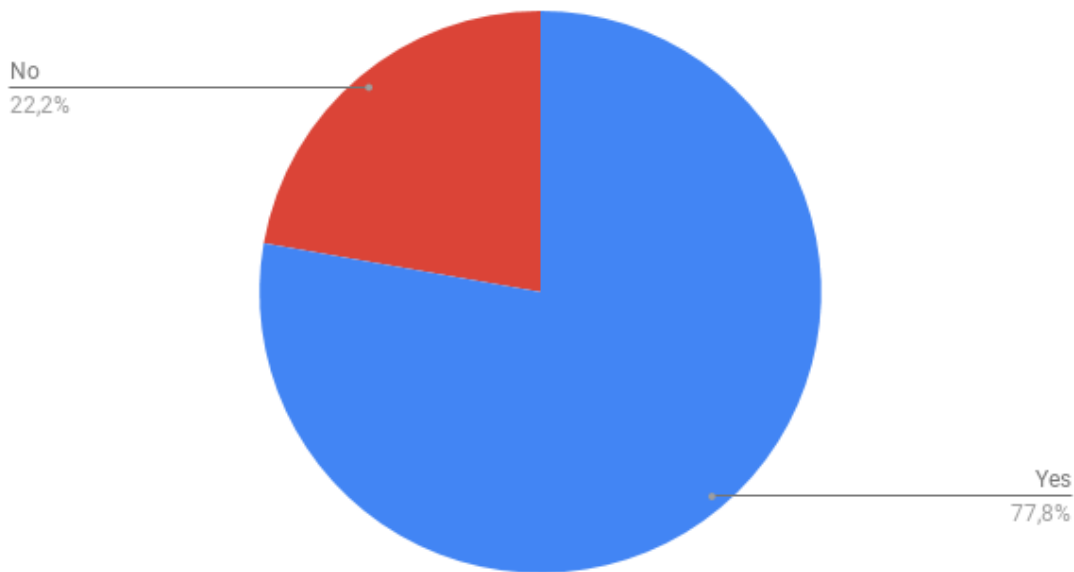


Figure 8.3: A DevOps awareness in the group of respondents

A solution to this problem may be an initiative for the DevOps model's communication improvement by performing a set of workshops for example.

8.1.4 *How do you rate DevOps influence on the communication in your team?*

This question was asked in order to identify an opinion of ILS employees about the DevOps influence on the communication with their colleagues from the team. In the rate range from 0 to 9, 0 mean definitely negative impact and 9 - definitely positive. As expected, majority (77%) respondents assessed this impact on 5 or above which means that majority of the employees positively assess an impact of a new structure on the communication among the team. However 22% rated negatively. It may be related to the early stage of DevOps implementation, so probably some of the employees does not use DevOps processes properly yet.

How do you rate a devops influence on the communication in your team?

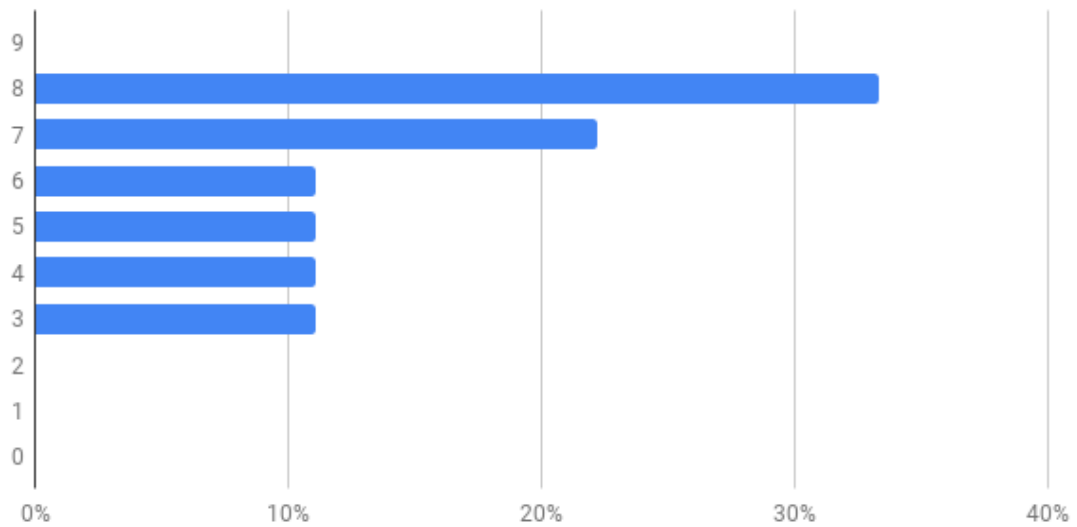


Figure 8.4: An influence of DevOps on the communication according to the respondents

8.1.5 *How do you rate a DevOps influence on the speed of your work?*

Responses to this question were not so positive. Only 66% of respondents rated DevOps impact on the speed of their work at 5 or higher. More alarming is the fact that 33% of them rated this impact at 2 and 0. It means that the employees feels that this new structure slows down their work.

How do you rate a devops influence on the speed of your work?

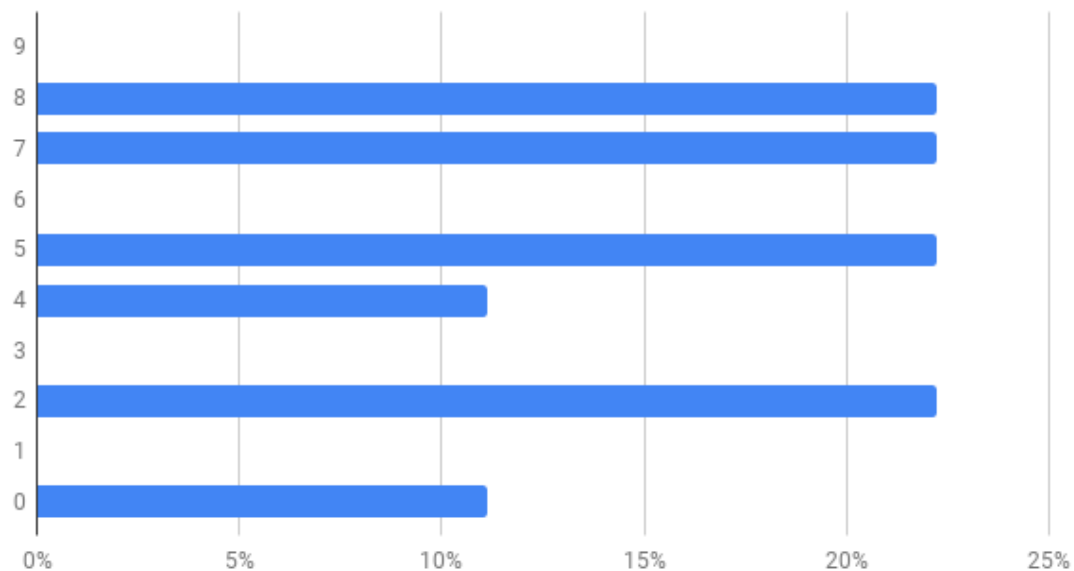


Figure 8.5: An influence of DevOps on the speed of work according to the respondents

The reason may be laying in the fact that not all processes works properly yet. In the transition phase there are still many projects that were opened in the previous structure. Such distraction may result in temporary lags. However, it is an important factor, and the management team should be aware of that.

8.1.6 *Do you feel that DevOps improves your work? How?*

Generally answers were positive. As an improvements respondents described a positive impact on the co-operation with clients. Most of the responses pointed out the improvement on the communication in the team. There were also responses that it is to early to evaluate an influence of a new structure on their work.

8.1.7 *Explain briefly in your own words, what is DevOps about (how DevOps works)*

Answers to this question had different leght and complexity. However most of them was more or less like the example below:

Stages integration of the project development within one or few clients needs

It means that the majority of answers was aligned to what ILS wants to achieve by DevOps implementation

8.1.8 Conclusion

In general employees of ILS positively rate an impact of DevOps on the communication of the teams. Very important aspect that was pointed out by respondents is a customer service and co-operation. Not all of the respondents are aware of an implemented organizational structure what may suggest, that it was not properly communicated. DevOps implementation is a process, therefore there is a need for more initiatives regarding an employees preparation for a change.

8.2 Survey to the management team

The management team include three directors of all business lines. In this case the attendance was 100%. This survey comprised four question about their subjective rating of DevOps influence on a performance of their teams. The rating range was from 0 to 9 where 0 is definitely negative and 9 is definitely positive.

8.2.1 *How do you rate an impact of devops on the business results of your team?*

With this question the author investigates what DevOps effects does business lines directors feels. Each respondent answered with different rating but overall the responses were positive.

How do you rate an impact of devops on the business results of your team?

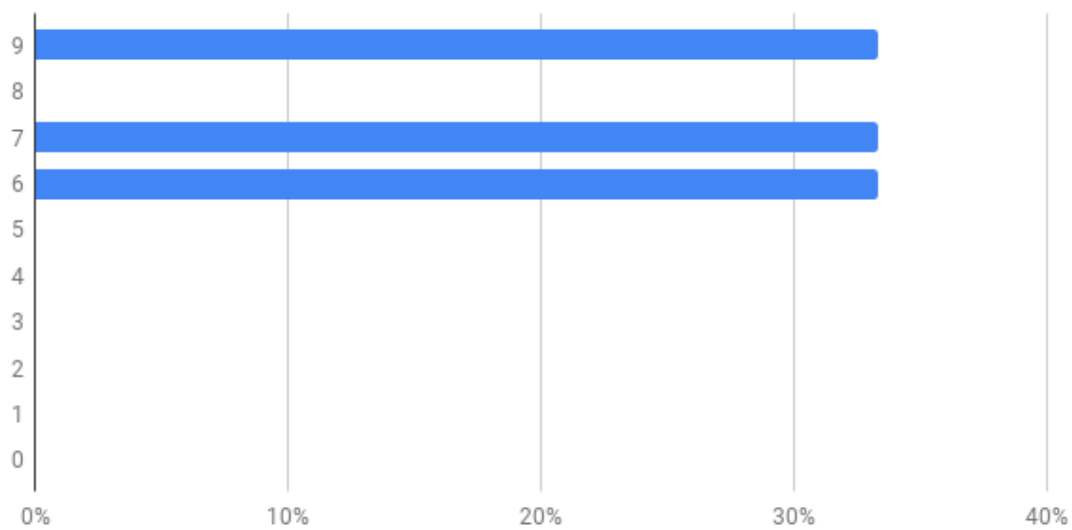


Figure 8.6: An influence of DevOps on the business results according to the respondents

8.2.2 *How do you rate a devops influence on the communication in your team?*

A purpose of this question was to identify how does the communication among the employees of each team is being affected, and how does the management team see it.

All of the responses were at high rating which in general reflects the responses of the employees.

How do you rate a devops influence on the communication in your team?

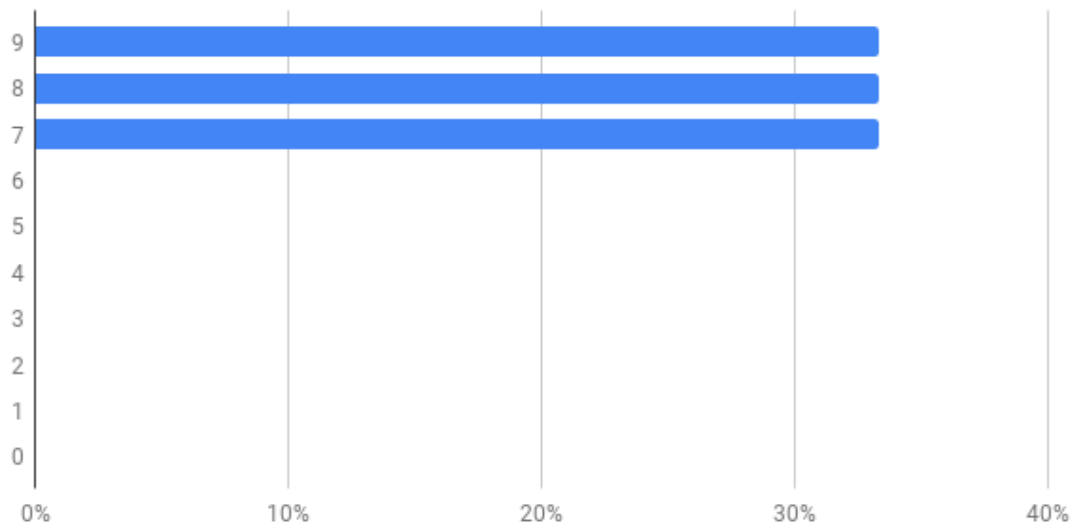


Figure 8.7: An influence of DevOps on the communication among the teams according to the respondents

8.2.3 *How do you rate a devops influence on the work speed of your team?*

That question was asked in order to identify a perception of a teams work speed improvement from the management team point of view. All directors rated this impact generally positively, however none of them gave the highest rate. Such result is not aligned with the responses of the employees. Among the answers of employees there was also a rate at values 2 and 0.

How do you rate a devops influence on the work speed of your team?

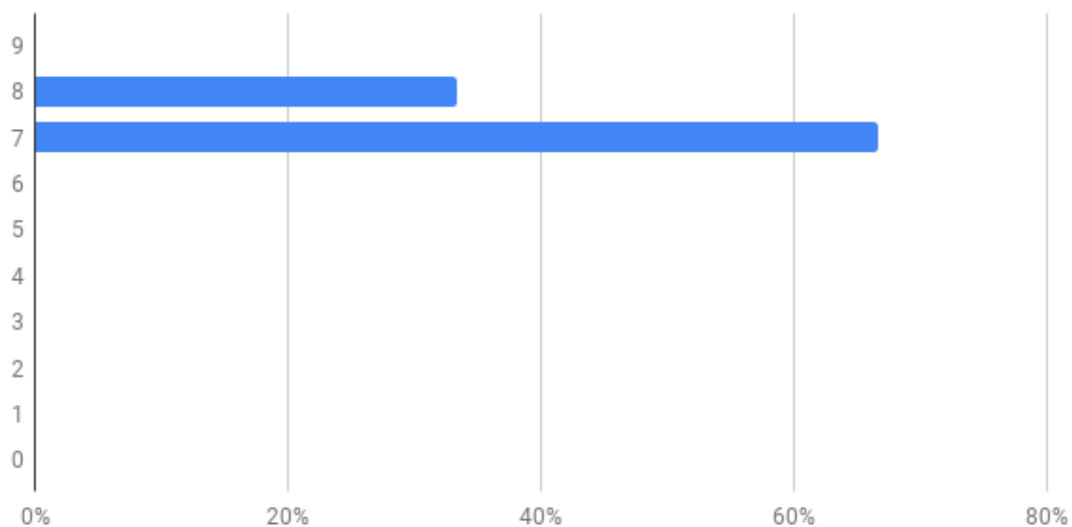


Figure 8.8: An influence of DevOps on the work speed of the teams according to the respondents

That may be brought by fact that the employees perceive a tempo of their work in scope of one or few projects. Some of these project probably were started at the time before the new structure, and by shuffling the responsibilities project management became less efficient. At the same time directors of the business lines answer in the scope of the whole team so they have perception of a number of projects.

8.2.4 *How do you rate a DevOps influence on the communication with your clients?*

The result of this question highlights an importance of DevOps. All respondents gave a high rates to this question. An improved communication with clients was also mentioned in the responses to the open question about the other impacts of DevOps on the work of the respondents in the previous survey. It is a good sign, that emphasize the fact that DevOps improves a performance of a company.

How do you rate a devops influence on the communication with your clients?

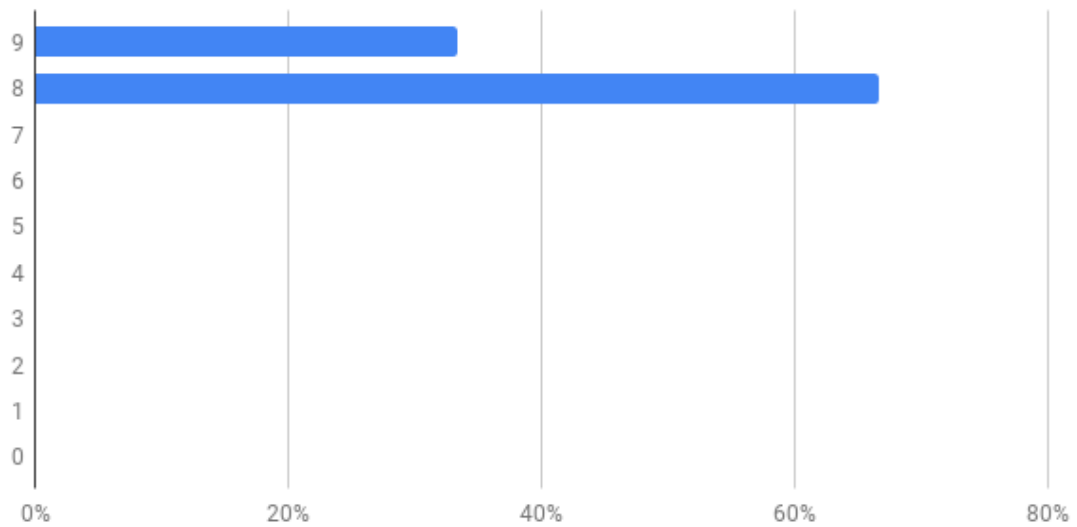


Figure 8.9: An influence of DevOps on the communication with clients according to the respondents

8.2.5 Conclusion

The results of this survey shows, that the communication between the employees as well as with clients is improved thanks to DevOps. However there is something of discrepancy between the responses of the employees and the management team in terms of the question about the speed of work improvement. It may be caused by fact, that the employees answer to the questions in scope of the projects they are into, while the management team applies a scope of a whole team.

Conclusion 9

DevOps is an interesting proposition for a companies that want to optimize their processes and improve a communication of their teams as well as with the clients. The results of the surveys made by the author of this thesis confirmed a positive influence on a communication in the teams and among the employees. It also confirmed an improved communication with clients.

It is hard to confirm the positive influence of DevOps on the processes inside the company yet. The results of surveys among the all employees of ILS as well as in the management team were slightly misaligned in that matter. Most probably it is a to early stage to infer the impact of DevOps on the speed of work and improvement of processes in ILS.

In the nearest future ILS decided to implement Key Performance Indicators set which was elaborated in the collaboration with the author of this master thesis, and use them in order to improve the processes and procedures in the company. KPI should be reviewed periodically in terms of its usefulness, for example once a quarter. Each KPI review should include a deep analysis of the circumstances which impacts them.

KPI should be used as a control alerts for an optimization problems identification. However any decisions should not be based directly on them. The value of each KPI is being affected by many factors therefore its level does not give a picture of the whole incident.

As the author has shown, each indicator affects or may affect a value of another one. They are dependent to each other and therefore KPI should be interpreted in the scope of the whole set.

DevOps had a positive influence on the performance of ILS. Its implementation is also a process which require time. In the current moment ILS is in the transition process from its previous structure to the new one. Therefore not everything in ILS is properly implemented yet. There are some projects that are dependent on

the old structure. Such structural shuffle cause a partially inefficient performance of the teams. That problem highlights an importance of a need for KPI to enable observation of this implementation process. ILS should also keep working on a communication of DevOps values and procedures. As it was emphasized in the results of the surveys not all employees are fully aware of how does DevOps works.

The DevOps structure is a real solution for companies for their problems with inefficient processes. However implementation of it is time consuming and it needs a time to be fully and properly implemented. Definitely it is able to identify its positive influence on the ILS performance. DevOps improves communication among the team as well as with the customers, and a customer satisfaction is the most important result of the company's performance.

Bibliography

Adela del-Río-Ortega and Manuel Resinas. Towards modelling and tracing key performance indicators in business processes. pages 57–67, 2009. ISSN 1988-3455.

AlertOps. Mtttd vs. mttf vs. mtbf vs. mttr, 2018. URL <https://alertops.com/mttd-vs-mttf-vs-mtbf-vs-mttr/>.

Scott W. Ambler. *The Object Primer 3rd Edition*. Cambridge University Press, 2004. ISBN 0-521-54018-6.

Andreas Opelt, Boris Gloger, Wolfgang Pfarl. *Agile Contracts*. John Wiley & Sons, Incorporated, 2013. ISBN 9781118640128.

Alanna Brown. Cloud automation: Paypal's openstack secret sauce, 2013. URL <https://puppet.com/blog/cloud-automation-paypal-s-openstack-secret-sauce>.

CollabNet, VersionOne. the 12th annual state of agile repor. page 16, 2018.

Aliza Earnshaw. 5 kpis that make the case for devops, 2013. URL <https://puppet.com/blog/5-kpis-make-case-for-devops>.

Damon Edwards. What is devops?, 2010. URL <http://dev2ops.org/2010/02/what-is-devops/>.

Gail Ferreira. Use lean thinking to accelerate devops performance for agile teams, 2015. URL <https://insights.stackoverflow.com/survey/2016#developer-profile-age>.

Michael Hüttermann. *DevOps for Developers*. Apress, 2012. ISBN 9781430245698.

Jon Jenkins. Velocity culture - the unmet challenge in ops, 2011. URL <http://assets.en.oreilly.com/1/event/60/Velocity%20Culture%20Presentation.pdf>.

Arie van Bennekum Alistair Cockburn Ward Cunningham Martin Fowler James Grenning Jim Highsmith Andrew Hunt Ron Jeffries Jon Kern Brian Marick

- Robert C. Martin Steve Mellor Ken Schwaber Jeff Sutherland Dave Thomas Kent Beck, Mike Beedle. Manifesto for agile software development, 2001. URL <http://agilemanifesto.org/>.
- L. Dooley, G. Lupton, D. O'Sullivan. Multiple project management: a modern competitive necessity. pages 14–15, 2004.
- Ernest Mueller. What is devops?, 2010. URL <https://theagileadmin.com/what-is-devops/>.
- National Service Knowledge Network. The incident command system, 2019. URL https://www.nationalservice.gov/sites/default/files/olc/moodle/ds_online_orientation/viewf265.html?id=3139&chapterid=908.
- David Parmenter. *Key Performance Indicators : Developing, Implementing, and Using Winning KPIs*. John Wiley & Sons, Incorporated, 2015. ISBN 978-1-4842-1067-3.
- Joshua Partogi. Scrum and devops, 2018. URL <https://dzone.com/articles/scrum-and-devops>.
- Dan Radigan. Kanban how the kanban methodology applies to software development, 2019. URL <https://www.atlassian.com/agile/kanban>.
- F. John Reh. The basics of key performance indicators, 2018. URL <https://www.thebalancecareers.com/key-performance-indicators-2275156>. (visited March 1, 2019).
- Richard Sagor. Guiding school improvement with action research, 2000. URL <http://www.ascd.org/publications/books/100047/chapters/What-Is-Action-Research%2%A2.aspx>.
- Stack Overflow. Developer survey results, 2016. URL <https://www.cmcrossroads.com/article/use-lean-thinking-accelerate-devops-performance-agile-teams>.
- Thomas Stober and Uwe Hansmann. *Agile Software Development Best Practices for Large Software Development Projects*. Springer-Verlag Berlin Heidelberg, 2010. ISBN 978-3-540-70832-2.
- University of Southern California. Organizing your social sciences research paper: 5. the literature review, 2019. URL <http://libguides.usc.edu/writingguide/literaturereview>.

Tathagat Varma. *Agile Product Development*. Apress, Berkeley, CA, 2015. ISBN 9781119019848.

Matt Watson. 15 metrics for devops success, 2017. URL <https://stackify.com/15-metrics-for-devops-success/>. (visited March 1, 2019).

Andreas Wittig and Michael Wittig. *Amazon Web Services in Action*. SManning Press., 2016. ISBN 978-1-61729-288-0.

Survey to all employees **A**

ID	Start time	Completion time	What is your age? --- W jakim jesteś wieku?	What is your role in the company? --- Na jakim stanowisku pracujesz?	Are you aware of devops structure in your organization? --- Czy jesteś świadom/świadoma struktury devops w Twojej firmie?	How do you rate a devops influence on the communication in your team? --- Jak oceniasz wpływ devops na komunikację w Twoim zespole?	How do you rate a devops influence on the speed of your work? --- Jak oceniasz wpływ devops na prędkość Twojej pracy?
1	17/5/19 9:24:52	17/5/19 9:29:58	35-50	assistant manager	Yes / Tak	6	5
2	17/5/19 9:36:22	17/5/19 9:40:54	35-50	dyrektor linii biznesowej	Yes / Tak	7	7
3	17/5/19 16:16:03	17/5/19 16:22:54	35-50	Dyrektor ds. Rozwoju Oprogramowania	Yes / Tak	8	8
4	22/5/19 14:44:21	22/5/19 15:00:40	20-35	Senior Analyst/Software Developer	Yes / Tak	8	2
5	22/5/19 11:24:07	23/5/19 10:50:52	20-35	Developer	Yes / Tak	7	7
6	23/5/19 11:50:41	23/5/19 12:07:09	35-50	Specjalista ds testów i wdrożeń	No / Nie	5	5
7	23/5/19 15:11:18	23/5/19 15:13:23	20-35	Junior Programist	Yes / Tak	8	8
8	24/5/19 6:45:09	24/5/19 8:03:25	50-65	analityk-programista	Yes / Tak	3	0
9	24/5/19 10:30:43	24/5/19 10:35:51	35-50	Specjalista ds. testów i wdrożeń	No / Nie	4	4

Do you feel that devops improves your work? How? (Write "not" if so)

Czy czujesz że devops usprawnia Twoją pracę? (Napisz "not" jeśli nie)

Not really

kompleksowa koordynacja projektu do realizacji - od analizy, przez programowanie, po testy + kompleksowe utrzymanie

Tak - lepsza obsługa klienta - poprawa komunikacji

DevOps model is quite new in my company. I noticed that the communication in my team is much better but still we have problem with organisation of working time, testing, development and delivery processes.

Yes

Don't really know yet. I think it's too early to me too judge his job.

Yes, it impacts a lot a communication across the team

not

średnio, zależy od projektów

Explain briefly in your own words, what is devops about (how devops works)

Opisz krótko swoimi słowami, na czym polega devops

describes software development life cycle

zespolenie ludzi w ramach potrzeb jednego/kilku klientów

DEVOPS - to uwzględnienie we wdrożeniach zarówno części operacyjnej jak i wytwórczej w ramach tego samego cyklu - oraz próba poprawy komunikacji pomiędzy działami operacją i wytwórstwem IT

Devops is a software development methodology / development model t emphasizes communication, collaboration and integration between developers and operation specialists.. It is based on Continuous Integration, Continuous Delivery, Continuous Testing, Continuous Monitoring and Continuous Deployment processes.

Communication between employees in the implementation of projects, the use of appropriate tools

I admit that don't know his scope, but i wat to believe, that his actions make our work would be more effective.

DevOps is a framework that integrates people with different capabilities in a one team

Dla mnie to ścisła współpraca (synergia) nie tyle pomiędzy zespołami utworzonymi w wyniku podziałów administracyjnych, ile między ludźmi związanymi wspólnym projektem, uwzględniając nie tylko techników od infrastruktury serwerowej, ale także ludzi po stronie biznesu. Współpraca, a więc wypracowanie wspólnego kompromisu pomiędzy wymaganiami, możliwościami i terminami. Nie polega natomiast na wprowadzeniu jedynie narzędzi do komunikacji między ludźmi, narzucaniu terminów i egzekwowaniu ich pod groźbą kary.

Współpraca między zespołami, przepływ wiedzy i informacji, metodyka przy częstych cyklach wydawniczych, itp.

Survey to the management team

B

ID	Start time	Completion time	How do you rate an impact of devops on the business results of your team? --- Jak oceniasz wpływ devops na wyniki biznesowe Twojego zespołu?	How do you rate a devops influence on the communication in your team? --- Jak oceniasz wpływ devops na komunikację w Twoim zespole?	How do you rate a devops influence on the work speed of your team? --- Jak oceniasz wpływ devops na prędkość pracy Twojego zespołu?	How do you rate a devops influence on the communication with your clients? --- Jak oceniasz wpływ devops na komunikację z klientem?
1	17/5/19 16:30:04	17/5/19 16:31:49	6	8	7	8
2	17/5/19 16:32:37	17/5/19 16:33:16	9	9	8	9
3	17/5/19 16:32:41	17/5/19 16:35:34	7	7	7	8