

Aligning agile software development with enterprise architecture framework

Karolis Noreika
Kaunas, Lithuania
E-mail: knoreikos@gmail.com

Saulius Gudas
Institute of Data Science and Digital Technologies
Vilnius University
Vilnius, Lithuania
E-mail: saulius.gudas@mii.vu.lt

Abstract. *The effectiveness of internal processes is a key in modern day economy for companies of all sizes. This also includes the effectiveness in software development management and its alignment with business goals both short and long term. But it is not always easy to align IT development with organizational goals. This paper suggests a method for aligning modern software development approaches with enterprise architecture frameworks.*

Keywords: *agile software development, business and IT alignment, enterprise architecture framework, TOGAF*

I. INTRODUCTION

Agile approach is a popular software development methodology. Agile approach currently is being adopted to business strategy execution, decision making to achieve strategic goals. Companies are “going agile” [1] in order to improve productivity of software teams as well as business teams making business decisions. “Going agile” is a big organizational change. It means that employees in all levels of organization will need to adapt to the new way of working, which is getting the results of their daily duties evaluated much faster than in the traditional way of working. However, when “going agile”, the overall goals of the organization are not always supported with an organizational change. There are researches that emphasize the importance of supporting the agile way of working from organizational perspective (provide appropriate physical atmosphere, work environment that encourages creativity) [2]. The gaps between business and IT strategies appear. It might result in not sufficient quality of software products, that are not in line with overall goals of the organization both short and long term.

Eventually the misalignment becomes so significant that organizations get into a position when there is no way back –

either decommission the system or accept the extremely costly support of it (i.e. mainframe systems in financial institutions).

Enterprise architecture is a well-defined practice for conducting enterprise analysis, design, planning, and implementation, using a comprehensive approach at all times, for the successful development and execution of strategy [3]. The agile methodology life cycle could be structured and aligned with TOGAF life cycle, which is a standard for enterprise architecture development. TOGAF is a framework – a detailed method – for designing, planning, implementing, and governing an enterprise architecture [4], [10].

This paper proposes a methodology for how agile methodology life cycle can be aligned with TOGAF enterprise architecture framework.

II. AGILITY IN BUSINESS MANAGEMENT AND SOFTWARE DEVELOPMENT

Agile approach allows business representatives to see the value of the software product being developed faster compared to traditional software development. Traditional or “waterfall” software development dates back to around 1970ties when the development of large enterprise IT systems was started to be described in a scientific way [5].

The waterfall methodology utilizes the idea that each phase in software development is sequential and cannot repeat. The agile methodology promotes the idea of repeated and iterative steps, which are explained in Fig. 1 below.

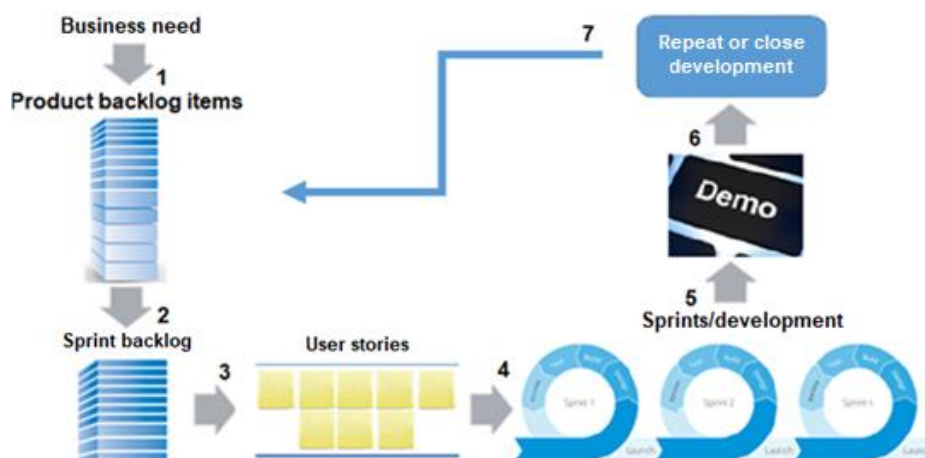


Fig. 1. THE CONCEPTUAL DIAGRAM OF THE AGILE SOFTWARE DEVELOPMENT

Fig. 1 explains how the software product is being developed after the business need is received. It contains the whole agile life cycle which is organized by having different

information/knowledge flows. The detailed description of each of the elements is in table 1 below.

TABLE 1. DESCRIPTION OF AGILE LIFE CYCLE ELEMENTS

Step/element no.	Step/element name	Step/element description
1	Data flow no. 1	The data flow, containing the information needed for understanding the business need, problem. It is transformed into product backlog item (element no. 2)
2	Product backlog items	List of features and requirements that the solution should have once completed.
3	Data flow no. 2	An incoming data flow for the next element in the life cycle – sprint backlog (element no. 4). It contains the features that software should contain after development iteration – sprint.
4	Sprint backlog	List of features that will be developed in the next sprint. Sprint is a time frame with a list of features described and approved by business and IT representatives.
5	Data flow no. 3	Once the high level features to be developed in the next sprint are agreed upon – the details must be clarified to the level needed in order to accomplish the business needs. This data flow contains the sprint backlog items or features explained in smaller pieces of information or requirements – user stories.
6	User stories	The detailed requirements are worked on – analyzed by the development team so that both IT and business representatives understands the problem each user story would solve.
7	Data flow no. 4	The detailed user stories are placed in some software tool that would allow keeping track of the progress of development of user stories.
8	Sprints/development	This is the most beneficial part of the agile life cycle. It is a method of constantly developing small part of the overall software solution and getting the feedback fast. Each sprint consists of: a) Design – designing the user interface, business rules placed in the solution. b) Build (develop) – coding, styling, working on the solution from development perspective. c) Test – test the developed solution against the requirements. d) Review – review the solution test findings and decide what to improve. e) Launch – after the items that were agreed to be developed at the end of sprint are verified against the solution itself and the found changes that were necessary to do are done, the project team decides should the solution at current stage be deployed into production (or live) environment, where it could be already used by business representatives. Note: agile promotes the approach that the project team should be able to continue the iterative development for indefinite amount of time. Therefore, the number of sprints with the same phases as mentioned above could continue indefinitely.
9	Data flow no. 5	The information gathered from the business need at the beginning of the project and throughout development phase aggregated to prepare the demo of the solution.
10	Demo	The demo for the solution is a system presentation conducted to all relevant stakeholders.
11	Data flow no. 6	The decision after the demo whether the solution should be included into production environment or should the development continue with taking the next set of requirements made.
12	Repeat or close development	As Agile methodology describes – self-organizing team should be capable of keeping the accepted efficiency for product development indefinitely. This means that if business managers decide – the team should be able to repeat the whole cycle indefinite amount of times until the repetition does not increase the value significantly. If the decision is made to close the project – the agile life cycle is completed.
13	Data flow no. 7	If it is decided to continue development – the next set of requirements is taken from the product backlog items list (element no. 2) and the agile life cycle is repeated.

There are a lot of details and techniques how agile life cycle should be managed to achieve the best efficiency [6], [7], [8], but this is not a subject of this paper.

However, running a successful business is not only about doing software development in an agile way. Often IT development is ahead of business decisions where to come up with a suitable software solution development teams needs quick decisions by business that might be applicable across

multiple projects in same business area (i.e. store related documents in single repository, have same classification of them, etc.) The business side in the enterprises is also starting to take decision based on agile methodology, although it is often perceived as a part of startup culture – i.e. not something established and large organizations would do. Table 2 below represents the comparison of agile and traditional approach on business decision making in agile and traditional ways.

TABLE 2. AGILE AND TRADITIONAL DECISION MAKING COMPARISON

Methodology	Flexibility	Risk	Adapting to change	Amount of data needed to make decision
Agile (including variations)	Higher	Higher	Faster	Smaller
Traditional	Lower	Lower	Slower	Larger

Agile methodology could be applied to business decision making by mapping the agile phases to decision making process – i.e. limit the information needed to make decision could be compared to sprint backlog. Having a deadline for a decision could be understood as the date for demo. Adjust to new information on the decision could be understood as review part of sprint.

III. IDENTIFICATION OF GAPS BETWEEN BUSINESS AND IT STRATEGIES

A. Business and IT alignment model

The business and IT alignment model was created by Henderson and improved by Venkatraman to represent business strategy alignment with IT strategy thus providing analysis method aimed for competitive advantage [9]. Fig. 2 below represents the strategic alignment model.

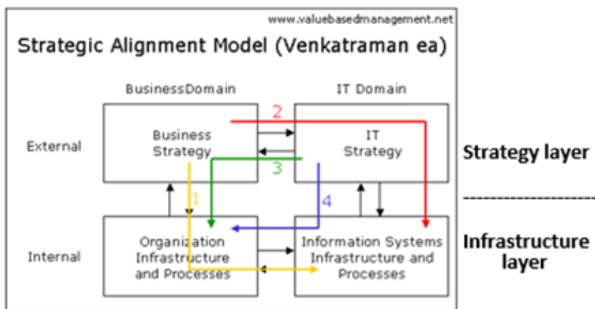


Fig. 2. THE BUSINESS AND IT ALIGNMENT MODEL

There are four domain alignment perspectives where each focuses on different aspect of alignment between the business and IT alignment, i.e.:

1) Strategy execution – business strategy is the driver for organization design changes and the logic of IT infrastructure. In this perspective, the top management of the organization dictates the strategy of the company and the IT management is the strategy implementer.

2) Technology potential – business strategy is the driver for change, however it is closely aligned with IT strategy as well, therefore the IT systems are more aligned with IT strategy and also business strategy.

The top management should provide the vision of the technology to articulate the logic and choices to IT strategy what would best support the chosen business strategy. The role of IT manager in this perspective should be of the technology architect – the IT manager should efficiently and effectively design and also implement information system infrastructure that is consistent with the IT strategy. This alignment perspective could be used for aligning business and IT strategy along with IT systems in an agile way as vision is

also one of key aspects to have for the agile development teams to be successful and self-organizing.

3) Competitive potential – focuses on utilizing emerging IT capabilities to impact new products and services also to influence key attributes of strategy (distinct competences) as well as form new relationships (business governance). This perspective also allows the changing of business strategy via emerging IT capabilities. The role of the management is of business visionary who dictates how emerging IT competences and functionality would impact the business strategy. The role of IT manager is of the one who identifies and interprets the trend in the IT environment to assist the business managers to understand the potential opportunities and threats from an IT perspective and handle them accordingly.

4) Service level – this perspective focuses on building world class IT team. Therefore, the role of IT manager is also of a business leadership with tasks of making the internal business succeed with the operating guidelines from top management.

B. TOGAF

TOGAF is framework for designing, planning, implementing, and governing an enterprise information technology architecture. The TOGAF standard includes a content framework to drive the Architecture Development Method (ADM). TOGAF is an iterative process model (enterprise architecture development life cycle) supported by best practices and a re-usable set of existing architecture assets. TOGAF supports Capability-Based Planning of enterprise architecture [10].

The TOGAF framework is presented in Fig. 3 below.

Enterprise architecture development life cycle (defined in TOGAF) could be used for analysis of the agile software development approach.

The TOGAF life cycle in Fig. 3 was transformed to a schematic view of a table (Fig. 4) in which the columns represent the phases of TOGAF enterprise architecture development life cycle and agile methodology life cycle and the activities in the intersecting sections – the phases of agile development (design, build, test and deploy) [11]. This approach could be used into applying agile way of working for building up and aligning with enterprise architecture implementation that TOGAF provides.

Although companies can change strategies quickly, they then face the big slowdown of executing one or several strategies. For enterprise architects, this has traditionally meant defining a new target state, comparing it with the current state, and then developing a road map. But this multistep process is now perceived as taking too long — by the time EA has all of these documented and approved, the business will have moved on [12].

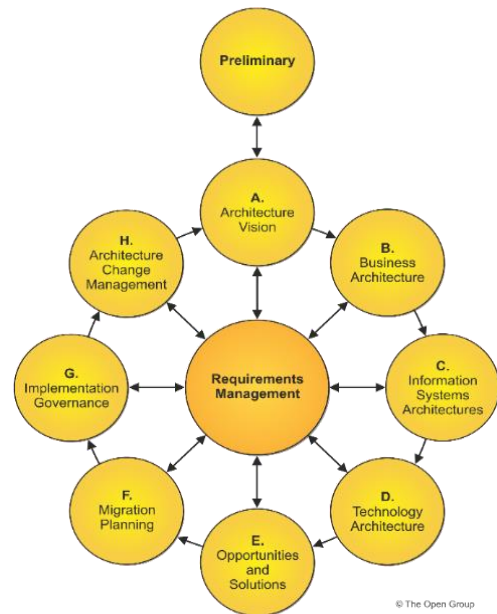


Fig. 3. ENTERPRISE ARCHITECTURE DEVELOPMENT LIFE CYCLE TOGAF (<https://www.opengroup.org/togaf>)

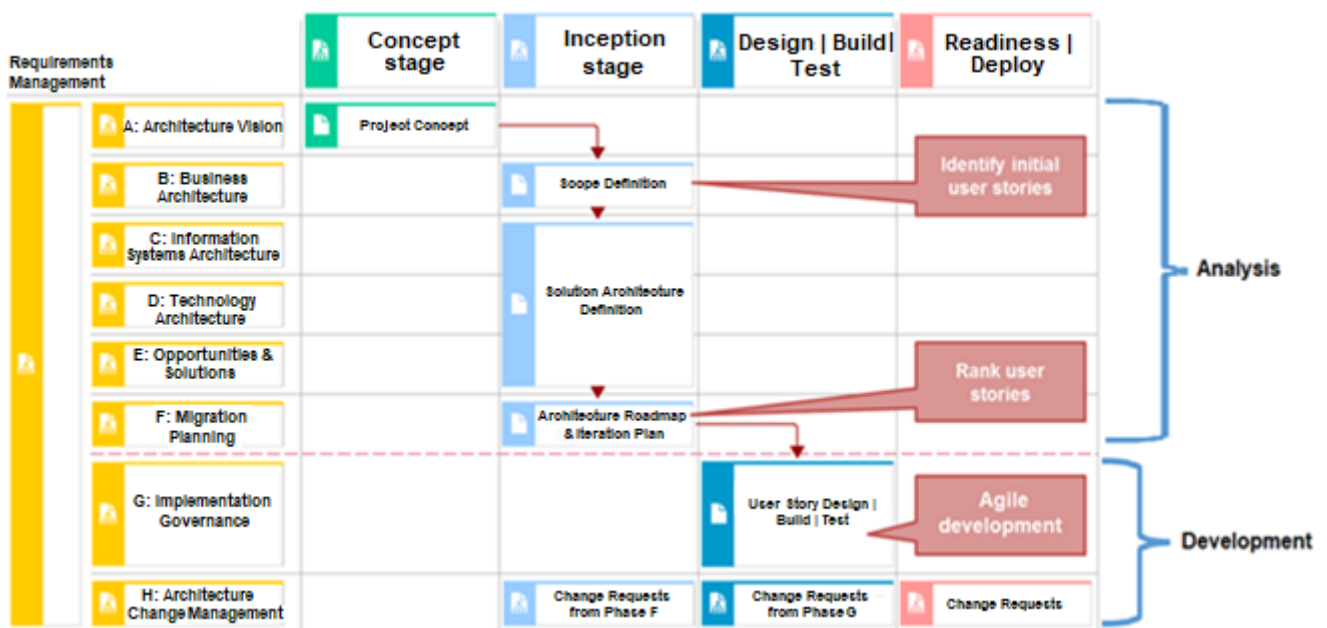


Fig. 4. TOGAF ENTERPRISE ARCHITECTURE FRAMEWORK AND AGILE METHODOLOGY ALIGNMENT MODEL

IV. ALIGNING AGILE LIFE CYCLE WITH ENTERPRISE ARCHITECTURE FRAMEWORK

It is a common belief that TOGAF and also other large enterprise architecture frameworks are “waterfall”. This is a common misinterpretation largely due to these models encompassing all related IT activities and not specific. But basically all these enterprise architecture frameworks are sets of tools, similar like agile where one also should choose the tools and methods suitable for each specific case. A problem in large organizations is that there are different levels of maturity of agile of different teams. Business representatives

(also called stakeholders, subject matter experts or in agile – product owners) represent the business only fragmentally – whenever there is a question regarding IT and business alignment – it is solved on ad-hoc basis, but a long term IT and business strategy should be capable of answering these questions on a higher – strategic – level which is orchestrated by using the TOGAF methodology.

The idea behind mapping TOGAF to agile life cycle is use the strategic vision that TOGAF provides by using its framework and utilize the benefits of agile continuous

improvement and “inspect and adapt” approach. The suggested mapping is displayed in Fig. 5.

When TOGAF is used for overall overview on the enterprise architecture and agile is used for project’s iterations, the business gets benefit from even faster deliveries and projects are aligned with business goals at all times.

V. CASE STUDY

Large enterprises often combine the IT infrastructure „in-house“ together with outsourcing it. It could be only storing part of data or all the data of the enterprise. These decisions are made according to IT strategy mostly and not always these decisions are aligned with business strategy. As the

technology advances to the cloud based solutions more and more companies are concerned about the safety of the data in the cloud based systems. Combining these concerns with the agreed service level agreements provided by external vendors not being maintained for enterprises to run their operations smoothly (i.e. important IT system being outsourced is not working part of the day due to agreed service level agreement breached) might lead to decisions to insource the IT and IS infrastructure. But the cost of such decisions is very dependent on the level of alignment between IT and business strategy and the less is the alignment, the bigger are the costs. Whenever an enterprise is faced with such decision, it is very important to keep the alignment between business and IT strategies moving on.

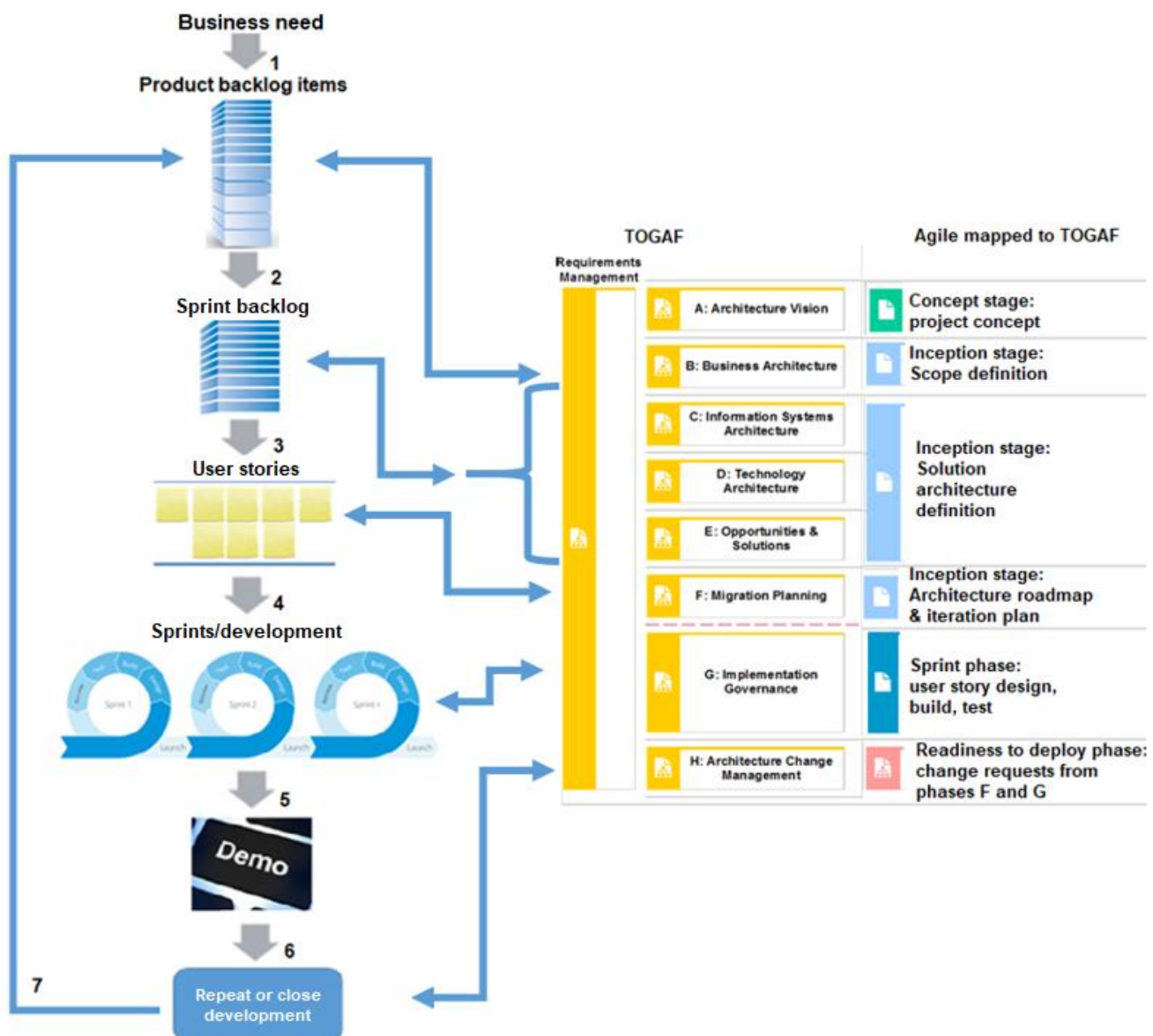


Fig. 5. THE SIMPLIFIED MAPPING OF TOGAF TO AGILE LIFE CYCLE

By using the TOGAF enterprise architecture framework and agile methodology alignment suggested in chapter 4, the company facing such decision might significantly reduce the cost and impact of the migration from outsource provider to in-house solution by constantly aligning the enterprise architecture which TOGAF describes with the constant feedback, “inspect and adapt” approach that agile promotes.

The case where such suggestion was made was about large enterprise moving over 2000 servers of different purposes from outsource to in-house. When using the suggested method of enterprise architecture framework TOGAF being aligned with agile methodology the implementation of the change could have taken at least 10 % less effort both in terms of cost and time needed for the change as the comparison of activities

by using only agile methods and the suggested method showed. Also it is worth noting that this situation could have been avoided if all the tools and methodologies mentioned in this paper were used: IT and business alignment model for overseeing the potential IT infrastructure decisions, TOGAF for overseeing enterprise architecture and the TOGAF and agile alignment model suggested by this paper which helps to see the potential gaps between agile software development and business strategy much faster.

VI. CONCLUSIONS

The agile way of working is something that in some enterprises is new but others are already very far away in implementing this approach into daily decision making process both business and software development. These decisions need to be constantly aligned with the overall business strategy to have the effective enterprise run smoothly. Therefore, it is very important to align the enterprise architecture of the organization with agile approach to make the most benefit of enterprise architecture framework like TOGAF, which provides the tools to ensure business and IT alignment whereas agile provides the speed and the possibility to adapt to changes. Method, suggested in this paper, supports utilization of those mentioned benefits from both tools and allows to improve not only software development process which agile supports, but also keep the alignment between IT and business strategy by constantly keeping IT projects aligned with business strategy which TOGAF supports to make sure right solutions are developed and aligned with long term goals of the enterprise. The proposed approach could be further improved through the use on different types of organizations (i.e. financial, trade, manufacturing) and adapting it in a generalized way for further usage.

REFERENCES

- [1] M. Pikkarainen, J. Haikara, O. Salo, P. Abrahamsson, J. Still, Pikkarainen, M., Haikara, J., Salo, O. et al. "Empir Software Eng" (2008) 13: 303. <https://doi.org/10.1007/s10664-008-9065-9>
- [2] Crawford, Broderick & León de la Barra, Claudio & Soto, Ricardo & Misra, Sanjay & Monfroy, Eric. (2013). "Agile Software Development: It Is about Knowledge Management and Creativity". ICEIS 2013 - Proceedings of the 15th International Conference on Enterprise Information Systems. 2. 10.5220/0004447802650272.
- [3] Federation of EA Professional Organizations, "Common Perspectives on Enterprise Architecture," Architecture and Governance Magazine, Issue 9-4, November 2013 (2013).
- [4] Dirk Draheim, Gerald Weber "Trends in Enterprise Application Architecture" 2nd International Conference, TEAA 2006, Berlin, Germany, November 29 - December 1, 2006, Revised Selected Papers
- [5] Winston Royce, "Managing the Development of Large Software Systems", Proc. Westcon, IEEE CS Press, 1970, pp. 328-339
- [6] Jeffrey Verret "Implementing Agile Methodology: Challenges and Best Practices" University of Oregon (2018)
- [7] Darrell K. Rigby, Jeff Sutherland, Hirotaka Takeuchi "Embracing Agile" Harvard Business Review 2016, pp.40-48
- [8] Zamudio, Lizbeth & Aguilar, José & Tripp-Barba, Carolina & Misra, Sanjay. (2017). A Requirements Engineering Techniques Review in Agile Software Development Methods. 683-698. 10.1007/978-3-319-62404-4_50.
- [9] Henderson, John and Venkatraman, N." Strategic alignment: A model for organization transformation via information technology" Working Paper 3223-90. Massachusetts Institute of Technology, 1990, 458 p. ISBN 9781245057264.
- [10] The TOGAF® Standard, Version 9.2 <https://www.opengroup.org/togaf>
- [11] BPM utils.com Enterprise Agile solution Delivery Framework 2017
- [12] Henry Peyret "EA Methodologies Enlarge To Address The New Business Landscape", 2013 <http://entreprise-agile.com/ForresterEA.pdf>