AGILE METHODOLOGY USING SCRUM AND KANBAN

Hristina Spaseska

Course: Information System Management

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Nowadays the interest in the project management is rapidly growing. At its very first beginnings its purpose was providing schedule for the military resources but right after its popularity it started to spread in all different industries. (Schwalbe, n.d.)

The project is defined as a temporary endeavor undertaken to create a unique product, service or result. Many projects arise in the society and the way they were developed, the methods and techniques used for their leading started to be known as project managing. There are three main objectives that determinate the project: scope, time and cost and in the following past years few other objectives were added to this list like quality, human resources, communications and risk management. Also, every project is defined by its own characteristics such as that projects are unique in their own way. The main purpose of doing a project is to achieve a certain goal. Project management has become more hyped term and the new methods of management were needed after many changes happened in the real world such as increasing human knowledge, increased demand for more complex, more sophisticated goods and services and the increased evolution of worlds markets. (Meredith, Shafer & Mantel, n.d.) For one company to be successful the project management plays a main role in their overall performance, that enables realizing projects with positive results. Project management is defined as a process that enables defining, planning, monitoring, controlling and delivering projects. It is described as form for implementing and improving the work in the companies in order to achieve greater results and better overall performance. (Tereso, Ribeiro, Fernandes, Loureiro & Ferreira, 2018).

The traditional project management in years was changed with the agile project management that offered better solution as an answer of the changes happened in the ecosystem. The purpose of this paper is to emphasize the importance of the agile methodology in the IT project management for the company, make comparation between the traditional and agile methodology, but also a comparation between the Kanban and Scrum agile methodologies, and define the advantages and disadvantages of these methods. Goals of the project will be to improve the performance of the companies through these process improvements, to increase the quality of the working processes in the company and to succeed with a whole new process that will enable predicting results, business process agility and overall well-organized management.

1.IT Project Management

Projects differ between themselves they can be large or small and include few or many people and finished in one week or one year. Many new technology strategies impact the projects such as IoT, computers everywhere, cloud computing, 3D printing and so on.IT projects are those that are using hardware, software and network in order to crate product, service or result. An examples of an IT projects are the following:" A car company builds an eclectic car", "Students of Business Informatics are building an application" etc. (Schwalbe, n.d.)

The time, scope and cost play an important role in the successfulness of the IT projects, but also many other factors like the quality, the team, stakeholders and communications have a great impact. Also, a curtail part has the IT project manager that has specific skills and knowledge expertise such as leadership, problem solving, communication skills, critical thinking and conflict resolution. (Schwalbe, n.d.)

The project management includes the following steps: initiations, planning, controlling, execution and closing. These steps are the phases of the management life cycle. The *initiating phase* is conducted by the sponsor of the project in order to make the financial feasibility of the project and define the project scope. *Planning* the most important phase where the overall flow of the project is defined, then *execution step* right allocation of the resources, *controlling* comparing planned with achieved and *closing* end of the project and see if there is a need of additional improvements. (Ibbs & Kwak, 2000)

1.1. Agile Project Management

The new answer for more flexible and efficient projects was presented with the agile approach in order to make software engineering. The very first beginning of using the agile was for developing software in IBM and Motorola. (Halevi, 2010). The agile project management is using the following principles:

- Collaborative effort of self-organizing cross-functional terms
- Adaptive planning
- Evolutionary development
- Early delivery

- Rapid and flexible response to change, and
- Continuous improvement

The agile projects should be consisted of several cycles that are called sprints. So, referring to this every agile project must have the following sprints:

- Pre- defined scope
- Backlog and
- Stages of software development

As main stages in the development software are defined the following:

- Analysis
- Development
- Testing and
- Deployment

The most common agile methodologies are scrum and Kanban.

Characteristics of the agile methodology:

Agile PM environment = (Uncertainty + Unique expertise) x Speed

As shown in the equation upon the most important factors that describe the agile project management environment are the uncertainty, unique expertise and the speed. According to these characteristics we can easily choose if we need the agile methodology for our project. (Halevi, 2010)

1.2. Traditional vs. Agile

At the very first beginnings the companies were using the *traditional* project management model based on the Waterfall model for software development. Many project failures were delivered on the market so, as a result of the increased demand for new approach the *agile* model was presented. First AgileManifesto was introduced and very rapidly gain success and popularity. The main reasons why the agile methodology was more successful was the faster product delivery, the greater customer satisfaction, high quality etc. (Sameen Mirza & Datta, 2019).

The traditional project management is based on the waterfall model for software development, a methodology that has the following stages: requirements specification, design, implementation, verification and maintenance. In the traditional model every phase is sequential, all requirements in the previous stage must be completed so the project can go further to the next stage and testing is done only once after the codes are delivered. Each phase is done without any overlapping, and all the tasks are completed in a certain period. A documentation of the process for the project is done for every stage. There are many advantages of this model such as that the phases are completed in a exact period of time and after completing one stage the next one can start, it is easy for implementation in any company because of its linearity and it does not require big amount of resources. One of the biggest disadvantages is that many problems are left unsolved after finishing the phase and they are automatically are transmitted into the next stage and the testing is done after the whole process is over, also another con is that if the customer wants any change, that cannot be done. (Balaji & Sundararajan Murugaiyan, 2012)

On the other hand, is the agile software. The agile software development makes a big impact on how software is developed nowadays. When we use the term "agile" we want to emphasize the challenges of the new world, employees' values and relationships that make the software developed. (Dyba & Dingsoyr, 2009). The term refers to "moving quickly" and according to the main meaning of the term the agile method is consisted of a team that is responsible for quick response on dynamic and changing requirements from the clients. There are many pros that credit this method such as greater customer satisfaction of efficient and fast delivery of the software and the ability to fast respond on the changes of the project. Also, there are some disadvantages such as that the agile method is very profitable if it stands a word for a small to medium projects, but if the project is too big than it becomes difficult to conduct the whole life cycle of software development, and also another disadvantage is that only senior developers can make a decision on using the agile development method that does not leave a place for the young developers to show in best. (Balaji & Sundararajan Murugaiyan, 2012)

On the following picture are presented the main differences between the traditional and agile project management methods.

Table 1: Traditional vs agile software development

	Traditional view	Agile perspective
Design process	Deliberate and formal, linear sequence of steps, separate formulation and implementation, rule-driven	Emergent, iterative and exploratory, knowing and action inseparable, beyond formal rules
Goal	Optimization	Adaptation, flexibility, responsiveness
Problem-solving process	Selection of the best means to accomplish a given end through well-planned, formalized activities	Learning through experimentation and introspection, constantly reframing the problem and its solution
View of the environment	Stable, predictable	Turbulent, difficult to predict
Type of learning	Single-loop/adaptive	Double-loop/generative
Key characteristics	Control and direction Avoids conflict Formalizes innovation Manager is controller Design precedes implementation	Collaboration and communication; integrates different worldviews Embraces conflict and dialectics Encourages exploration and creativity; opportunistic Manager is facilitator Design and implementation are inseparable and evolve iteratively
Rationality	Technical/functional	Substantial
Theoretical and/or philosophical roots	Logical positivism, scientific method	Action learning, John Dewey's pragmatism, phenomenology

Source: (Dyba & Dingsoyr, 2009)

2.Kanban

The word Kanban is Japanese word that its meaning is single card or labeled. The Kanban system evolved as a subsystem of TPS (Toyota Production System) and first was created to control the inventory in the factory, the production and the supply of all the components and the raw materials. Kanban was used for continuous improvement and automation of the work. The methodology is based on six practices:

- Visualization of the work
- WIP or work in progress
- Managing the flow
- Using explicit policies
- Fast feedback loops
- Improvements based on methods and models

Kanban is not a method of software development it is just a methodology that brings change in the software development lifecycle. As we mention Kanban visualize the work on a

whiteboard using sticky notes, and an electronic board can be used too. How does Kanban work? It is a method that uses cards and visualize work to trigger and control the work process. First, we need to split the work into small pieces called items. Every card symbolizes different work item. So, if we want to start a new work it can be done only if there is any card available, then the work is attached to the card. So, it is a pull system showing all the steps in the workflow, and right after a work is finished is pulled into the next step. On every card is written "who" is working on "what" and the time needed to finish the certain work.

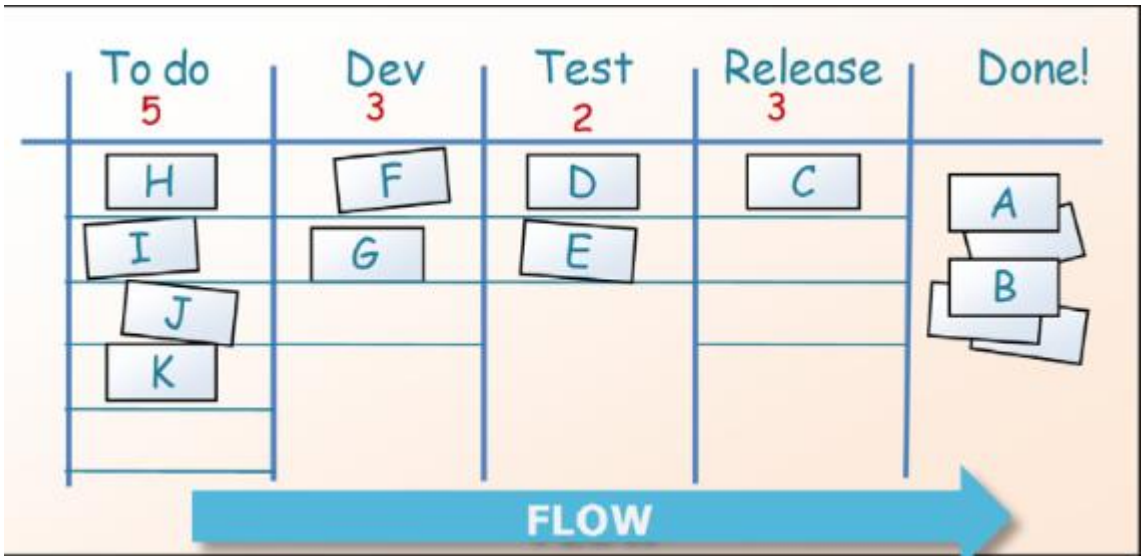

Figure 1: Kanban flow

Source: (Kniberg, Skarin, Poppendieck & Anderson, 2010)

The description of the work is also known as a user story and the card is consisted of avatars and blockers, that are used in case if the work is still not finished or if someone is not present at work. In Kanban different kind of work is done like technical items, bugs or features, so they should all be presented with different colors for instance the bugs with red, the technical items with green and the features with yellow color. Also work in progress (WIP) limits are present here that limit how much work e.g. how many cards can be moved into the work in progress column. (Thor Ingason, Gestsson & Ingi Jonasson, 2013)

3.Scrum

The word scrum origins from the rugby game and has the meaning restarting the game for even a small mistake.

- The work is split into small pieces to the team
- Split the work into a list and sort the work on the list by priorities
- Define a split work time

This methodology is consisted of few roles. Scrum master (SM), team and product owner (PO). The scrum team is usually consisted of 7 members, that are self-managed, and cross functional organized. The product owner and the scrum master are focus on building the product and they must be aware with the product functionalities and their responsibilities are to build and manage a product backlog for the product under development. The PO role is important for:

- Product planning
- Release planning and
- Sprint planning

The working progression here is known as sprint that usually least from 2 to 4 weeks and the SM is responsible for organizing sprint meetings on daily basis where each team discuss about the work in progress. This is a way to react immediately on any obstacles that can appear in the sprint. The progress of the sprints is represented in a burn down chart. (Thor Ingason, Gestsson & Ingi Jonasson, 2013)

Task number	Task description	Status	Owner	Estimated effort (hours)	Remaining effort (hours)
1	Create kitten photo database	In progress	John	12	8
2	Create an application icon	Closed	Bill	6	0
3	Create a home page	Open	John	10	10

Figure 2: Example of burn down chart

Source: (2020)

3.1. Kanban vs. Scrum

The both methodologies Kanban and Scrum can be used for software development. They have many similarities such as having an agile approach but also many differences. In the table above are explained the main differences between the two methodologies.

Table 2: Differences between Scrum and Kanban

Scrum	Kanban
Timeboxed iterations prescribed	Timeboxed iterations optional
Team commits to a specific amount of work for this iteration	Commitment optional
Uses velocity as default metric for planning and process improvement	Uses lead time as default metric for planning and process improvement
Cross-functional teams prescribed	Cross-functional teams optional. Specialist teams allowed.
Items broken down so they can be completed within 1 sprint.	No particular item size is prescribed.
Burn down chart prescribed	No particular type of diagram is prescribed
WIP limited indirectly (per sprint)	WIP limited directly (per workflow state)
Estimation prescribed	Estimation optional
Cannot add items to ongoing iteration.	Can add new items whenever capacity is available
A sprint backlog is owned by one specific team	A Kanban board may be shared by multiple teams or individuals
Prescribes 3 roles (PO/SM/Team)	Doesn't prescribe any roles
A Scrum board is reset between each sprint	A Kanban board is persistent
Prescribes a prioritized product backlog	Prioritization is optional

Source: (Thor Ingason, Gestsson & Ingi Jonasson, 2013)

The main difference starts from the Kanban board that visualize the work in progress with sticky notes and the Scrum board is separated into progress columns for each sprint. In the Scrum methodology there are 3 main roles prescribed: Product Owner, Scrum Master and Team but in the Kanban, there is no prescription of any roles. Scrum is more prescriptive than Kanban – prescriptive means that Scrum has more rules to follow that the Kanban methodology for example in Scrum are prescribed timebox iterations and estimations but that is not same in Kanban where they are optional *(Thor Ingason, Gestsson & Ingi Jonasson, 2013)*

CONCLUSION

The traditional way of software development has main goal to make optimization and on the other hand the agile methodology presents an adoption that is very difficult to predict, also the both methodologies operate in different environment, the traditional in more stabile and predictable and the agile in more turbulent and unpredictable environment.The agile methodology made a big impact on the way many companies operate and make them more flexible, more knowledgeable and adaptable to the changes in the ecosystem. Many agile methodologies were presented but Kanban and Scrum have been widly adopted by many companies. The reason using them is they are very easy to use; both simplify the work and help teams focus on realizing the main goals. Agile methodologies have many advantages for the company like dividing the work into sprints and enabling the team members to focus on individual tasks, separating the project in sprints,faster feedback from the end users and flexibility and adaptability.

References

1. (2020). Retrieved 3 January 2020, from https://study.com/academy/lesson/sprint-burndown-chart-in-scrum-example-overview.html
2. Balaji, S., & Sundararajan Murugaiyan, M. (2012). WATEERFALLVs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC. *International Journal Of Information Technology And Business Management*, *Vol.2*(No. 1). Retrieved from http://jitbm.com/Volume2No1/waterfall.pdf
3. Dyba, T., & Dingsoyr, T. (2009). What Do We Know about Agile Software Development? *IEEE Software*, *26*(5), 6-9. doi: 10.1109/ms.2009.145
4. Halevi, G. (2010). *Handbook of production management methods*. Oxford: Butterworth-Heinemann.
5. Ibbs, C., & Kwak, Y. (2000). Assessing Project Management Maturity. *Project Management Journal*, *31*(1), 32-43. doi: 10.1177/875697280003100106
6. Ibbs, C., & Kwak, Y. (2000). Assessing Project Management Maturity. *Project Management Journal*, *31*(1), 32-43. doi: 10.1177/875697280003100106
7. Kniberg, H., Skarin, M., Poppendieck, M., & Anderson, D. (2010). *Kanban and Scrum*. United States: InfoQ.
8. Meredith, J., Shafer, S., & Mantel, S. *Project management*.
9. Sameen Mirza, M., & Datta, S. (2019). Strengths and Weakness of Traditional and Agile Processes - A Systematic Review. *Journal Of Software*, *14*(5), 209-219. doi: 10.17706/jsw.14.5.209-219
10. Schwalbe, K. *Information technology project management*.
11. Tereso, A., Ribeiro, P., Fernandes, G., Loureiro, I., & Ferreira, M. (2018). Project Management Practices in Private Organizations. *Project Management Journal*, *50*(1), 6-22. doi: 10.1177/8756972818810966
12. Thor Ingason, H., Gestsson, E., & Ingi Jonasson, H. (2013). The Project Kanban Wall: CombiningKanbanand Scrumfor Coordinating Software Projects. *PM World Journal, Vol.II, –August 2013*(Issue VIII). Retrieved from https://pdfs.semanticscholar.org/8cc3/2f3ebdd98ba879f3c7a3efd8176a33243121.pdf