SOFTWARE PROCESS IMPROVEMENT

Submitted by

A. REVANTH (192110492)
B. PRATHAP (192112127)

In partial fulfillment for the completion of the

PROJECT

DEPARTMENT OF PROGRAMMING

SAVEETHA SCHOOL OF ENGINEERING

SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES,
CHENNAI – 602 105

SEP 2022

SUPERVISED BY

Dr. Vijay Bhaskar

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **“software process improvement"** submitted by " A. Revanth (192110492), B. Prathap (192112127)” to Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, is a record of bonafide work carried out by him/her under my guidance. The project fulfills the requirements as per the regulations of this institution and in my appraisal meets the required standards for submission.

Dr. Vijay Bhaskar
Department of Programming

Internal examiner **External Examiner**

ACKNOWLEDGEMENT

I take immense pleasure in expressing my gratitude to our Honourable Chancellor**,** **Dr. N. M. VEERAIYAN** for being our constant source of inspiration.

I sincerely thank our Vice-Chancellor, **Dr. L. JAWAHAR NESAN** for providing the academic ambience to complete the project successfully.

With profound gratitude, respect and pride, I express my sincere thanks to our Director, **Mrs. RAMYA DEEPAK**, whose enthusiasm and dynamism being the driving force behind all our accomplishments.

I take this opportunity to extend our heartiest thanks to our beloved Principal, **Dr. Ramesh** for providing us excellent facilities and unstinting support which were the sustaining factors in carrying out the work successfully.

I owe a deep sense of gratitude to Dr. Vijaya Bhaskar, Department of Programming, for his keen interest on me at every stage of my project.

It is a genuine pleasure to express my deep sense of thanks and gratitude to my mentor Dr. Sungeetha Asst.Prof/Asso.Prof, Department of Nanotechnology. His/her dedication and keen interest had been solely and mainly responsible for completing my work.

Also, I am greatly indebted to my project guide **Dr. Vijay Bhaskar** Department of Biotechnology. His timely advice, meticulous scrutiny and scientific approach have helped me to a very great extent to accomplish this task.

Last, but not least, a very special thanks to my parents and friends for their moral support and care extended during the demanding times.

ABSTRACT

This a port describes the case studies approach applied at a number of Siemens software development organizations to observe the impact of software process improvement methods. In addition, the report provides guidance to software development organizations that want to improve their processes. A set of organization performance measures are defined to help an organization observe its software process improvement over time. An approach is given for selecting software process improvement methods. The report concludes with a description of common implementation problems, and recommendations for organizations to improve their software processes. Once the improvement method is in place, there is also the question of what to do next. It is necessary to determine whether the method was implemented successfully, whether the process is mature enough to consider implementing additional methods, or whether the selected method propriate for use within the current process maturity level and environment.

INTRODUCTION

In 1992 a joint project was initiated between the Software Engineering Institute (SEI) and Siemens to investigate the impact of software process improvement methods. There were two problem questions that motivated the project: * How does one measure the result of software process improvement methods? * How should an organization select methods for software process improvement? A joint SEI/Siemens project on measuring software process improvement methods was initiated to integrate the methods developed at the SEI with actual practices used within Siemens software development organizations. This report is an output of this joint project. The project will identify specific process improvement methods that can be tailored to the current maturity level of the organization that wishes to improve. This project will also provide practical suggestions concerning the implementation and impact of process improvement methods in order to provide the foundation for continuous process improvement.

Need for Process Improvement motivation to improve a software process usually results from a business need such as strong competition, increased profitability, or external regulation. Approaches to improve a software development process, such as those , are often initiated by an assessment of the current practices and maturity. A number of improvement methods are then recommended and implemented. The selection and successful implementation of the improvement methods are dependent on many variables such as the current process maturity, skills base, organization, and business issues such as cost, risk, implementation speed, etc. Measuring the impact and predicting the success of a specific improvement method are difficult. This is often due to environmental variables external to the method such as staff skills, acceptance, training effectiveness, and implementation efficiency. It is necessary to determine whether the method was implemented successfully, whether the process is mature enough to consider implementing additional methods, or whether the selected method is ap propriate for use within the current process maturity level and environment.

OBJECTIVES:

The goals of the SPI in this collaboration were:

* To obtain access to software engineering practices of industrial software development organizations in Siemens companies.

* To obtain methods for conducting case studies for SEI validation efforts.

* To gain access to Siemens process measurement research, standards, and practices. * The selection and successful implementation of the improvement methods are dependent on many variables such as the current process maturity, skills base, organization, and business issues such as cost, risk, implementation speed, etc.

* Measuring the impact and predicting the success of a specific improvement method are difficult.

LITERATURE SURVEY

Mahmood niazi ,king fahd university (2016):

SOFTWARE PROCESS IMPROVEMENT: AROAD TO SUCCESS

Software process improvement (SPI) has received much attention in both academia and industry. SPI aims to improve the effectiveness of the software development process. Several different approaches have been developed for SPI, including the SEI's Capability Maturity Model (CMM), more recently the Capability Maturity Model Integration (CMMI) and ISO's SPICE. Research shows that the effort put into these approaches can assist in producing high quality software. This paper has a two-fold objectives first to review and summarise the empirical evidence thus far on the costs and benefits of SPI approaches; second to establish a relationship between different approaches to SPI and to seek and identify whether these approaches fulfil all the needs for an effective SPI initiative. The aim of this review is to analyse material about SPI approach and to set the Scence for future research in the area of software process improvement.

Emilia Meneds ,Blekinge institute of technology(2013)

Software process improvement:In small and medium web companies

The objective of this paper is to identify existing Software Process Improvement (SPI) models and techniques used by small and medium Web companies. Four papers applied SPI techniques or models to Web companies, and our results showed that none suggested any customized model or technique to measure the SPI of Web companies. The SLR also revealed the characteristics of some small and medium companies and suggested that they have tight budget constraints, tight deadlines and a short term strategy. Finally, our SLR showed that the measures of success for small and medium Web companies included development team and client satisfaction, increase in productivity, compliance with standards and overall operational excellence. The results of this review showed that very few studies have specifically focused on SPI for Web companies, despite the large number of existing Web companies worldwide, and the even larger number of Web applications being currently developed.

Daniel Dias de(2017)

Software Process Improvement and Capability Determination

Software Process Line (SPL) has been claimed as a suitable paradigm for tailoring and reuse of software processes. However, despite its increasing importance, there is still a lack of research that systematically characterizes and analyzes the state of the art of SPL approaches, in particular focusing on how such a paradigm has been used to improve software processes. This paper presents the method followed to perform a systematic literature review on SPL in order to investigate the state of the art of this area, as well as the results of this review focusing especially on how variability is represented.Our results indicate that the software engineering community has increasingly invested effort in this area. However, it is still considered an immature area with many open issues such as the lack of the modeling of well-known process standards and models using SPL concepts and the lack of empirical evaluations.

Michael unterkalmsteiner (2016)

Evaluation and Measurement of SPI

Software Process Improvement (SPI) is a systematic approach to increase the efficiency and effectiveness of a software development organization and to enhance software products. This paper aims to identify and characterize evaluation strategies and measurements used to assess the impact of different SPI initiatives. method the systematic. The selected papers were classified according to SPI initiative, applied evaluation strategies and measurement perspectives. Potential confounding factors interfering with the evaluation of the improvement effort were assessed. RESULTS-Seven distinct evaluation strategies were identified, whereas the most common one, "Pre-Post Comparison", was applied in 49% of the inspected papers. Quality was the most measured attribute (62%), followed by Cost (41%) and Schedule (18%). Looking at measurement perspectives, "Project" represents the majority with 66%. CONCLUSION The evaluation validity of SPI initiatives is challenged by the scarce consideration of potential confounding factors, particularly given that "Pre-Post Comparison" was identified as the most common evaluation strategy, and the inaccurate descriptions of the evaluation context.

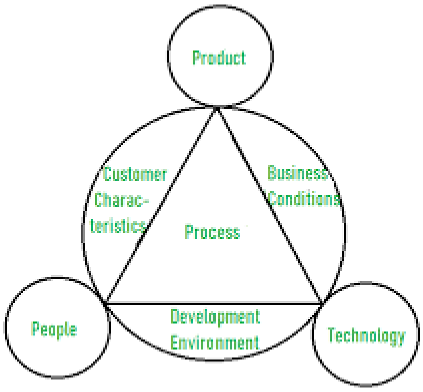
SPI Jacky keung,city university of hong kong(2018)

Agile software development

Software Process Improvement (SPI) maturity models have been developed to assist organizations to enhance software quality. Agile methodologies are used to ensure productivity and quality of a software product. Amongst others they are applied in Small and Medium – sized Enterprises . However, little is known about the combination of Agile methodologies and SPI maturity models regarding SMEs and the results that could emerge, as all the current SPI models are addressed to larger organizations and all these improvement models are difficult to be used by Small and Medium – sized firms. Combinations of these methodologies could lead to improvement in the quality of the software products, better project management methodologies and organized software development framework. The majority of the Agile methodologies and SPI maturity models are addressed to large or very large enterprises. Thus, little research has been conducted for SMEs. The combinations of the Agile methodologies and SPI maturity models are usually performed in experimental stages.

SPI Methodologies:

SPI methodologies defined as part of SPI frameworks are composed of high level activities or phases. They provide guidelines and highlight points to be considered while organizing the SPI activities. For example, ISO/IEC 15504-4 suggests defining an SPI lifecycle and an action plan in Process Improvement Programme Plan. Other than highlighting major headings of the action plan, many decisions are left to the practitioners. Similarly, IDEAL advises the activities to be determined in the “improvement agenda” of the SPI strategic plan. The practitioners on the other end need a practical SPI scheme, specific guidelines on how they will organize and conduct SPI and detailed descriptions of activities.

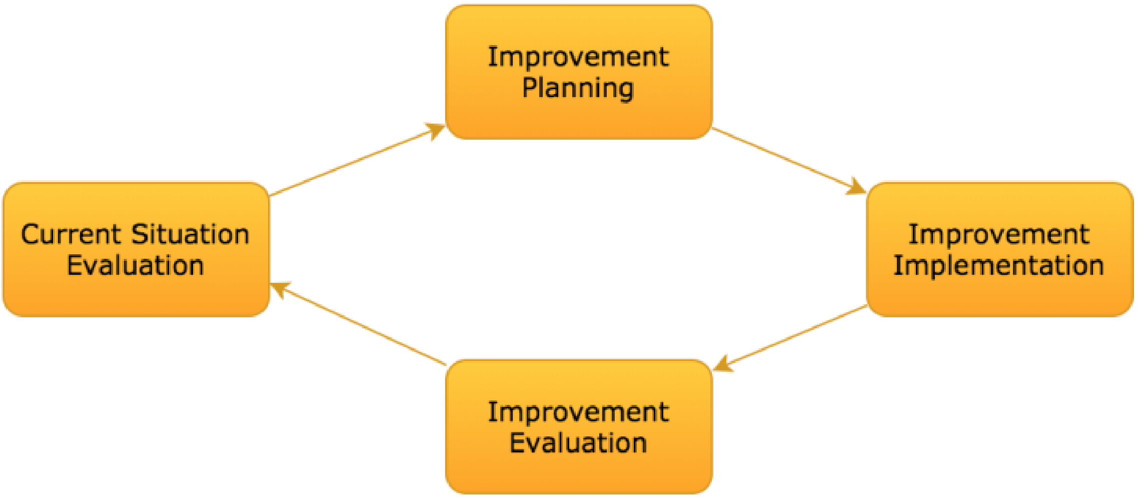


The frameworks for small and medium sized organizations provide more detail on the implementation of SPI, as these organizations require ready to use descriptions. MESOPYME provides a specific implementation approach. With the same perspective of this study, discusses that specific implementation solutions are required to plan SPI activities, and "current methods do not provide guides to elaborate these elements". The solution is using "action packages" to start SPI activities in process areas, proven to be successful in their experiences.

SOFTWARE PROCESS IMPROVEMENT

Software Process Improvement (SPI) methodology is defined as a sequence of tasks, tools, and techniques to plan and implement improvement activities to achieve specific goals such as increasing development speed, achieving higher product quality or reducing costs.

SPI mainly consists of 4 cyclic steps as shown in the figure below, while these steps can be broken down into more steps according to the method and techniques used. While in most cases the process will contain these steps.



A software process improvement is a process of dividing software development work intom smaller, parallel, or sequential steps or sub-processes to improve . It is also known a **software development life cycle** (**SDLC**). The methodology may include the system pre-definition of the specific deliverables and artifacts that are created and completed by a project team all to develop or maintain an application. The main idea of the SDLC has been "to pursue of the development of information systems in a very deliberate, structured and methodical way, requiring each stage of the life cycle—from the inception of the idea to delivery of the final system—to be carried out rigidly and sequentiall within the context of the frame work being applied. The main target of this methodology framework in the 1960s was "to develop large scale functional business systems in an age of large scale business conglomerates. Information systems activities revolved around heavy data processing and number crunching routines.

Current Situation Evaluation

This step is the initial phase of the process and it is mainly to assess the current situation of the software process by eliciting the requirements from the stakeholders, analyzing the current artifacts and deliverables, and identifying the inefficiencies from the software process. The elicitation can be conducted through different techniques. For example, individual interviews, group interview, use-case scenarios, and observations.

The key considerations in this step to identify organization goals and ask the solution-oriented questions. Moreover, identifying the measurement using the GQM (Goal – Question – Metric) technique that will help in measuring the current status and measuring the effectiveness of the improvement process.

Improvement Planning

After analyzing the current situation and the improvement goals, the findings should be categorized and prioritized according to which one is the most important or have the most severity. We should observe what is the new target level of improvements should look like.

Moreover, in this step, the gap between the current level and the target level should be planned in terms of a set of activities to reach that target. These activities should be prioritized with the alignment of the involved stakeholders and the organization goals, for example, if the project is using the CMMI model, the target could be reaching maturity level 4 and the company at level 3, in that case, the plan should be focused on the process areas and their activities which is related to that level of improvement with the alignment of the organization goal.

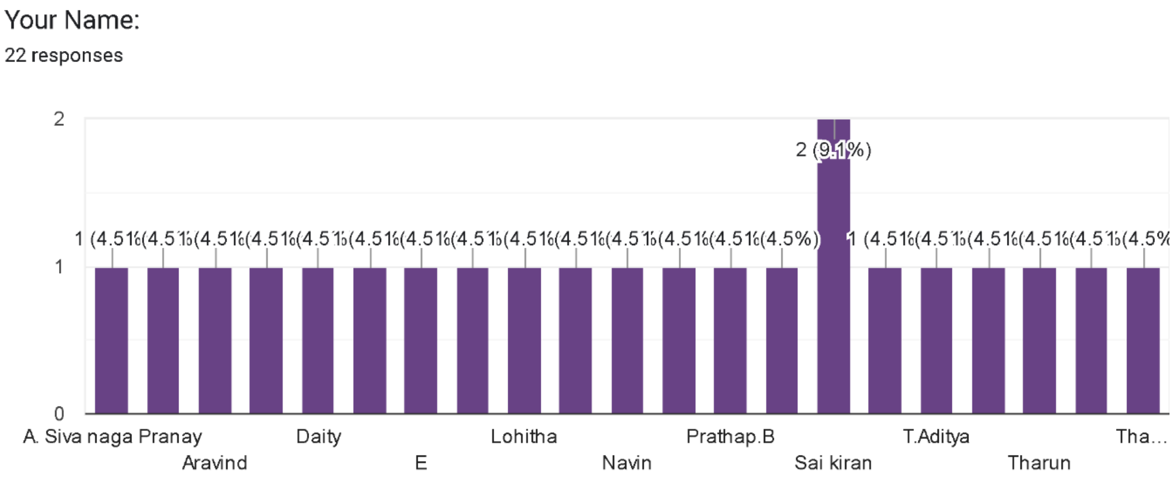
Improvement Implementation

In this step, the planned activities are executed and it puts the improvements into practice and spreads it across the organization, what can be effective at the 2nd, 3rd, and 4th step that planning and implementation could be an iterative way, for example, implementing improvement for improving requirements first, then implementing the reduction for testing process time, and so forth. This iterative way of implementation will help the organization to realize the early benefits from the SPI program early or even adopt the plan if there is no real impact measured from the improvement.

Improvement Evaluation

What is cannot be measured cannot be improved, that’s why in this step, the impact measurement is applied compared with the GQM. The before improvement measures, after the improvement measures, and the target improvement measure. Measurement, in general, permits an organization to compare the rate of actual change against its planned change and allocate resources based on the gaps between actual and expected progress.

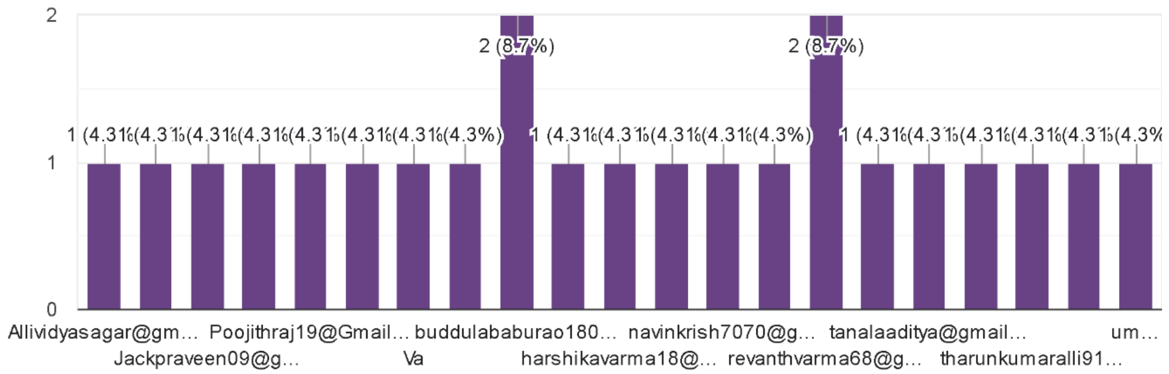
SURVEY


Your Name:

22 responses

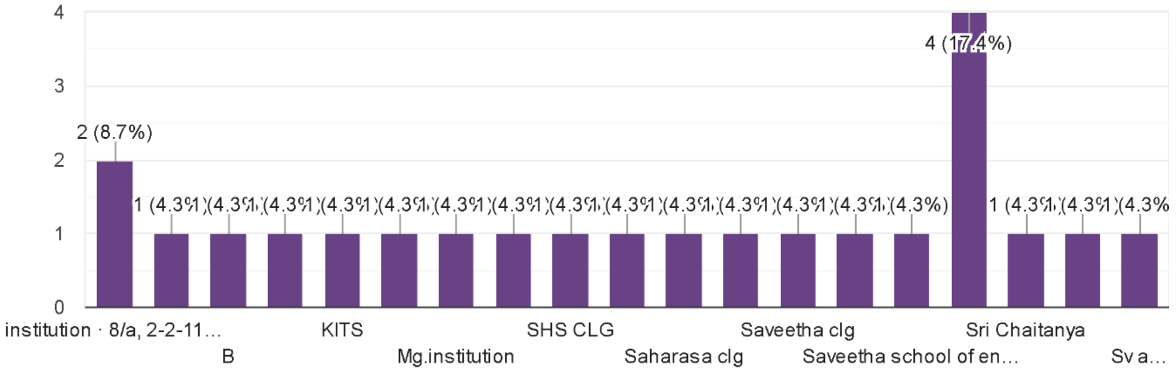
Email ID:

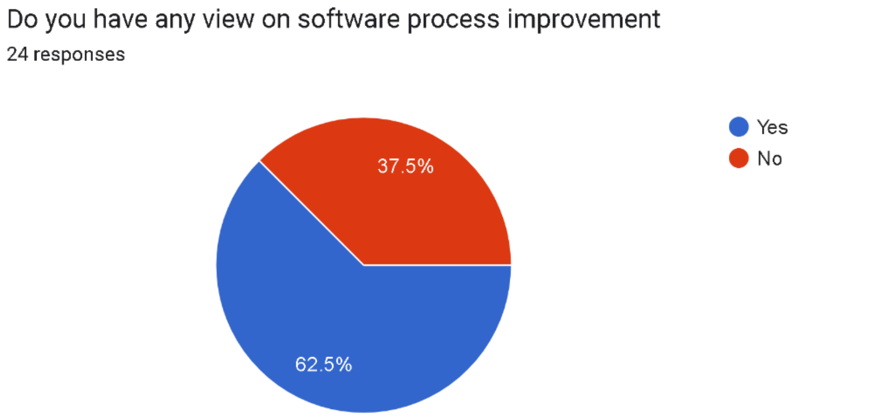
23 responses



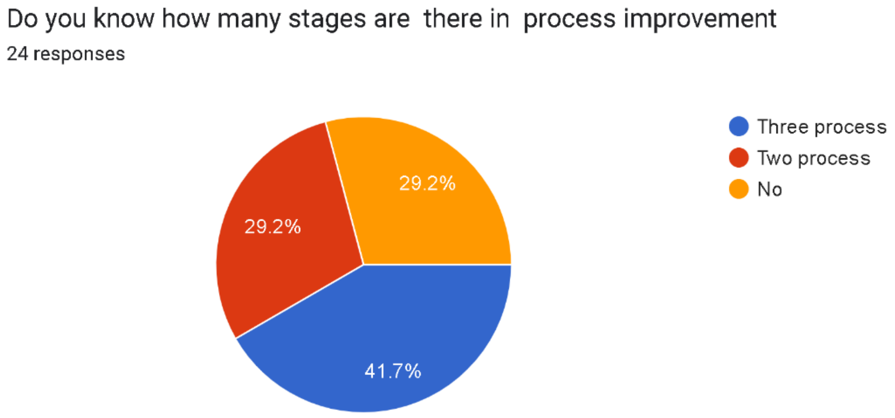
Institution Name:

23 responses

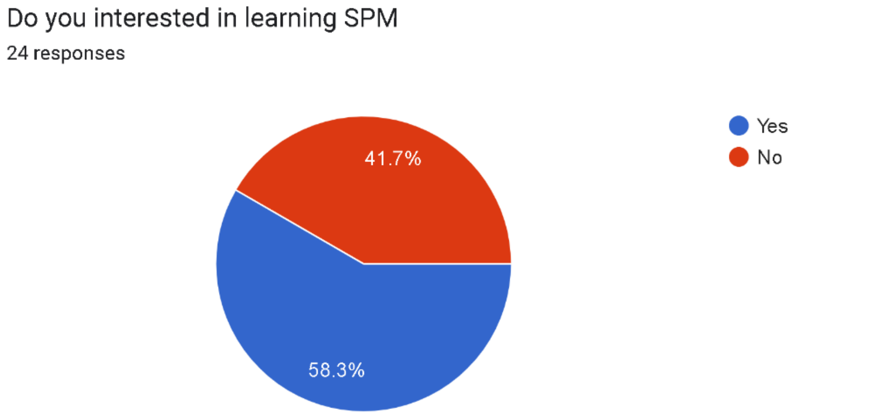



Do you have any view on software process improvement

24 responses


Do you know how many stages are there in process improvement

24 responses


Do you interested in learning SPM

24 responses

CONCLUSION

Changing the results of an organization can be a daunting task. The greatest challenge is knowing what to work on and where to start. By following a few simple steps, you can plan and manage an effective improvement program. The essence of everything we have said is to start small and make all improvements useful. Your job is to create and sustain this environment.

The notion that it is good practice to split a project into smaller, more manageable activities. When developing good software systems, you should focus on the users’ needs and, wherever possible, make use of replaceable and reusable modules – components. The overall software architecture should be constructed around the users’ requirements.

REFERENCES:

1)Barry W. Boehm (2018). Software cost estimation with Cocomo .

2) Curious Lab (2022).journal of multi software process, Retrieved.

3) Humphrey, W. S, (2019), Managing the Software Process, Addison-Wesley, Reading, MA.

4)Humphrey, W. S., July (2017) Snyder, T. R., and Willis, R. R., "Software Process Improvement at Hughes Aircraft," IEEE Software.

5) Deming, W. E,(2020) Out of the Crisis, MIT Center for Advanced Engineering Study, Cambridge, MA.

6)Markus Rerych,(2019) Institut für Gestaltungs- und Wirkungsforschung, TU-Wien. Accessed on line.

7) Lübke, Daniel; van Lessen, Tammo (2018). "Modeling Test Cases in BPMN for Behavior-Driven Development". IEEE Software.

8) Geoffrey Elliott (2014) Global Business Information Technology: an integrated systems approach. Pearson Education.

9) Barry Boehm (2019). Software cost estimation with Cocomo II: Volume 1.

10) Lübke, Daniel; van Lessen, Tammo (2018). "Modeling Test Cases in BPMN for Behavior-Driven Development". IEEE Software.

11) Richard H. Thayer, Barry W. Boehm (2015). Tutorial: software engineering project management. Computer Society Press of the IEEE

12) marketplace.atlassian.com. Retrieved 2022-09-12. *basecamp.com*.

13) Staples, M. and Niaz, M. (2019). Two Case Studies on Small Enterprise Motivation and Readiness for CMMI.