Journal of
Computer Science and Software Development

Research Article

Open Access

Selection of Continuous Integration Tools for Agile Methodology: An Analytic Hierarchy Process (AHP) Approach

Fatma K[1]*, Tokdemir G[2]

[1]Department Software Architectural Design, Tubitak (The Scientific and Technological Research Council of Turkey), Ankara, Turkey

[2]Department Computer Engineering, Cankaya University, Ankara, Turkey.

***Corresponding author:** Fatma K, Department Software Architectural Design, Tubitak (The Scientific and Technological Research Council of Turkey), Ankara, Turkey, Tel: +905077931397, Email: fatmakel6@gmail.com

Received Date: April 11, 2022 **Accepted Date:** May 16, 2022 **Published Date:** May 18, 2022

Citation: Fatma K, Tokdemir G (2022) Selection of Continuous Integration Tools for Agile Methodology: An Analytic Hierarchy Process (AHP) Approach. J Comput Sci Software Dev 1: 1-11

Abstract

In the software world, it is known that continuous integration technology, importance of which is growing rapidly, enhances the efficiency of projects and saves significant time to the user. Institutions that adopt agile software methodology want to include the continuous integration, which is part of agile software, into their business lives. Continuous integration provides automation of various workloads such as version control of the projects, build issues, problems after build, running and reporting of tests. In this way, time to be spent on these tasks and unnecessary workload loss is reduced. Thus, by ensuring more effective use of employees, the efficiency of projects is significantly improved. This technology, which is very popular, brings along a lot of competition in the market. There are a number of continuous integration tools on the market and all of them have different characteristics they offer to get ahead of each other. Choosing the right tool among the alternatives is a challenging task for the companies. Especially for an institution with a closed network such as TÜBİTAK SAGE, it is difficult to choose and make the right choice for a continuous integration tool. The closed network brings the obligation to keep the repositories of the tools to be used in the local network. Therefore, if a tool does not meet the expected requirements and cannot be used, it will cause material and moral damage rather than a benefit. If the wrong tool is selected, then the repositories created for a new tool need to be reinstalled from the beginning. When the continuous integration tool is chosen wrongly, switching to a new technology will cause loss of time and demotivation of the employees. Hence, the expectation of increase in productivity by the usage of such tools will diminish and will bring inefficiency instead. There is no study in which the AHP method is used for continuous integration tools in literature. The difference of the AHP process applied in this study is also a part of the originality of the study. In line with all these, in this study, it is aimed that a continuous integration tool that is suitable for TÜBİTAK SAGE culture and working style is selected. This selection is performed using Analytic Hierarchy Process, which is a multi-criteria decision making method. A systematic process was applied with the consortium of the study group and the expert group. As a result, criteria set consisting of compatibility, flexibility and expandability, functionality and reliability were determined to be used when selecting continuous integration tools. As a result, the priority levels of these criteria were functionality with the highest value with 0.33427, while compatibility with the lowest value with 0.19172. Finally, four different continuous integration tools selected according to the corporate culture and requirements were evaluated with the AHP method. As a result, among the alternative, alternative 1(Jenkins) was chosen first with rate of 37,14%, followed by Alternative 2(TeamCity) with rate of 28,86%.

Keywords: Continuous Integration, Agile Methodology, Continuous Integration tools, Multi-Criteria Decision Making, AHP.

Introduction

In today's world where everything is racing with time, time and speed factors have an important place in the software Industry. Henry Ford emphasized the Importance of time with the following words: "It has been my observation that most people get ahead during the time that others waste"[1]. For this reason, companies in the software sector compete with time and want the products to come out quickly. The Impatience of customers and the prolongation of the schedule of projects put serious pressure on software developers and managers. One of the external qualities that stakeholders expect from software developers Is speed [2,3].

Agile methods have started to be adopted in the name of speed and dynamism in Software Engineering. One of the 12 principles of agile software methodology Is “Our Highest Priority Is tosatisfy the Customer through Early and Con-tenuous Delivery of Valuable Software” [4]. Agile software development methodology requires fastness. This methodology determines the points to be considered while being fast and shortens your project time. In this way, it enables the tests to be carried out while providing faster and easier communication. [5,6]. In Institutions that have adopted the agile software methodology, there are cases of teamwork, weekly maybe daily code additions to the project, and the control of the added codes due to the continuous development. ItIs an Important element that these situations do not affect the working system.

In projects that are developed by software developer teams, many team members make changes on the code; write separate modules, which are integrated to the project. These changes and additions affect the whole system. The smooth running of the system despite all the additions and changes is an important factor. Even the slightest mistake that occurs because of changes and additions causes the project process to be prolonged. Project team members spend most of their time looking for the module that the error originated from rather than developing the project and making the project work. Detecting the problematic parts of the code is of great Importance in terms of speed in large project. With all these in mind, automating the software process of the project will ensure that the problem Is solved In an efficient, safe and functional manner [7].

Software companies are in a fierce race to Improve and make their development processes more efficient. Companies have to constantly adapt to new developments to gain a competitive advantage. This need for rapid development and adaptation has led to the DevOps culture and philosophy, which Is a more holistic approach with continuous Integration (CI), continuous delivery (CD) [8]. DevOps emerged to eliminate the bottleneck preventing development teams from delivering to operations faster and more frequently. DevOps approach requires a strong and continuous connection between development and operations teams [9]. Continuous Integration is atthe heart of the entire DevOps lifecycle.

Continuous Integration Is defined as “software development practice where members of a team Integrate their work frequently, usually each person Integrates at least daily - leading to multiple Integrations per day. Each Integration Is verified by an automated build (Including test) todetect Integration errors as quickly as possible” [10].

Continuous Integration Is one of today’s technologies that ensure both quality work and minimization of time loss In large software projects [11]. In this way, the risk of generating errors in large projects, where team members develop various software modules, can be minimized. When an error occurs, the module in which the error occurs Is communicated to the authorized person, eliminating the search for the origin of the error and providing the developer more time and speed gain, and thus more focus on writing quality code [12].

Continuous Integration is the name given to the automatic processing of the written code and the developed module while being Included in the project. Continuous Integration is a system that checks that the entire system is operational after every change made to the code. It is also the method used to determine if the change caused breaks in some parts of the system. All actions such as downloading dependencies from repos, including them in the project, passing the developed code snippet through tests (unit, Integration, etc. ...), Informing the developer or the responsible person according to possible situations (push notification, sms, mail, etc.), packaging the work, usually takes place in our lives as part of this concept [13,14].

Continuous Integration Is the method used to determine if the entire system is operational after each change in the code, and that the change does not cause breaks in some parts of the system. Unit tests are needed to detect breakages [11]. These tests are run automatically after a newbuild is prepared because of the change. Since the change is a part of the new structure, errors in the tests mean that the change made broke the system. All programmers are informed about this situation and It Is ensured that the error is eliminated as soon as possible and the tests always give positive results. With continuous Integration, because of the work done on the code by theprogrammers, working version is created always.

In this context, there are many CI tools with diverse features. In order to avoid problems in the future project Integration and process, Institutions should determine the criteria that they expect from continuous Integration. They should try to choose the right tool that meets these criteria in the most optimal way.

Many criteria should be taken into consideration in the selection of CI devices. We can list these criteria as follows; functionality, compatibility, reliability, long life availability, flexibility, extensibility, be open source or commercial. The abundance of criteria and alternatives led us to mute-scale selection methods [15]. In this study, it was decided to use Analytical Hierarchy Process, one of the multiscale selection methods. AHP Is one of the Multi Criteria Decision Making methods put forward by Thomas L. Saaty that helps the decision maker [16]. AHP has been used in many areas and in the software field in the decision-making process in multi-criteria situations, and it has been effective in making an optimal decision for the user.

Background and Related Works

Nowadays, software companies understand the value of continuous integration, and because continuous integration increases the software speed, hence the quality and reduces the cost, companies have turned to this area. The high demand has led to the proliferation of continuous integration tools and increased competition. The fact that there are many tools and the competition between them made the tools differ in many criteria such as their features, price, functionality, flexibility, expandability, compatibility, usability, safety and longevity. Some of the tools are open source and some are paid. Just because the product is expensive does not mean it is better than a low cost or open source tool. Generally, all of these tools try to automate the softwarecreation process. Despite this fact, Continuous Integration tools have their own strengths and weaknesses, so choosing the right tool for a job is really important.

Wrong choice can reduce overall flexibility; it can increase the time required to perform ordinary actions or cause some problems in the Continuous Integration process. A decision that is not made right from the beginning will increase the whole adjustment process, employees' adaptation to the tool and the integration process of projects.

In line with the increasing demand, there has been competition in the market in this areaand a lot of continuous integration tools have emerged. Each strives to stay in the market with extra features to compete with others. At this stage, it is of great importance for the institutions to choose the right tools [15].

In this direction, features that should be evaluated in continuous integration tools are faster builds, integration, quick and easy setup, container support, testing with multiple runtimes,versions, and environments, integrated code coverage and test results visualization can be listed as flexible infrastructure options [17]. In addition to these, it will be very difficult to choose a tool thatmeets many criteria, such as functionality, compatibility, reliability, long-term availability, flexibility, extensibility, availability, open source or commercial, and can accommodate them among all options [15,18,19]. On the other hand, a wrong decision will be slow and weary. It is important that the CI tool is compatible with the build automation tool used in the projects, an add-on to the test automation tool and is flexible and extensible for future features [20,18,21]. In the event of a malfunction in these features, the tool used to accelerate the software process will have the opposite effect, causing the process to slow down, delay, or make no progress, and will not serve its real purpose. If there is a problem with at least one of these, the tool selection process willstart over and this will be weary. For the user, getting used to something new, breaking the order isa challenge in itself, and the problems that come on top of it can cause the breaking of the resistance. This problem will be even more cumbersome if a closed network is used as in the institution where this study will be conducted. The reason for this is that since the intranet is used, you have to download the plugins that the tool will use and set up your own repo, and this is not aneasy process. On top of that, preparing everything from scratch in case of a problem, establishing anew repo will be financially and morally wearing. Although not all CI devices have open source code, it will be a wasted investment in the direction of wrong selection. Considering all these, it is thought that choosing the right tool will provide financial and moral benefits for the institution andthe user in addition to its contribution to the projects. Polkhovskiy, compared CI tools in his study and underlined the importance of correct selection [15].

In this study, Analytical Hierarchy Process (AHP) which is, one of the multiscale selection methods, is chosen for the selection of right CI tool for quality of software process, correct management of time and increasing efficiency in projects.

The Analytical Hierarchy Process (AHP) is a powerful and flexible multi-criteria decision- making method introduced by Thomas L. Saaty that is applied to solve unstructured problems in a variety of decision-making situations, such as complex intensive decisions, including simple personal decisions [22]. One of the biggest advantages of this method is that it enables analytical evaluation of criteria without numerical values by means of comparison methods. AHP's application areas are quite wide. AHP is used for various situations such as location establishment location applications, supplier selection, recruitment, selection of continuous improvement projects, hardware and software selection, Investment decision, determination of the most appropriate strategy, performance evaluation, and selection of product alternatives for consumers. The study of Thomas L. Saaty shows that it has been observed that the best decisions are made with AHP, which is used in many fields of health, military, education and many other sectors [23]. Yılmaz et al mentioned the use and importance of AHP in the field of health where vital decisions are taken [24]. In the field of software, Laplante emphasized the selection process of the software project management tool with AHP in her studies [22]. Li et al. proposed a framework for selecting software reliability criteria based on AHP theory and expert opinion, aiming to solve the problem of how to scientifically select appropriate software reliability criteria for software reliability engineering at each development stage [25]. In addition, in the software field, we can seethat the study of Tekin uses the AHP method in the selection of measurement metrics in the software development process [26]. It can be seen in Ming-Chyuan Lin et al. study that AHP is alsoused in the customer-oriented product design process [27].

Methodology

First of all, the problem of choosing the appropriate continuous integration tool wasdetermined. After that, a questionnaire was applied to the study group. Then, AHP method was applied by the contribution of expert group. The priorities of the criteria and alternatives were evaluated. Finally, the alternative with the highest priority was chosen as the final decision. Figure 1 shows all the steps of the case study process. Considering all of these, the research question of this study includes: Which is the continuous integration tool that meets the criteria and features suitable for our institution?

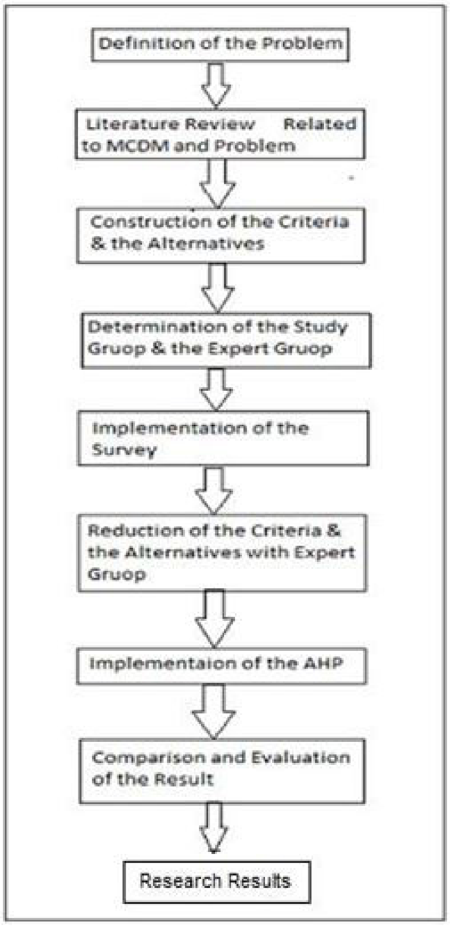
Definition of the Problem
Literature Review Related to MCDM and Problem
Construction of the Criteria & the Alternatives
Determination of the Study Gruop & the Expert Gruop
Implementation of the Survey
Reduction of the Criteria & the Alternatives with Expert Gruop
Implementaion of the AHP
Comparison and Evaluation of the Result
Research Results

Figure 1: Step of the Case Study

Analytic Hierarchy Process (AHP)

The Analytical Hierarchy Process (AHP) was first introduced by the Myers and Alpert duo in 1968, and in 1977, it was developed as a model by Saaty and made usable in the solution of decision making problems. AHP is a decision-making method used in the solution of complex problems involving multiple criteria. The most important feature of AHP is that it can include bothobjective and subjective thoughts of the decision maker in the decision process [28].

A decision-making problem solved with AHP has to go through some stages. These stages are explained below with their formulas.

Step 1: Definition of Decision Making Problem

In the first step, the decision-making problem should be defined clearly. The suitability foranalysis with AHP should be investigated. Defining the decision-making problem consists of two stages. In the first stage, decision points are determined. In the second stage, criteria affecting decision points are determined.

Step 2: Creating the hierarchy

This step is the process of decomposing a decision problem into sub-problems in a hierarchical order that will make it easier to grasp and evaluate of problem. The hierarchy structure of AHP is shown in Figure 2.

Step 3: Creating a Comparison Matrix between Criteria

The comparison matrix between criteria is an nxn square matrix. The matrix components on thediagonal of this matrix take the value 1. The comparison matrix is shown below.

$$\text{Alternatives} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

Components on the diagonal of the comparison matrix, i.e. in the case of i = j, take the value 1. Because in this case, the relevant factor is compared with itself. Comparisons of criteria are made one-to-one and mutually in line with their relative importance values. Score scale should be used which is proposed at Table 1 by Saaty when these matrices are forming [29].

Comparisons are made for all values above the diagonal of 1 in the comparison matrix. Naturally,it will be sufficient to use the formula below for the components below the diagonal

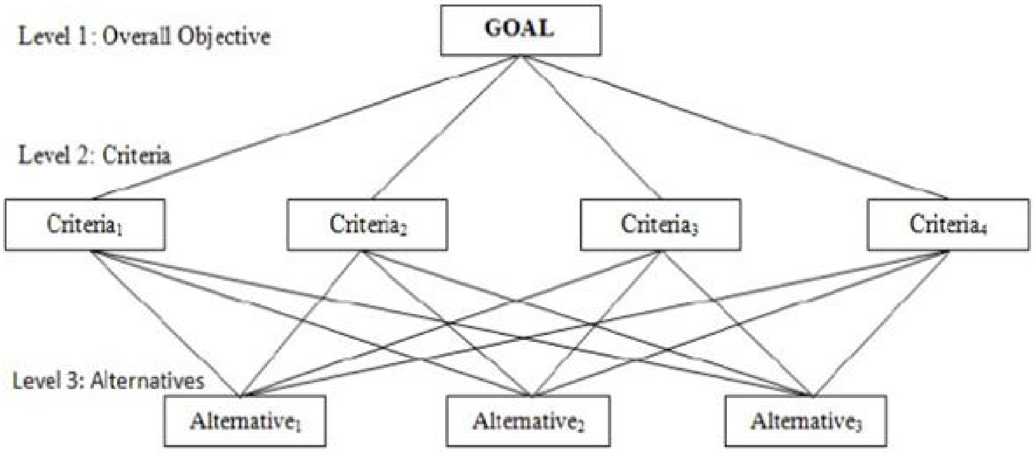

Figure 2: General Hierarcy Structure o AHP

Table 1: Scoring Scale of AHP

Intensity of Importance	Definition	Explanation
1	Equal Importance	Requirements I and j are equal value
3	Moderate Importance	Requirements I has slightly highervalue than j
5	Strong Importance	Requirements I has strongly highervalue than j
7	Very Strong Importance	Requirements I has very stronglyhigher value than j
9	Extreme Importance	Requirements I has very an absolutely higher value than j
2,4,6,8	Intermediate Values	These are Intermediate scales between two adjacent judgments

Step 4: Determining the Importance value of the Criteria

The comparison matrix shows the importance levels of the criteria relative to each other within a certain logic. Column vectors forming the comparison matrix are used to determine percentage significance distributions. B column vector with n grain and n components is constructed.

The formula below is used to calculate the B column vectors.

$$b_{ij} = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}}$$

When the steps described above are repeated for all criteria, B column vectors will be obtained as many as the number of factors. When n grain B column vectors are combined in a matrix format, the C matrix shown below will be created.

Using the C matrix, the arithmetic average of the row components forming the C matrix is obtained as shown in the formula and the column vector W called the Priority Vector is obtained. Percentage of importance level of criteria can be presented by

Step 5: Measuring Consistency in Criterion Benchmarks

The realism of the results will depend on the consistency of the decision maker in the one-to-one comparison between the criteria. AHP proposes a process for measuring consistency in these comparisons. For the consistency index, λ must be calculated. For the calculation of λ, the D column vector is obtained from the matrix product of the comparison matrix A and the W priority vector.

After calculating a, the Consistency Indicator (CI) can be calculated using the formula below.

In the last stage, CI is divided by the standard correction value called Random Indicator (RI) andCR is obtained with the following formula.

Table 2: Random Consistency Index Table

N	RI
1	0
2	0
3	0,58
4	0,90
5	1,12
6	1,24

If the calculated CR value is less than 0.10, it shows that the comparisons made by the decision maker are consistent. A CR value greater than 0.10 indicates either a calculation error in AHP or the inconsistency of the decision maker in comparisons

Platform

Super Decisions software version 2.10.0 was used to implement the AHP method in this study. The software provides to calculate weights and compare pairwise alternatives and criteria

Study Group

The study group consists of project software developers in software department during the software development process. This group, which has a corporate culture, has been chosen by people with a sense of responsibility and sufficient experience in the field of software.

Expert Group

The expert group members consist of software developers with titles such as team leader, unit manager, coordinator and / or chief scientist. Each of them has at least 5 years of working experience in software development. This group is responsible for any decision made in the fieldof software development process.

Implementation of Survey

A meeting was held in the software unit with the participation of the study group and the expert group. At this meeting, participants were informed about Continuous Integration. Continuous integration, what it is, its advantages, disadvantages, purpose of use, has been mentioned extensively. In addition, it was announced that in this survey, where users will be surveyed, information about their work experience, areas of experience, graduation departments and work units will be collected. In addition, it was announced that the users will be surveyed and information about their work experience, areas of experience, graduation departments and work units will be collected in this survey.

The survey has been prepared on Google forms. While preparing the survey, the user was reminded of the general features of the continuous integration tools in a table. In addition, an explanation has been added to the options offered for selection. The analytical process format of the collected data was converted into graphic formats with the help of Google Sheets and Microsoft Excel. In this way, the results were evaluated in detail by presenting them to the expert group. According to the results of the survey, the criteria and alternatives were reduced by the expert group.

Findings and Discussions

40 people, including 25 electrical and electronics engineers and 16 computer engineers,participated in the survey.

The features that the participants expect from the tool are as shown in Figure 4**.**

As a result of these expectations, alternative CI tools were determined by the Expert Group.

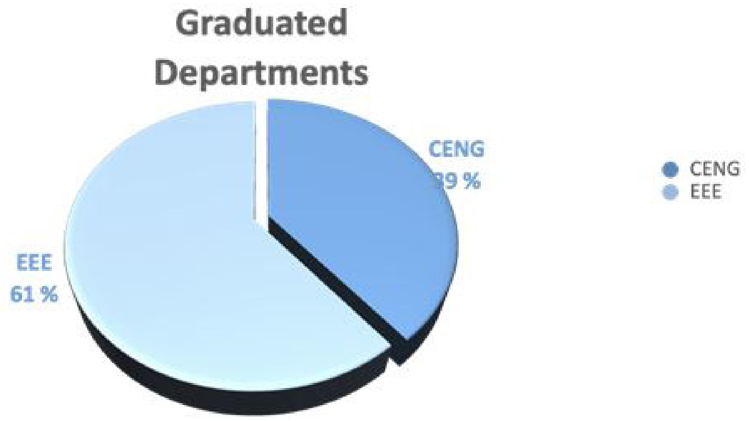

Figure 3: Study Gruop Graduated Department

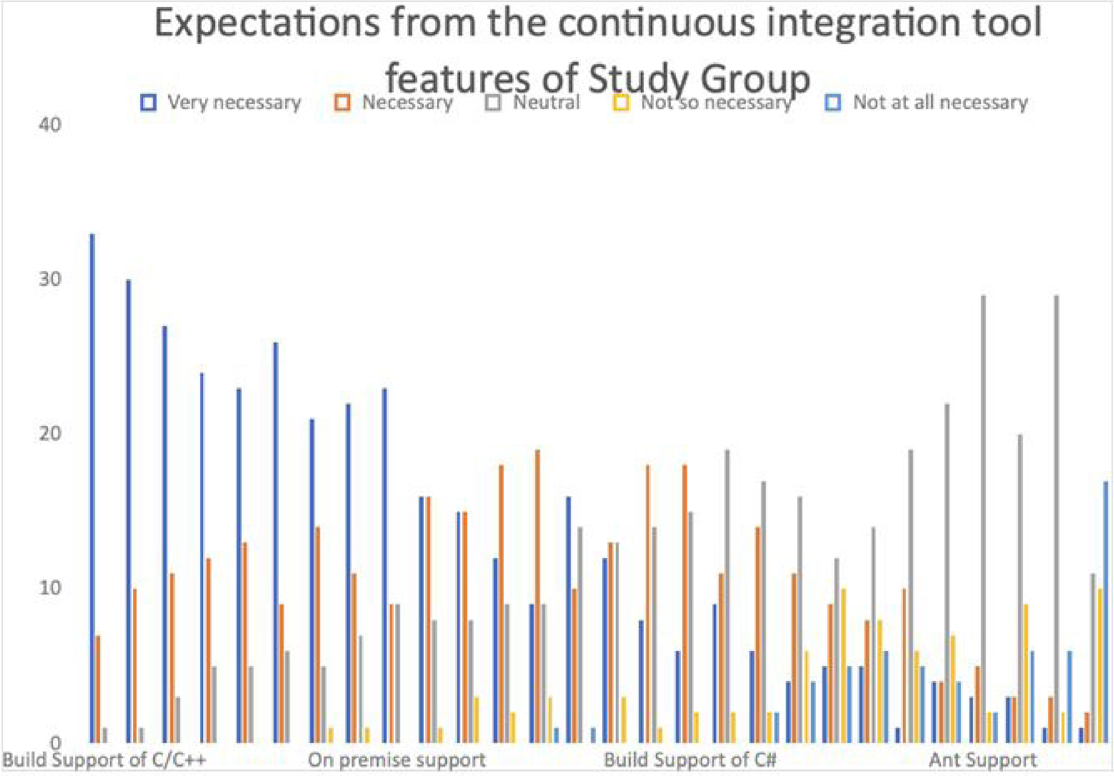

Figure 4: Expectations from the continuous Integration tool features of Study Group

Table 3: CI Alternatives

CI Tool Alternatives
Alternative 1
Alternative 2
Alternative 3
Alternative 4

The criteria for these CI tools to be evaluated were chosen by the expert group

Table 4: Criteria

The Criteria
Functionality
FlexibiIity & Expandability
Compatibility
Reliability

AHP hierarchical structure, which is the selection of Continuous Integration tool suitable for our corporate culture, is included. This hierarchical structure is formed between the specified criteria and alternatives.

As a result of the decision matrix created with the Super Decision program, the weights of thecriteria were obtained. The priorities for the criteria set for the continuous integration tool selection problem is in Figure 6. According to this figure, the Functionality criterion has the maximumweighty 0.33427. The compatibility criterions followed by the 0.19172.

All these operations and the creation of the decision matrix were done with the Super Decision application. As a result of the calculations, the CI tool alternative 1 have highest value with 0.371432 and CI tool alternative 4 have lowest value with 0.1487562.

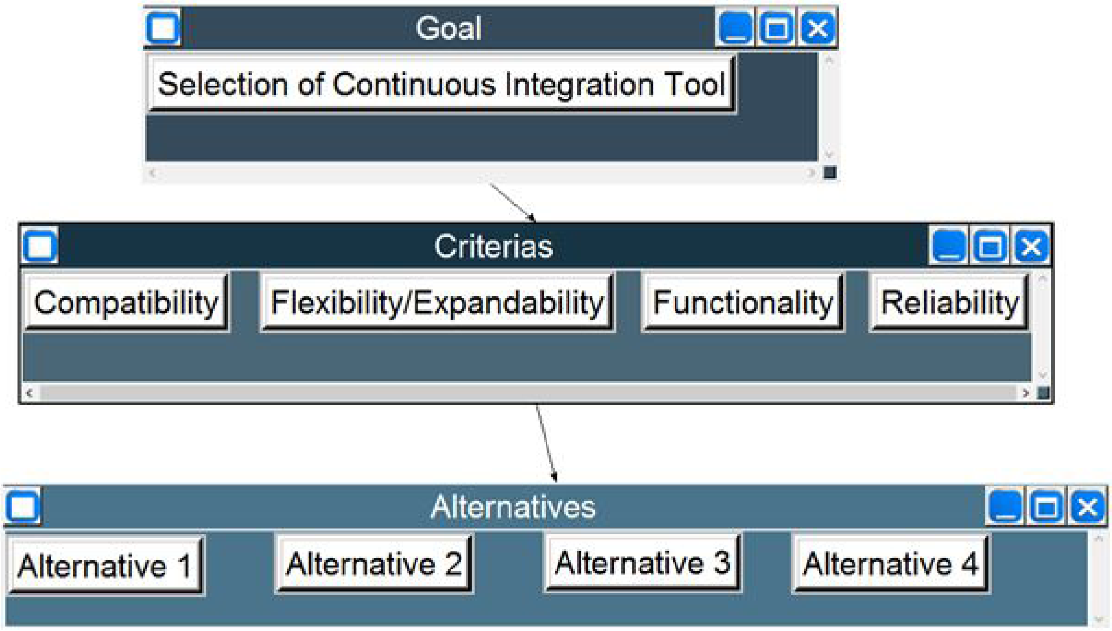

Figure 5: The AHP hierarchy structure of the component selection

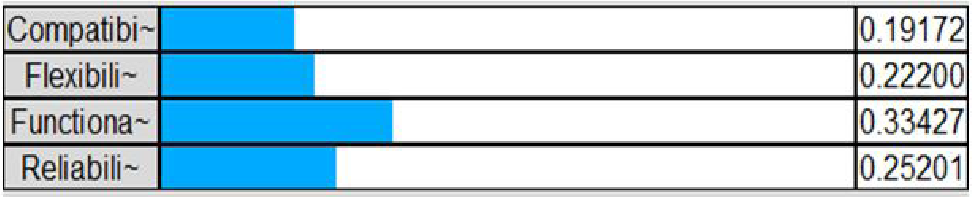

Figure 6: The mean of criteria weights

Table 5: Mean Priorities of the Alternatives

Mean Priorities of the Alternatives	Normalized By Cluster
Alternative 1	0,371432
Alternative 2	0,288618
Alternative 3	0,1911942
Alternative 4	0,1487562

With the AHP method applied, it was determined which continuous integration tool is the most suitable for our institution's culture and working style. In the selection method used, a result was obtained with the participation of 5 experts in the field. Firstly, each member of the expert group was asked independently from each other to evaluate the criteria determined by pairwise comparison. As a result of this evaluation, average criterion weight values were obtained. Technical informations, market reviews and a demo of all continuous integration alternatives were given to the expert group for their review. In the light of this information, they were asked to makea pairwise comparison of alternatives among themselves. As a result of these evaluations, the priorities values as a final result was obtained by using average criteria weight values.

Conclusion and Recommendations

In this study, the appropriate continuous integration tool selection is made using a systematic process and multiscale selection method. The importance of continuous integration, one of today's technologies in projects, has been mentioned. In addition, the place of continuous integration in agile software methodology was touched on. The importance of continuous integration has been referred about choosing the right continuous integration tool and the disadvantages when making the wrong choice. In line with all these, the most suitable continuous integration tool for the institution culture, the most suitable tool was selected as a result of the evaluation of the expert group with the requests determined with the participation of the study group.

In this case study using AHP, the graphics of the survey results of the study group were evaluated by the expert group. As a result of this evaluation, functionality was considered by the expert group to be the most important criterion. Considering the importance of the projects, the reliability criterion has not been ignored. In addition, flexibility and expandability criteria were also preferred in line with the demand for different additions in the graphic results. It follows witha very close value in compliance criteria which was selected considering the priorities of the units.

As a result of evaluating the alternatives, the expert group determined Alternative 1 first with a rate of 37.14%. During evaluating the alternatives, it was ensured that the Experts were not affected by each other. In addition, sufficient time was given to appraise the Alternatives that was given detailed information about. The expert group evaluated the alternatives among themselves inthe light of this information and in line with the wishes of the working group. Comparing the alternatives with each other using the pairwise comparison method prevented them from focusing on a single alternative. The fact that the average of the evaluations of five experts was not selected for a single person made the study more objective.

This case study has proven that AHP method can be used efficiently in the selection of Continuous Integration tool in the software field. This study has been an exemplary study showing that AHP methodology can be used in a situation that requires multiple decision-making methods in software field, if needed in the future. With the right tool selection, productivity has increased and time has been saved. To sum up, this study can be supported with fuzzy AHP method, and hybrid method work can be achieved by using ANP and TOPSIS methods.

Conflict of Interest

We confirm that there are no conflicts of interest to be declared for this publication

Acknowledgments

We should like to thank the Senior Editors and Associate Editors of the journal as well as the following reviewers who have generously given up valuable time. The success of the journal depends upon their care and competence. Their conscientiousness is much appreciated.

References

1. It has been my observation that most people get ahead during the ... - Quote. 2020.

2. Jones C (2012) Software Quality Metrics: Three Harmful Metrics and Two Helpful Metrics," Namcook Anal. LLC.

3. Jones C and Bonsignour O (2012) The Economics of Software Quality.

4. Al-Saqqa S, Sawalha S, and Abdelnabi H (2020) Agile software development: Methodologies and trends. Int J Interact Mob Technol 14: 246-270.

5. Sunner D (2016) Agile: Adapting to need of the hour. Int Conf Appl Theor Comput Commun Technol. 130-135.

6. Gaurav K and Pradeep BK (2012) Impact of Agile Methodology on Software Development Process. Int J Comput Technol Electron Eng 2: 46-50.

7. Hemon A, Lyonnet B, Rowe F, and Fitzgerald B (2020) From Agile to DevOps: Smart Skills and Collaboration. Inf Syst Front 22: 927-945.

8. Arachchi SAIBS and Perera I (2018) Continuous integration and continuous delivery pipeline automation for agile software project management. Int Multidiscip Moratuwa Eng Res Conf 156-161.

9. Fitzgerald B and Stol KJ (2017) Continuous software engineering: A roadmap and agenda. J Syst Softw 123: 176-189.

10. Continuous Integration (2020)

11. Shahin M, Ali Babar M, and Zhu L (2017) Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices. IEEE Access 5: 3909-3943.

12. Khan A, Chawla H, Kathuria H, Chawla H, and Kathuria H (2017) Practicing Continuous Integration and Continuous Delivery on AWS Accelerating Software Delivery with DevOps. Researchgate 1-10.

13. Seth N and Khare R (2016) ACI (automated Continuous Integration) using Jenkins: Key for successful embedded Software development. Int Conf Recent Adv Eng Comput Sci RAECS 2015, 2016.

14. Bobrovskis S and Jurenoks A (2018) A survey of continuous integration, continuous delivery and continuous deployment. CEUR Workshop Proc 2218: 314-322.

15. Polkhovskiy D (2016) Denis polkhovskiy comparison between continuous integration tools. Indian J Sci Technol 7.

16. White GP (1987) The implementation of management science in higher education administration. Omega 15: 4283-290.

17. 10 Things to Consider While Choosing a CI Platform - DZone DevOps (2020)

18. Makam V and Express A (2020) Continuous Integration on Cloud Versus on Premise: A Review of Integration Continuous Integration on Cloud Versus on Premise: A Review of Integration Tools.

19. Larrea VGV, Joubert W, and Fuson C (2015) Use of Continuous Integration Tools for Application Performance Monitoring. 1-12.

20. Arachchi SAIBS and Perera I (2018) Continuous integration and continuous delivery pipeline automation for agile software project management. MERCon 2018 - 4th Int. Multidiscip. Moratuwa Eng Res Conf. 156-161.

21. Owen Rogers R (2004) Scaling continuous integration," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 3092: 68-76.

22. Ahmad N and Laplante PA (2006) Software project management tools: Making a practical decision using AHP. Proc. 30th Annu. IEEE/NASA Softw Eng Work 30: 76-82

23. Saaty TL (2002) Decision making with the Analytic Hierarchy Process. Sci Iran 9: 215-229.

24. Yilmaz BO (2020) Multi-Criteria Decision Making (MCDM) Applications in Military Healthcare Field 2: 149-181.

25. Li H, Lu M, and Li Q (2006) Software reliability metrics selecting method based on analytic hierarchy process. Proc. - Int Conf Qual Softw 337-344.

26. Tekin MN (2019) Selection of The Software Development Process Measurement Component On Scrum Software Development: An Analytic Hierarchy Process ApproacH.

27. Lin MC, Wang CC, Chen MS, and Chang CA (2008) Using AHP and TOPSIS approaches in customer- driven product design process. Comput Ind 59: 17-31.

28. Dağdeviren M (2008) Decision making in equipment selection: An integrated approach with AHP and PROMETHEE," J. Intell. Manuf 19: 397-406.

29. Saaty TL (1998) WHAT IS THE ANALYTIC HIERARCHY PROCESS ? Introduction In our everyday life, we must constantly make choices concerning what tasks to do or not to do , when to do them , and whether to do them at all . Many problems such as buying the most cost effective.