The Academic
International Journal of Multidisciplinary Research

An Online Peer Reviewed / Refereed Journal
Volume 3 | Issue 11 | November 2025
ISSN: 2583-973X (Online)
Website: www.theacademic.in

Employing Agile Methodology for Geographic Information Systems

Dr. Navin Chandra[1], Manisha Baxla[2], Dr.Manish Kumar[3]

[1]Assistant Professor
University Department of Physics
Vinoba Bhave University, Hazaribag, Jharkhand-825301
Email: Navin.c.0001@gmail.com

[2]Assistant Professor
University Department of Computer Applications
Vinoba Bhave University, Hazaribag, Jharkhand-825301
Email: Manishabaxla07@gmail.com

[3]Assistant Professor
University Department of Computer Applications
Vinoba Bhave University, Hazaribag, Jharkhand-825301
Email: Manish14.kk@gmail.com

DOI : https://doi.org/10.5281/zenodo.17953532

ARTICLE DETAILS

Research Paper

Accepted: 28-11-2025

Published: 10-12-2025

Keywords:

Geographic Information Systems, Agile, Scrum, extreme programming, dynamic systems development (DSDM), Kanban

ABSTRACT

GIS geographic information systems have ended significant progress due to improved hardware performance and software development technology. GIS is very handy for planning data analysis and making decisions. GIS is clearly a computer application that allows you to create, accumulate, visualize and analyze geographic information. 1. GIS deals with many in turn format and requires an appropriate expansion approach to achieve its objectives. GIS teams must choose the most appropriate development method based on the nature of the project team's assets and the customer's needs. This article focuses on some of the agile systems development methodologies. Extreme programming of dynamic scrum systems. dsdm and kanban development methodology and describes in detail the phases of these methods this article evaluates GIS development techniques evaluates the usefulness of the most used GIS development techniques and analyzes and discusses their main advantages and disadvantages.

I. INTRODUCTION

The systems development life cycle (SDLC) is a speculative model used in scheme management that explain all stages of information systems scheme development, from preliminary feasibility studies to maintenance of the whole software system. One of the generally important tasks faced by system developers is choosing the most appropriate methodology for system development. This is because the methodology you choose has a noteworthy impact on numerous prospect of the development process, including cost, time, budget, and resources required. A variety of SDLC methodologies have been developed to handle and implement related enlargement process at a consistent scale.

Methodologies have evolved from familiar SDLC techniques such as the waterfall model to the modern models known nowadays like agile software development. One of the newest systems development methodologies that have proven positive in many IT projects across altered disciplines is the Agile methodology. It is one of the non-traditional system improvement methodologies that is calculated as an adaptive methodology rather than a projected one. It does not accentuate questionable detailed planning at an early stage of development. Other important characteristics of agile methods include proactively responding to changes, even in the finishing stages of development, and keenly concerning users during the development process. Agile methods consist of extreme programming (XP), Scrum, Crystal, dynamic systems development methodology (DSDM), resource-driven development, and adaptive programming. We'll begin our research by taking a closer look at the two most commonly used agile methodologies, XP and Scrum, and then look at other agile methodologies.

Agile methodologies include Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Methodology (DSDM), Feature-Driven Development, and Adaptive Programming. We'll start our investigation by taking a closer look at the two most widely used agile methodologies, XP and Scrum, and then look at other agile methodologies. Extreme Programming (XP) is one of the most widely used agile methodologies and takes an iterative approach to the development cycle. Although XP still has discrete steps of requirements analysis, design, implementation, and testing, similar to traditional methodologies such as waterfall, XP views these activities in an interconnected and continuous manner. Each activity includes a specific set of practices that help system developers respond to and embrace the changes that are inevitable at any stage of the system development lifecycle. Besides Extreme Programming, there is the Scrum methodology, which is proven by leading companies not only in software development but also in other areas such as finance and research. Scrum is a management

methodology that enhances an iterative and step-by-step approach and paves the way to managing and controlling software development.

Another methodology is Dynamic Systems Development Methodology (DSDM), which is one of the newest agile methodologies. This is the basis not only for IT projects, but also for project management and solution delivery in general. Like any other agile method, it has its benefits. Kanban is an agile method and is not necessarily iterative. It supports valuable team collaboration in software development projects and enables more effective teamwork. The iterations are short, and each iteration has a clear beginning and end. Kanban improves workflow and minimizes time cycles. Geographic information systems (GIS) are defined as “computer applications that can create, store, manipulate, visualize, and analyze geographic information” [1]. These are considered powerful tools for data analysis and decision making. The use of GIS has increased significantly in recent years. GIS is a powerful computing tool that can collect, store, manipulate, analyze, and display spatially referenced information [1].

GIS's primary constituents are persons, facts, software, and hardware. Geographic data collecting mechanism, database mechanism, and presentation mechanism are the three levels of perspective from which GIS mechanism can be seen. A few of the features of GIS are short turnaround times, a focus on user interface, and rapidly evolving technology [2]. It is essential to evaluate the system design procedure to make sure that geographic information processing capabilities are incorporated into a system. Text, pictures, audio, and video are just a few of the information formats that GIS can handle. Thus, attaining this objective necessitates a sound development strategy. [2]

II. SYSTEM DEVELOPMENT METHODOLOGIES

A system development methodology is a process or strategy that may be used to conveniently and practically design and implement each stage of the systems development life cycle (SDLC). Planning, analysis, design, and implementation are the four main phases of the system development lifecycle that are shared by the majority of system development approaches. To answer the question, "Why should we develop the system?" planning is in charge of detecting and recognizing design issues in the initial phase. The scope and constraints of the current system were ascertained, the capabilities of the proposed project or system were explored, and the necessary customer requirements were identified during the second phase, which was called the analysis phase.

Through the creation of predetermined functional and non-functional requirements, the development team decides how the entire system will operate in the third stage. The development team creates and

constructs the system in the last phase. These system development methodologies are categorized based on a number of factors, including whether they are traditional or heavyweight, agile or lightweight, predictive—where the project's scope can be accurately expressed—and adaptive—where it is challenging to articulate the project's requirements and scope early in the SDLC. [3]

III. AGILE DEVELOPMENT METHODOLOGIES

Agile development refers to a software development methodology formed based on an account of software development ideology by Kent Beck and 16 other authors. Methodologies for development of agile are considered a modern software development technique that emerged in the mid-1990s. The key intention of these methodologies is to decipher the teething troubles and inconvenience of conventional software development by keeping customer needs stable in business and IT environments and at the same time delivering high quality software on time while accommodating rapidly changing demands.

Therefore, key characteristics of agile development, such as short product delivery times and customer involvement at most stages of the SDLC, are considered to be at the core of these approaches, providing greater flexibility to accept customer changes and assess project priorities at any time, even at the last moment of project development. And through these essential characteristics of agile development, individual creativity and innovation are stimulated through the implementation of policies and procedures. [3]

IV. PHASES INVOLVED IN THE AGILE SOFTWARE DEVELOPMENT LIFE CYCLE

This section defines the distinct phases of the agile software development life cycle and the corresponding activities related to each phase that are performed and accomplished.[4][5]

4.1 Project Vision and Approval- The project vision and approval stage is considered the first stage of the agile software development life cycle, analyzing the key issues of the current system and identifying requirements for the new system. Therefore, the project manager and development team members define the objectives, feasibility studies, scope, and limitations of the proposed system. Additionally, at this stage, technical and economic risks are identified and the likely cost and duration of the project is also assessed. This phase is not repetitive and usually lasts 2-3 weeks. A general description of the system and a preliminary estimate are essential documents produced at this stage.

4.2 Exploration Phase- The Exploration Phase is an iterative and progressive phase aimed at investigating and discovering the basic requirements demanded by the customer through continuous

meetings between stakeholders and team members, through interviews, workshops, brainstorming sessions, or other forms of requirements elicitation techniques. Requirements are captured as user stories and can be documented in story maps for future use. The members of the development team are divided into two groups, one group containing experienced members and the other group containing inexperienced members. Experienced teams are responsible for communicating with stakeholders to understand and acknowledge issues and requirements, while less experienced team members are trained in agile processes and technologies to improve the quality of the products being developed. Additionally, feedback on the latest version is taken into account during this stage, and major changes in the latest version are identified as new requirements.

4.3 Iterative Planning Phase- The Iterative Planning phase is the most important phase in the Agile software development life cycle and includes many activities required for project planning. Since agile projects consist of multiple releases in the form of iterations and sprints, the first activity focuses on analyzing the features gained from the previous release with respect to the features of the latest iteration. The second activity involves prioritizing requirements based on a number of factors, including value, knowledge, and financial return. Additionally, the stack requirements list is updated based on feedback and requirements received from customers. This phase also evaluates the size and duration of previously planned iterations and confirms system resource requirements. The result of this step is to prioritize the stack requirements so that the next iteration selects a specific set of requirements from the stack.

4.4 ADCT Stage- This stage is an iterative stage where ADCT refers to analysis, design, coding, and testing, which is responsible for creating the functionality of the system and refining and improving the functionality of the system with each iteration. Iteration plans are broken down into smaller releases and implemented over many iterations over 1-4 weeks. The first iteration selects the appropriate story for the overall system architecture, and the remaining iterations design, code, and test it. Therefore, during the final iteration, the system will be ready for deployment to customers. Main activities in This phase includes simple design, maintaining coding standards, test-driven development (TDD) testing, and unit and functional testing. The main deliverables of this phase are the design documentation and system code.

4.5 Launch Phase- This phase consists of two main phases: the pre-release phase and the production phase. First, during the pre-production phase, additional testing, such as functional testing, acceptance testing, and integration testing, must be performed to ensure that the functional and non-functional requirements required by the customer are met. However, the production phase is responsible for

deploying the system to the customer and requires training to be performed in the customer's environment to ensure the highest level of usability and efficiency.

5. AGILE DEVELOPMENT METHODOLOGIES

There are many process models based on agile development principles, such as Extreme Programming, Scrum, DSDM, and Kanban. The following sections briefly describe four well-known agile models.

5.1 Extreme Programming- Extreme Programming (XP) is considered the first agile systems development methodology that takes an iterative approach to the project life cycle. In contrast, iterative methods, similar to traditional development methods such as waterfall, can have separate development phases such as requirements analysis, design, implementation, and testing, but in XP these activities are interconnected and considered continuous, so they occur iteratively. Furthermore, it was aimed at a small development team and at the same time required a small number of detailed and relatively systems-related artifacts. Extreme Programming is a list of 12 core practices that help development team members consistently embrace and embrace change. Therefore, these methods can be divided into four stages of system development: planning, coding, design, and testing. During the planning stage, these practices include daily meetings, frequent small outings, and moving people around. During the design phase, practices include looking for the simplest solution first, avoiding over-planning for future growth, and refactoring the code by continually improving its structure. During the coding phase, practices include being always available to the client, coding unit test cases first, introducing pair programming, where two programmers collaborate on the same code, and taking ownership of units of code, another feature of XP. Finally, during the testing phase, it is customary to create a new test case and run unit tests on the entire code when an error is discovered. [6]

5.2 Scrum- The Scrum methodology was started by Geoff Sutherland in 1993 and its main purpose is to apply the principles of agile methodology [7]. Scrum is a gradual, iterative approach called sprints. Sprints are iterations in which requirements are implemented; lasting no more than 30 days, and a shippable version of the software product is prepared and delivered at the end of each sprint. [8] [9] A sprint includes the following events: Sprint planning: At the beginning of each sprint, the Scrum Master, Product Owner, and team members meet to discuss product backlog items (backlogs are functional items). project requirements) are implemented during the sprint. Planning should consider the effort required and the prioritization of product backlog items. [10]

Daily Scrum meeting: Daily Scrum meetings are held at the beginning of the day for the teams involved in the development of the project and last no more than 15 minutes. Each member of the team should share their work progress and any difficulties encountered. [10] **Sprint Review**: At the end of each sprint, the Scrum Team and stakeholders meet to demonstrate the sprint results, discuss project progress, and share feedback. [10]

Sprint Retrospective: After the Sprint Review, a Sprint Retrospective is held. This is a discussion aimed at improving communication between team members and reflecting on what happened during the sprint. [10]

The main roles of Scrum are:

Product Owner: Responsible for recording software specifications and product backlog requirements. [7]

Scrum Master: Responsible for the implementation of Scrum methodologies by the team. [7]

Team: Responsible for project development and delivery. [7]

The Scrum methodology is one of the most popular among other agile methodologies because it is very flexible and allows changes to be made to the project at any time to ensure a suitable solution is provided. Scrum increases productivity by establishing effective communication between team members and giving teams the freedom to choose the best solution. On the other hand, when external users are involved, Scrum is not the best choice. Users need to be able to view the results of each sprint, which prevents them from having a clear view of project progress and defeats the purpose of the Scrum methodology. [9][11].

5.3 DSDM

One of the contemporary approaches to agile methodology is the Dynamic Systems Development Method (DSDM). It was initially solely concerned with software development, but it eventually evolved into a framework for project management in general, not just IT projects, in addition to solution delivery. Like every agile methodology, it has various benefits, including the customer's ongoing involvement throughout the project and the process's periodic and recurring nature. [12]

Its numerous tenets—such as prioritizing business needs and quality over quality, delivering projects on schedule, developing projects incrementally through iterations, relying on the cooperative principle,

which involves returning the project to users on a regular basis to gather feedback, and establishing control through plans—all help identify the best solution.[12]

DSDM includes the following steps:

- **Pre-Project Phase:** this stage involves deciding which project to develop, outlining its objectives and scope, and identifying the key players. Organizing the feasibility research stage as well. [13]
- **Phase of the Feasibility Study:** this stage examines the project's viability in terms of risk, resource availability, etc. A high-level strategy and a feasibility study are the end products of this phase. [13]
- **Business Study Phase:** the development team and the customer's experts hold workshops throughout this phase to establish system requirements, comprehend the business domain, and outline high-level procedures. [13]
- **The Functional Model Iteration Phase:** It is an iterative stage that involves repeated analysis, coding, and prototyping. The prototype is a non-throwaway that is regularly evaluated to be enhanced and incorporated into the finished product. It has also been constructed to specify the functional and non-functional requirements of the system. [13]
- **Design and Build Iteration Phase:** The needs from the functional model iteration phase are developed during the design and build iteration phase, which is likewise an iterative process. Users test the system and provide the development team with input so that it can be improved. [13]
- **Phase of Implementation:** during this stage, the system is delivered in its final form and users receive the required training. Depending on the system's size, this stage may be iterative. [13]
- **Post Project Phase:** The purpose of the post-project phase is to assess customer satisfaction and the attainment of business objectives following the project's completion and handover. [13]

The DSDM model has a number of distinct roles, including:

- **Developer:** The development team, which includes analysts, designers, programmers, testers, and others, is known as the developer. [14]
- **Technical Coordinator:** The technical coordinator is in charge of creating the system's architecture and guaranteeing its technical soundness. [14]
- **Ambassador User:** An ambassador user is a member of the user community who will ultimately utilize the system and who can communicate to developers the needs of other users as well as what the developers accomplish. [14]

- **Advisor User:** In contrast to ambassador users, advisor users are members of the user community who will ultimately use the system and provide specific project requirements. [14]
- **Visionary:** A visionary is a user who has knowledge of the business domain and its objectives; they act as a mentor, ensuring that the project is moving in the right path. [14]
- **Executive Sponsor:** An executive sponsor is a representative of the client's company that finances the project. [14]

One may argue that the DSDM has numerous benefits, such as its ability to construct systems quickly, adapt to best practices from other approaches, and offer a project's perspective from various angles (e.g. project management, risk control, etc.). There are numerous drawbacks, such as the abundance of roles, the lack of regard for the project's significance, and an unclear idea of the team's size and iteration duration. [12]

5.4 Kanban

One of the key agile approaches used as a software development project management tool is kanban. Furthermore, Kanban is employed as a Lean Manufacturing tool [15]. "A method enabling teams and companies to visualize their work, identify and eliminate bottlenecks, and achieve operational gains in terms of throughput and quality," kanban is a Japanese phrase that means signboard or graphic [16].

Phases of continuous implementation are used for the Kanban processes. The primary goal of these stages is to satisfy client needs. The "just in time" (JIT) approach to software development is the focus of kanban, which emphasizes tasks that must be completed.

Five traits for successful Kanban projects were identified by David Anderson, who developed and refined the Kanban approach in software engineering. These include:

- Visualize the workflow;
- Limit Work in Progress (WIP);
- Manage flow;
- Clearly specify Process Policies;
- Improve Collaboratively.

High levels of client, product owner, and project team involvement are made possible by the Kanban process, which takes into account an adaptive approach and incremental improvement process. All project tasks and functions will be kept in a product backlog following the definition of customer needs

with relevant stakeholders. A list of all tasks and functions arranged according to priority based on their significance is called the product backlog [16].

The Kanban board, which is regarded as a crucial component of the Kanban system, is used to display the job progress. Each task in the Kanban methodology goes through several stages before being completed, giving the approach the appearance of a queue [16]. Each column on a kanban board represents a particular stage or set of stages in a project, and the sticky notes represent a task within the project. This board facilitates workflow monitoring and task switching according to the product backlog. The Kanban board includes a limit for each column. Work in Progress (WIP), like the design column, is limited to two WIPs, meaning it can't design more than two jobs simultaneously [17].

There are many benefits of using Kanban in project management. Increasing flexibility, efficiency, and production is one of these benefits. If done correctly, the Kanban implementation will yield good costing [16]. A high level of engagement with the Kanban technique makes adjustments simple and boosts customer satisfaction [16]. A large project team is more suited for Kanban than a small one [17]. Managing client interaction during project work is one of the largest Kanban problems, and it calls for a highly skilled team [18].

6. GIS

An effective collection of computer-based tools called a Geographic Information System (GIS) can be used to collect, store, analyze, manipulate, and visualize geographic and spatial data. GIS is made up of three primary parts:

- A database component for storing and modifying spatial data;
- A presentation component for visualizing the data in an easy-to-understand format for spatial analytical operations;
- A component for entering geospatial data from field surveys, printed and digital maps, and remote sensing sources into the GIS.

Unlike other general information systems, GIS is a multi-dimensional information system that combines tabular data, sometimes referred to as attribute data, with geographical elements in a single information system. As a result, it is a multifaceted, intricate information system that integrates several fields of science and technology, such as operations research, computer science, geodesy, mathematics, and mapping.

Not just in cartography projects, but in many other areas like healthcare, agriculture, marketing, pollution control, emergency response teams, and many more, GIS helps with decision-making.

These are the most popular basic applications of GIS [19];

6.1 Mapping Data Representation:

Providing a single mean for storing and presenting geographical data in a mapped format is the most fundamental and popular application of GIS. An estimated 80% of the data that we take into consideration contains some kind of geospatial component. Decision makers can better comprehend links and patterns in geospatial situations by using this GIS-provided activity.

6.2 Proximity Analysis:

To assess possible sites for advertising campaigns and select the best site for building new branches across the nation, the marketing and commercial sectors rely significantly on GIS. GIS offers data on the various social, economic, and demographic elements influencing these commercial operations. It also considers the availability of other rivals in these areas.

6.3 Buffering:

To show the sphere of impact for a certain site, the buffering approach is frequently employed in conjunction with proximity analysis. In order to obtain more detailed information using various analysis techniques, buffering entails establishing a zone around a certain geographic feature, such as a point, line, or polygon. For instance, to determine the number of pharmacies within 2000 feet of a hospital, a 2000-foot buffer was built around the facility.

6.4 Locate Clusters:

In general, clustering is a technique for choosing a collection of unrelated points that meet certain predetermined standards. Clustering can employ density models for racial profiles by highlighting the areas with particular densities of a given population. In GIS, the number of points in a cluster may be the same distance from one another or larger than a certain quantity.

7. USING AGILE TECHNIQUES IN THE GIS AREA

GIS must adhere to formal systems development processes, which are defined as "a collection of procedures, techniques, tools, and documentation aids to support system developers in their efforts to

establish new information systems," just as other information systems [20]. Methodologies are crucial for directing the project team through each stage of the GIS development process. They are a collection of organized tasks that each team member adopts at each step, producing a more effective final product that ensures user needs are satisfied as effectively as feasible.

The 13th annual State of Agile survey, which was carried out from August to December 2018, is shown in the most recent report from VersionOne.com. People from a wide variety of industries in the global software development community were invited to participate in the survey; 1,319 complete responses were gathered, examined, and included in this study. According to the poll, 64% of participants utilize Scrum, and Scrum/XP Hybrid has long been the most popular agile methodology among respondents' companies. Additionally, the poll indicates that just 1% of firms use XP, but 54% of all respondents' organizations used the scrum approach, which received the best score. The remaining percentages were split among the other agile development methodologies, as seen in the figure below. [21]

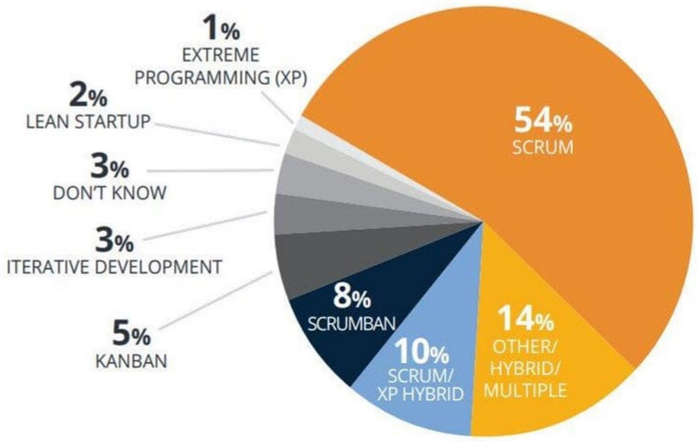

Fig 1: Agile Development methodology usage percentage wise

Methodology	Iterations	Artifacts	Team	Roles
XP	Typically, one or two weeks long for iteration	user stories, release and iteration planning	Self-organizing teams	No prescribed roles
Scrum	Defines iterations (called Sprints) from 2 to 3 weeks.	Must define product backlog, spring backlog, and burndown chart and keep them up to date.	Teams must be cross-functional	Product owner, development team, and Scrum master

DSDM	Different depend on project type	no artifacts	cross-functional team; specialized team	Ambassador user, Advisor user, Visionary, Development team, Product owner
Kanban	No define iterations	Can determine artifacts but not mandatory steps.	Team can cross-functional and specialized teams	No defines specific roles. Can define a product owner, development team or can split works flexibly between the team.

Table 1: Comparison of Agile Methodology

This document discusses many system development approaches in detail; each has advantages and disadvantages of its own. Depending on the needs of the clients, the resources available to the team, and the nature of the project, the GIS team must select the best option. They can use one of the pre-existing approaches, modify one to better suit the project, or create a brand-new approach tailored to the requirements of their project.

8. CONCLUSION

It is highly convenient to use agile approaches when developing Geographic Information Systems. The most effective development methodology is agile, which guarantees fast delivery, thorough design and testing, simple collaboration, and flexibility to adapt as the solution changes. We may conclude that there is no particular methodology to use in GIS creation after evaluating agile approaches. The type of project, available resources, and project requirements all influence how a GIS is developed. Out of all the agile approaches, Scrum is the one most frequently utilized to create successful apps, per the VersionOnce.com survey [21]. The most popular agile methodology worldwide is Scrum. Because of its flexibility, simplicity, ease of use, high performance, and other benefits, the scrum technique is preferred over other approaches for designing systems. Based on it, we recommend designing GIS apps utilizing the scrum process.

REFERENCES

1. Albert Ellis and Lisa Heneghan, Harvey Nash / KPMG CIO Survey 2017, KPMG International, https://assets.kpmg.com/content/dam/kpmg/xx/pdf/2017/07/harvey-nash-kpmg-cio-survey- 2017.pdf, 2017.

2. Appelo J., 2011. Management 3.0. Leading Agile Developers, Developing Agile Leaders. Addison-Wesley Professional.

3. Abbaspour, AS Mirkamali, M. Hesam Amiri, R. Moradi, K (2014): Explaining the Role of Human Capital Agility in Strategic Agility Development (Case Study of Ansar Bank), Quarterly Journal of Monetary and Banking Management, 2 (4), pp. 1 -24

4. Beatty, R. (2005), Workforce Agility: The New Frontier for Competitive Advantage, New York: Price Waterhouse Coopers LLP. (White Paper)

5. Breu, K., Hemingway, C.J., Strathern, M., Bridger, D.,(2002). Workforce agility: the new employee strategy for the knowledge economy. Journal of Information Technology 17(1), 21 -31.

6. Brandes U., Gemmer P., Kuschek H., Schültken,L., 2014. Management Y: Agile, Scrum, Design-Thinking & Co.: so gelingt der Wandel zur attraktiven und zukunftsfähigen Organisation. Campus-Verlag, Frankfurt am Main.

7. Brinkmann S., 2013. Qualitative interviewing, Series in understanding measurement. Oxford University Press, Oxford.

8. Conforto E.C., Salum F., Amaral D.C., da Silva S.L., de Almeida L.F.M., 2014. "Can Agile Project Management Be Adopted by Industries Other than Software Development? "Project Management Journal 45, 21–34.

9. Dove, R. & Wills, D. (1996). Transforming faculty into an agile workforce. in L. Richlin (Ed.), To Improve the Academy, Vol. 15, pp. 195 -207.

10. Dyer, L., and Shafer, R., (2003), "Dynamic organizations: achieving market -place and organizational agility with people", In: Peterson, R.S., Mannix, E.A. (Eds.), Leading and Managing People in the Dynamic Organization. Laurence Erlbaum Associates, Mahwah, NJ., pp. 1 -39

www.ijcrt.org © 2022 IJCRT | Volume 10, Issue 11 November 2022 | ISSN: 2320-2882

IJCRT2211052 International Journal of Creative Research Thoughts (IJCRT) www.ijcrt.org a445

11. Cappelli, P., & Tavis, A., (2021). HR Goes Agile. Harvard Business Review. Retrieved from https://hbr.org/2018/03/hr-goes-agile

12. Denning S., 2013. "Why Agile can be a game changer for managing continuous innovation in many industries". Strategy & Leadership 41, 5–11.

13. Dieckmann A. K. (2017). „Agilität im Einkauf–Eine Methoden-Architektur zur Konfiguration und Adaption einer agilen Belegschaft im Einkauf“. In Personalentwicklung in der Beschaffung (pp. 51-74). Springer Berlin Heidelberg.

14. Dove, R. & Wills, D. (1996). Transforming faculty into an agile workforce. in L. Richlin (Ed.), To Improve the Academy, Vol. 15, pp. 195 -207.

15. Dyer, L., and Shafer, R., (2003), “Dynamic organizations: achieving market -place and organizational agility with people”, In: Peterson, R.S., Mannix, E.A. (Eds.), Leading and Managing People in the Dynamic Organization. Laurence Erlbaum Associates, Mahwah, NJ., pp. 1 -39.

16. Ennals R., 2014. “Workplace Innovation”, in: Responsible Management. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1311–129.

17. Flemm, R., (2020). Six Ingredients of Agile Organizational Design. Retrieved from https://www.dcme.nu/the-six-ingredients-of-agile-organizational-design/

18. Gkiontsi D., & Karanika-Murray M. (2016). “Dealing with economic and demographic challenges: Workplace innovation practices as a timely and effective response to older workers' needs”. European Journal of Workplace Innovation, 2(1), pp.25-42.

19. Goll J., Hommel D., 2015. Mit Scrum zum gewünschten System. Springer Fachmedien, Wiesbaden. it-agile.de, 2015. The Scrum Cycle, it-agile.de. http://www.it- agile.de/wissen/methoden/scrum/

20. Goldman, S.L.; Nagel, R.N.; and Preiss, K., (1995), Agile Competitors and Virtual Organizations: strategies for Enriching the Customer, Van Nostrand Reinhold, New York, pp. 32 -39.

21. Griffin, B., Hesketh, B. (2003). Adaptable behaviours for successful work and career adjustment. Aust. J. Psychol, 55 (2), 65 -73.