The Barcelona Conference on Education 2023. Official Conference Proceedings.
The International Academic Forum (IAFOR)
DOI:10.22492/issn.2435-9467.2023.67

Understanding Agile: A Case Study in Educational Complexity

Stephen G. Barker, Cranfield University, UK

Barcelona Conference on Education 2023
Official Conference Proceedings

Abstract
As the nature of engineering concepts, products and processes becomes more intricate and detailed, so the task of understanding, managing and implementing them becomes more complex. This paper takes one such development concept, Agile, and examines how the orthogonal, multi-faceted nature of both the concept and its application makes it difficult to understand and employ and considers in this context the andragogical difficulties posed when trying to teach it to level 7 students. The practice of Agile is often considered to be either the application of methods such as Scrum, extreme programming (XP) or Scaled Agile Framework (SAFe), or the implementation of agile values or principles. In practice, however, a greater level of complexity exists and many more facets need to be considered, such as organisational structure, suitability of project and product, skill set of individuals, and even the mind set of those involved, because the adoption of agile requires more than the use of a technique as part of normal business procedures. In order to understand the complexity of a concept such as agile and the implications which it poses for andragogical teaching and learning strategies, the nature of the concept is first considered to identify the challenges that it creates for education, before a mapping of teaching strategies to agile facets is produced, and thought is given to the creation of an andragogical mechanism which will promote and engender student understanding of the concept and how it can be applied across a range of topic areas in context.

Keywords: Agile, Agility, Andragogy

Published by IAFOR. This is the Author Accepted Manuscript issued with: Creati e Commons Attribution Non-Commercial License (CC:BY:NC 4.0).
The final published version (version of record) is a ailable online at DOI:10.22492/issn.2435-9467.2023.67. Please refer to any applicable publisher terms of use.

Introduction

The practice of Agile, frequently taken to encompass ideas of agility or implementing agile principles, methods, or methodologies, is a topic that is often misunderstood or simplified to reduce the level of complexity and consequent difficulty in understanding implied by the concept. It can in fact encompass many factors that need to be considered before agile can be introduced and implemented successfully, and unless a more holistic view is taken on whether agile is an appropriate concept for use, at best intended benefits will not be realised, whilst at worse attempts to employ agile might result in disruption to organisational structure and processes. 'Agile' and 'agility' are concepts that include different ideas such as a manifesto (agile manifesto, 2001a), principles, approaches, methods and methodologies, some of which are presented as high-level structures or approaches, whilst others such as SCRUM (Schwaber, 1997; Sachdeva, 2016), XP (eXtreme Programming; Bryant, 2004; Stray et al, 2022), and SAFe (Scaled Agile Framework; Scaled Agile, 2023a). These mechanisms tend to promote the benefits of agile (Planview, 2023; Wrike, 2023a) and explain what agile is and how it can be achieved, but frequently they do not do so in context meaning that unless a specialist organization is contracted to implement an agile technique, these mechanisms can be both difficult to understand and difficult to implement. Essentially, whilst there are well-established and documented descriptions of what agility is, there is a lack of advice on how to tailor agile methods to particular problem situations in context or to specific project realities, and perceptions of how to do this tend to differ, which can only increase uncertainty as to which is the best way in which to proceed. Thus organisations (and individuals) tend not to either fully appreciate the nature of the task of 'doing' agile, or the issues in understanding how agile can be implemented.
In the face of these factors, it can be seen that agile has certain benefits, and is might be beneficial in certain circumstances. What those circumstances are, however, varies by factors such as context, need, project type and maturity, understanding and experience. The problem presented by this is therefore in determining:

(a) How can agile and agility – and its understanding – be understood?

(b) How can this be taught?

As such there is a need to examine the nature of agile, factors which characterise it and effect its understanding, what benefits it might bring, and when – crucially how – it is suitable for adoption. In pursuit of this understanding, this paper will endeavour to analyse concepts of agility comprising definitions and the various mechanisms for its achievement to understand their structure, advantages and benefits, before considering the potential pitfalls and disadvantages of adopting agile, and finally putting this in the context of how such factors can be best understood, and from the perspective of an educator, how they can be taught and what a suitable andragogical strategy might be to overcome issues of understanding and to provide a hopefully clearer understanding.

Concepts of Agility

In considering the nature of agile it can be seen that there are many definitions of 'agile', 'agility' and similar concepts, and there is a good deal of information available as to what agile is. Moreover, much has been written to advance theories which describe different approaches, methods, and methodologies which can be used to achieve it. If we first consider definitions of agile, the Oxford English dictionary (OED, 2023a) describes 'agile' as an adjective: "Able to move (esp. to climb or manoeuvre) quickly and easily; nimble, dexterous.

Also figurative and in extended use.", and 'agility' as a noun: "The ability to think and understand readily and quickly; quick-wittedness, alertness; mental dexterity; (also) an instance of this" (OED, 2023b). Other dictionaries provide similar definitions; for example, for 'agile', the Cambridge Dictionary (Cambridge Dictionary, 2023) suggests "[physically] able to move your body quickly and easily" and "[mentally] able to think quickly and clearly", whilst Dictinary.com (2023a) offers "quick and well-coordinated in movement". In a business sense, the Cambridge Dictionary gives us: "able to deal with new situations or changes quickly and successfully" (Cambridge Dictionary, 2023). Each of these definitions offers similar ideas of quickness, flexibility, clarity, involvement of individuals in understanding and reacting, and rapidity of reaction. These factors are also reflected in the principles of the Agile Manifesto for Software Development (agile manifesto, 2001b), as described at table 1 below.

Manifesto Principle	Rationale/method
Our highest priority is to satisfy the customer	through early and continuous delivery of valuable software
Welcome changing requirements	even late in development. Agile processes harness change for the customer's competitive advantage
Deliver working software frequently	from a couple of weeks to a couple of months, with a preference to the shorter timescale
Business people and developers must work together	Daily throughout the project
Build projects around motivated individuals.	Give them the environment and support they need, and trust them to get the job done
The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.	*Ensure efficient and effective conveyance of information*
Working software is the primary measure of progress	*As left*
Agile processes promote sustainable development.	The sponsors, developers, and users should be able to maintain a constant pace indefinitely
Continuous attention to technical excellence and good design	enhances agility
Simplicity is essential	the art of maximizing the amount of work not done
The best architectures, requirements, and designs	emerge from self-organizing teams
At regular intervals, the team reflects	on how to become more effective, then tunes and adjusts its behavior accordingly

Table 1: Principles of Agile Manifesto (Agile Manifesto, 2001b) split into principle and rationale

The principles at table 1 are based on the core values of the agile manifesto (agile manifesto, 2001a) as described below.

(a) Individuals and interactions over processes and tools
(b) Working software over comprehensive documentation
(c) Customer collaboration over contract negotiation
(d) Responding to change over following a plan

These principles set out a series of steps by which the values can be achieved, but it should be noted that they are software-focused and there are a series of assumptions and pre-requisites for successful adoption of these ideas not only in software development but also if they are to be adopted across a wider range of organisations and industries. Potential examples of this are described at table 2.

1.	Whilst having the customer as the highest priority is laudable, contractual value may result in prioritization of work and therefore response to customer (Hooles, 2017)
2.	Welcoming change to requirements is good, but may result in requirements creep, which in turn could negatively impact upon cost, workload, timelines and resourcing (Manos, 1993)
3.	Availability and workload may hinder or prevent business people having the time to work closely with developers especially on a daily schedule (Powl & Skitmore, 2005)
4.	Building projects around well supported motivated individuals is a good idea, but it assumes that people are enfranchised and motivated, and this is dependent upon factors such as workload and organizational culture (Milne, 2007; Kumar & Sundareshan, 2015)
5.	Most projects and organisations have a number of key performance indicators (KPIs), not just working software (i.e. project outcome)
6.	Agile processes are unlikely to promote sustainable development in isolation; other factors – and constraints – must be considered
7.	Simplicity – in product design and project performance – is an aim that is not always achievable due to pre- and post-requisites, project structure, risks, constraints etc
8.	The best architectures, requirements, and designs may well emerge from self-organizing teams, but not all organizational teams are capable of self-organisation (Weerheim et al, 2019), nor in context is this necessarily desirable
9.	Team reflection is a really good idea with significant benefits, but experience suggests that reflection and 'learning from experience' are activities for which there is often little time (Busby, 1998)

Table 2: Potential issues to adoption of agile principles

The qualities described by the agile principles can be seen as desirable should absolute speed of project or product development be necessitated, but these have to be balanced against the structure required for rigorous project practices, and these may well mitigate against the adoption of agile, especially if the project in question has interdependencies with other projects (or organisations), or is of a complex nature due to the type of procurement or development being undertaken. As such, it might be seen as unlikely that an organization could wholly adopt principles given the issues identified at table 2 unless it is entirely agile practices facing in its structure. Agility may therefore be considered as something that can be adopted either only in part, or for specific periods of project work or project tasks.

Agile Methods and Methodologies

A number of methodologies and methods have been proposed as approaches to agile project management and product development. These tend to be broadly linked to values and principles set out by the agile manifesto (agile manifesto, 2001b). Dictionary.com (2023b) describes agile development thus: "a philosophy of modular software development that delivers multiple successive versions of a working product that is improved after each iteration and evolves based on empirical evaluation of previous version", and agile methodologies are usually described as an iterative approach to delivering a product through a lifecycle (APM 2023; Wrike 2023b). These methodologies typically involve a number of stages e.g. Ideation, Development, Testing, and Operations, or some variation upon these. There are many organisations which promote such methodologies, describe what they are and their perceived benefits, and even provide case studies (Adaptovate, 2023). Whilst these demonstrate application of agile methodologies, however, they do not tend, in any great detail, explain the process of application, meaning that it is difficult for organisations to adopt such approaches for themselves and apply them independently to their own circumstances and contextual situations.
In addition to these methodologies, there are agile methods – and the terms 'methodology' and 'method' are sometimes conflated. There are many methods, but perhaps among the most well-known are SCRUM, eXtreme Programming (XP), and Scaled Agile Framework (SAFe). SCRUM involves devolving decision-making activities to the operational level of project hierarchy (Schwaber, 2004) and calls for project teams to work through specified goals in iterative, time-limited periods known as sprints. Each sprint involves a complete development activity, and progress is assessed via short 'stand-up' team meetings. This intensive form of development has been shown to produce results, but has also been subject to criticism, with it being suggested that the adoption of SCRUM does not constitute agile in itself and indeed the method can ignore agile principles (Fowler, 2018). Moreover, de Souza Santos et al (2023) opine that the method might not have any impact, either positive or negative, upon project success. XP is a related concept which suggests that frequent, iterative, development of product artifacts in short cycles will improve productivity through frequent code reviews, introduction of artifacts only when required, and frequent liaison with the customer. This is closely linked to agile manifesto principles (agile manifesto, 2001b), but has been subject to criticism, it being suggesting that practices such as pair programming and collective ownership of code can be problematic (Copeland, 2001). SAFe, meanwhile, is a framework and set of principles based around lean and agile practices (Scaled Agile, 2023b). The principles are wide ranging and there has been criticism that the scope and intent of SAFe is too ambitious (Eklund et al (, 2014), but the core idea is that the framework can be scaled to whatever circumstance is necessary – and to this end, it has four configurations: essential, large, portfolio, and full (which encompasses the other three). The guidance – framework and principles – are described in terms of what they are, and what they are intended to achieve , but there have been some concerns raised about the ability to scale agile, especially in large organisations (Kalenda et al, 2018). Having looked at the nature of agile and the ways in which it can be applied, we will now consider some of the difficulties in understanding and characterising these ideas.

The Difficulty with Agile

There are many factors which agile difficult to understand, adapt, and implement, some of which have already been identified at table 2. These range from the cognitive through to the

realities of organizational behaviour. The above section illustrates tat some individuals or organisations have expressed concerns with the ease by which methodologies/methods can be adopted and applied, but we can also identify wider issues around understanding what agile actually is, whether it is beneficial in context, and if so, what approach might be most suitable. These issues might be classified as follows:

(a) Difficulty with understanding agile
(b) Difficulty with selecting an agile approach
(c) Difficulty with implementing agile

To deal with each of these in turn, it can be seen that the definitions and ways of describing agile could be confusing and could also affect understanding of the concept; approaches are described as frameworks, methodologies and methods, and sometimes interchangeably as such: SCRUM, for example, is described in different quarters as both a methodology and a framework, whilst it can also be seen as a method linked to a wider methodology. In addition, whilst different organisations promote their own versions of agile methodologies and even demonstrate through case studies how such methodologies can be applied, anecdotal evidence suggests that there is a significant variance between being shown a case study and being able to apply the methodology oneself in the particular context of specific organisational need. The ability to understand methods is also impaired by bespoke terminology – SCRUM, for example, employs terms (Scrum.org, 2023) such as sprint, SCRUM master, burndown chart, definition of done, backlog, which may not be clear to many beyond the circle of agile expertise and is the language of technical specialists rather than users. Furthermore, it can be argued that the more intricate and complicated – or complex – a project or organisational structure is, the more difficult it will be to implement a concept such as agile.

If understanding of agile concepts and terminology can be achieved, the next difficulty might be in selecting which framework, method or methodology is most appropriate in context; different providers will champion their own offerings, but without expert knowledge it can be challenging to decide whether agile is appropriate at all given a lack of impartial guidance on the subject, and if so to differentiate amongst these offerings and choose the most appropriate approach to adopt and implement. Part of the problem with such a decision is that a wider context must be considered. Due to the nature of orthogonal nature of agile itself, and the potentially complex nature of both organisational and project into which agile might be adopted and implemented. Many additional issues must be considered. These are detailed at table 3. The variation of factors described illustrate the breadth of issues that need to be considered before agile should be adopted and implemented. The mindset which suggests that such decisions can be predicated on the simple or binary analysis of agile principles and their suitability or application of an agile method does not consider the necessary wider context. Adoption of agile approaches is likely to require a rethinking of organisational culture which might encompass consideration of organisational structure, philosophy and practices, and the skillsets of employees. Even if benefit is seen in the introduction of agile approaches and the organisation and its staff are enthusiastic at the prospect of agile, the organisation must still be made ready for adoption of the concept (Grossman et al, 2004). This in turn could necessitate the outlay of expenditure in reorganisation, retraining, and potentially recruitment of suitably qualified staff before agile can be successfully adopted – and this is a sizeable consideration and undertaking. Moreover, different levels of knowledge and thinking about agile will be required at different levels of organisational hierarchy. At a strategic level, there will exist the ned for those suitably qualified and experienced to gauge whether agile is a

good fit for the organisation at all, and if so, what changes will be needed to accommodate it, whilst at a project level, staff will need experience of agile methods, knowledge of how to apply and implement them , together with experience of different project and product needs, scenarios, and even customer behaviour in order to achieve successful adoption of agile. The nature and structure of projects, and type of product or service being procured, will also have a significant bearing on how – and if – agile can be applied.

Organisational structure and processes - Agile may require restructuring procedures - Agile will require new practices and adoption of new ideas and ways of working	**Need for suitably qualified and experienced personnel** - An understanding of the implications of agile - Ability to decide if agile will add value
Type and nature of project - Project scope - Product type - Project dependencies - Project maturity - Inherent complexity	**Mind-set of project team members** - Skills - Experience - Knowledge - Mental model - Systems thinking skills

Table 3: Organisational factors concerned with adoption of agile

A further factor that needs to e considered is scaling – the overarching principle behind SAFe (Scaled Agile, 2023b). Even if an organisation undertakes the activities necessary to embrace agile and understands how to select and apply an appropriate agile approach, that approach must be scaled and tailed to the specific organisational context required, and little in the way of impartial advice exists to support this activity. Therefore there is a heavy reliance upon the skills, experience and expertise of those within the organisation. Having now considered issues presented by understanding, adopting and implement agile, we now endeavour to analyse how these factors can be addressed andragogically from an educational context.

The Challenges of Educating Agile

Having considered the nature of agile concepts and the issues which might affect the adoption and implementation of those ideas, we can postulate a number of reasons why concepts of agile and agility might be difficult to relate from an andragogical perspective. These are listed at table 4. These are separated into four key areas which broadly encompass issues already considered within this paper. Understanding the essence of agile covers not only the philosophy and rationale behind the concept, but also what it is, what its potential benefits are – and indeed what any negative aspects might be – and whether it is the right option for adoption given the individual context of organisation, project and product. Preconceptions about agile tackles the possibility of pre-defined ideas which may be erroneous as well as any assumptions and cognitive barriers to adoption, variety of techniques covers the wide choice of different approaches on offer, their individual structure and requirements, as well as how to differentiate between them, whilst ‘no one size fits all’ addresses the potentially dark art of tailoring and scaling to project scope and

scenario/circumstance, and finally, the lack of impartial guidance on how to adopt and implement agile is considered.

Reason	Factors to be concerned with
Understanding of the essence of agile	- understanding what it is and - whether it is relevant
Preconceptions about agile	- the idea that Agile is the straightforward application of either - A set of principles or - A technique
Variety of agile techniques	- Approaches - Frameworks - Methodologies - Methods
No 'one size fits all' solution	- Need to tailor and scale to suit product type and project maturity and context
Lack of advice on how approaches and techniques are to be implemented	- Heavily reliant on expertise of the systems developer - Multi-faceted nature: - Multiplicity of competing/conflicting factors needing consideration

Table 4: Factors requiring consideration when educating agile

Having given thought to factors which pose a challenge when teaching agile, consideration will now be given to an andragogical strategy which might address these issues

Thoughts on an Andragogical strategy for Agile

It occurs that the biggest single task from an andragogical standpoint is dealing with the orthogonal nature of agile, its understanding, adoption and implementation. It is important that a correct blend of teaching mechanisms is found which both encompass the necessary understanding of agile facets and relates to the learning styles of students so to facilitate comprehension. A conventional means of teaching concept understanding might be to employ soft systems modelling and other techniques to achieve holistic problem situation appreciation before employing 'harder' modelling methods to specify structural and behavioural needs and setting this within the construct of a systems lifecycle. This systemic approach is embodied within the discipline of systems engineering (SE). With agile, however, there is a wider context as described at tables 2, 3, and 4. This can be seen to spread across multiple domains, notably business, management and leadership, and engineering. In order to encompass to totality of agile from philosophy through to implementation, it would therefore be advisable to structure the andragogical approach to include not only SE but also information from other disciplines as necessary. A variety of teaching methods and approaches can then be employed to relate understanding of agile to students, and examples of this are provide at table 5. Given the diversity in concepts and perceptions of agile, careful consideration must be given to how this is balanced with individual learning preferences. Honey and Mumford (1982) and Barker (2021) have described how individual learning styles

are vital to good understanding of an educational offering; there are many ways to understand the preferences of individuals, and it is important that this is done in order to maximise takeaway from the learning experience – whilst at the same time offering the ability to learn from others via mechanisms such as group work and peer-to-peer discussion (Garside, 1996).

Agile consideration	Potential teaching mechanism
Approaches and methods	Explain approach or method in overview, walk through each of the stages; provide examples and demonstrate application to a real life case study; assess understanding formatively and summatively via reflective assignment
Selection of appropriate approach	Class discussion as to whether agile might be appropriate; scenario-driven workshop on choice of approach; formative feedback. Possible thesis topic.
Organisational structure and adoption of new skills and practices	Explore using 'before and after' examples of how this has this been done, relate this to student experience and knowledge
Agile working and developing an agile mind set	Walk through of a representative project where agile has been successfully applied. Discuss how people can focus on and adapt to a rapidly changing environment
Mapping agile to a product development lifecycle	Illustrate difference between traditional and agile lifecycle management. Demonstrate tailoring of lifecycle stages to suit application to an agile approach. Workshop agile lifecycle management to examples of student choosing
Complexity of the problem situation	Examine complexity theory and approaches to map and characterise complex problem situations. Discuss case studies, and workshop understanding of complex adaptive systems to consider changes of problem scope, customer needs and requirements upon a project over time. Formative exercises and feedback

Table 5: Mapping Agile to teaching techniques

It is also essential to focus upon a variety of teaching techniques to allow students to place their understanding within context. As such, real-life case studies, scenarios, use cases and other means of exploring alternative solution realities are likely to be vital to holistic understanding. Group work should be promoted agile understanding and implementation is multi-stakeholder endeavour, and this also facilitates cross-domain dissemination of ideas and learning amongst students.

Conclusions and Further Work

The key to understanding – and educating – agile is the taking of a complete and holistic understanding not only of agile, but of the organisational environment and context into which it is introduced. Agile should not, given its multi-faceted nature, be treated as an initiative or as a panacea; it is by its nature difficulty to understand and cannot rightfully be reduced to the application or otherwise of a set of methods or principles. It is important to understand where and when agile might be appropriate – and indeed if it is appropriate – before any consideration is given to what methodology, method or framework is to be used. Although definitions of agile and agility are well-established, understanding of them is still variable in some quarters and it is therefore important that a means to further understanding and provide a basis for judgement on the appropriateness of agile and its implementation is developed. Additional work is required to provide guidance on where and when agile is suitable, as well as what kind of agile is most applicable. From an andragogical perspective, this paper outlines ideas for providing education on agile, but a suitable strategy encompassing the required breadth, variation, and educational rigour for the subject matter must be constructed and put into practice before being subject to review and improvement over time.

References

Adaptovate (2023) "Agile Methodology Examples and Case Studies": https://www.adaptovate.com/global/agile-methodology-frameworks-models-agile-examples-case-studies/ (accessed September 2023)

Agile Manifesto (2001a) "Manifesto for Agile Software Development": https://agilemanifesto.org (accessed September 2023)

Agile Manifesto (2001b) "Principles behind the Agile Manifesto": https://agilemanifesto.org/principles.html (accessed September 2023)

APM (2023) "What is Agile Project Management?: https://www.apm.org.uk/resources/find-a-resource/agile-project-management/?gad=1&gclid=EAIaIQobChMIwfaI-aKCggMVEeztCh2YCA_1EAAYBCAAEgLDnfD_BwE (accessed September 2023)

Barker, S.G. (2021) Altered Andragogy: Lessons from Lockdown for Systems Engineering Education. *IAFOR 9th European Conference on Education.* 15th-18th July 2021. London.

Bryant, S. (2004). XP: Taking the psychology of programming to the eXtreme. In *PPIG* (p. 13).

Busby, J. S. (1998). The neglect of feedback in engineering design organisations. *Design Studies*, *19*(1), 103-117.

Cambridge Dictionary (2023) definition of agile: https://dictionary.cambridge.org/dictionary/english/agile (accessed September 2023)

Copeland, L. (2001) "How to… Extreme Programming" Computerworld. https://www.computerworld.com/article/2585634/extreme-programming.html (accessed October 2023)

De Souza Santos, R., Ralph, P., Arshad, A., Stol, K-J. (2023) "Distributed Scrum: A Case Meta-Analysis", ACM computing surveys. https://doi.org/10.1145/3626519 (accessed October 2023)

Dictionary.com (2023a) definition of agile: https://www.dictionary.com/browse/agile (accessed September 2023)

Dictinary.com (2023b) definition of agile development: https://www.dictionary.com/browse/agile-development (accessed September 2023)

Eklund, U., Olsson, H., Strøm, N (2014). *Industrial challenges of scaling agile in mass-produced embedded systems.* ISBN 978-3-31914-3583

Fowler, M. (2018) "The state of agile software in 2018" https://martinfowler.com/articles/agile-aus-2018.html (accessed September 2023)

Garside, C. (1996). Look who's talking: A comparison of lecture and group discussion teaching strategies in developing critical thinking skills". *Communication Education*, *45 (3)*

Grossman, F., Bergin, J., Leip, D., Merritt, S., & Gotel, O. (2004). One XP experience: Introducing agile (XP) software development into a culture that is willing but not ready. In *Proceedings of the 2004 conference of the Centre for Advanced Studies on Collaborative research* (pp. 242-254).

Keefe, K., Sheard, J., & Dick, M. (2006, January). Adopting XP practices for teaching object oriented programming. In *Proceedings of the 8th Australasian Conference on Computing Education-Volume 52* (pp. 91-100).

Honey, P. & Mumford, A. (1982) *The Manual of Learning Styles.* P. Honey, Maidenhead, UK

Hooles, A. (2017). How to Contract Successfully for Agile Software Development. *Int'l. In-House Counsel J.*, *11*, 1.

Kalenda, M., Hyna, P., Rossi, B. (2018) Scaling Agile in Large Organizations: Practices, Challenges, and Success Factors. *Journal of Software: Evolution and Process.* 30 (10)

Kumar, S. A., & Sundareshan, V. (2015). Organisational Behaviour and Motivation Impact on Employee Performance. *ITIHAS-The Journal of Indian Management*, *5*(4).
Manos, K. L. (1993). Strategies for preventing future "requirements creep". In *INCOSE International Symposium* (Vol. 3, No. 1, pp. 375-380).
Milne, P. (2007). Motivation, incentives and organisational culture. *Journal of knowledge management*, *11*(6), 28-38.
OED (2023b) definition of 'agility': https://www.oed.com/search/dictionary/?scope=Entries&q=agility&tl=true (accessed September 2023)
Powl, A., & Skitmore, M. (2005). Factors hindering the performance of construction project managers. *Construction Innovation*, *5*(1), 41-51.
Planview (2023) "Benefits of Agile", https://www.planview.com/resources/guide/agile-methodologies-a-beginners-guide/benefits-agile/ (accessed September 2023)
Schwaber, K. (1997). SCRUM Development Process. In: Sutherland, J., Casanave, C., Miller, J., Patel, P., Hollowell, G. (eds) *Business Object Design and Implementation*. Springer, London. https://doi.org/10.1007/978-1-4471-0947-1_11 (accessed September 2023)
Schwaber, K. (2004) *Agile Project Management with Scrum*. Microsoft Press. ISBN 978-0-7356-1993-7
Sachdeva, S. (2016). Scrum Methodology. *Int. J. Eng. Comput. Sci*, *5*(16792), 16792-16800.
Scaled Agile (2023a) "What is SaFe?", https://scaledagile.com/what-is-safe/ (accessed September 2023)
Scaled Agile (2023b) "SAFe Lean-Agile Principles": https://scaledagileframework.com/safe-lean-agile-principles/ (accessed October 2023)
Scrum.org (2023) "Scrum Glossary": https://www.scrum.org/resources/scrum-glossary (accessed October 2023)
Stray, V., Hoda, R., Paasivaara, M., Lenarduzzi, V., & Mendez, D. (2022). Theories in Agile Software Development: Past, Present, and Future Introduction to the XP 2020 Special Section. *Information and Software Technology*, 107058.
Weerheim, W., Van Rossum, L., & Ten Have, W. D. (2019). Successful implementation of self-managing teams. *Leadership in Health Services*, *32*(1), 113-128.
Wrike (2023a) "The benefits and advantages of Agile": https://www.wrike.com/agile-guide/benefits-of-agile/ (accessed September 2023)
Wrike (2023b) "What is Agile Methodology in Project Management?": https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/ (accessed September 2023)

DAS: No new data was created or analysed in this study. Data sharing is not applicable to this article.

https://dspace.org/

Understanding agile: a case study in educational complexity

Barker, Stephen G.

2023-11-11
The International Academic Forum (IAFOR)
https://doi.org/10.22492/issn.2435-9467.2023.67
Barker SG. (2023) Understanding agile: a case study in educational complexity. 4th Barcelona Conference on Education 2023,19-23 September 2023, Barcelona, Spain. pp. 847-859

Downloaded from a DSpace repository. https://dspace.org/