*Corresponding author: Anabel Milagros Salazar, Project Management Department, Energy Systems Catapult, United Kingdom, E-mail: m.rana@ucem. ac.uk

Citation: Salazar AMS, Rana MQ, Leyba ACG, Saini M, Oladinrin OT, et al. (2024) Critical Success Factors in Large-Scale Agile Software Development. J Archit Eng Tech 13: 379.

Page   of

represents a holistic mode of thinking, transcending reliance solely on specific techniques or tools.

Large-scale ASD

The notion of 'agile development at scale' encompasses various interpretations [12-14] delineate two primary perspectives on scaling agile: tactical scaling and strategic scaling. Tactical scaling involves adapting agile delivery approaches to address scalability factors such as team size, regulatory requirements, and geographic dispersion. On the other hand, strategic scaling entails implementing agile practices across an organisation or its I.T. department. These perspectives are distinct yet interconnected; successful execution of tactical scaling is often a prerequisite for achieving strategic scaling.

Large-scale projects entail substantial risks and frequently encounter challenges such as delays, budget overruns, and even project failures [15-17] attribute this heightened risk to the complexity of governance systems in large-scale projects, which typically correlates with the number of teams involved in development. The greater the number of individuals and teams engaged in a large-scale project, the greater its inherent complexity.

Challenges when implementing large-scale ASD

According to Dikert [18], any organisational transformation involving many individuals will face challenges. Based on the annual State of Agile Survey [19], the most significant implementation challenges are related to organisational culture, particularly organisational culture-related resistance to change and deficiency in management support and sponsorship [19]. Figure 1 presents the challenges reported in the 2020 survey.

Moreover, Dikert [18] presented a systematic literature review on large-scale Agile transformations. 35 challenges were reported and classified into 9 categories, as shown in Table 1.

Challenges category subcategories

Difficulty in implementing Agile Misunderstanding agile concepts and lack of guidance from literature, Agile could have been customised better, reverting to the old way of working and excessive enthusiasm.

Change resistance General resistance, scepticism towards the new way of working, and top-down mandate create resistance and make management unwilling to change.

Lack of investment Lack of training, lack of coaching, too high workload, old commitments kept and rearranging physical spaces.

Coordination challenges in multi-team environments Difficulty interfacing between teams, autonomous team model challenges, global distribution challenges, and technical consistency are also challenges.

Different approaches emerge in a multi-team environment. The interpretation of agile differs between teams, and old and new methods are used side by side.

Hierarchical management and organisational boundaries, Middle managers' new role in agile is unclear; management is in waterfall mode, keeping the old bureaucracy and internal silos kept.

Requirements engineering High-level requirements management is largely missing in agile, requirement refinement are challenging, creating and estimating user stories is difficult, and there is a gap between long-term and short-term planning.

Quality assurance Accommodating non-functional testing, lack of automated testing and requirements ambiguity affect QA.

Integrating non-development functions in the transformation other functions are unwilling to change, and there are challenges in adjusting product launch activities, adjusting to the incremental delivery pace and rewarding model, and needing to be teamwork-centric. In essence, challenges can be considered a success factor (S.F.)- S.F.s is not measurements of success but rather something that needs to be done well to achieve objectives [20]. Therefore, a comparison of figure one and table one reveals that the most common S.F.s are general change resistance, scepticism towards Agile practices, lack of training, inconsistent processes and practices across teams and the pervasiveness of traditional mode.

Recommendations for implementing agile practices large-scale

According to [7], many managers preempt large-scale teams, which need help developing long-term projects. However, in essence, the number of agile teams that can be created and the size of the initiative are not limited. Leaders must, however, be practical. Organising every function into agile teams is inappropriate, as agile approaches are inadequate for certain activities. This view is supported by [13], who state that it is crucial to understand if agile approaches can be appropriate for large-scale as well as where and how. Therefore, customising agile methods has frequently been critical for agile implementation. Since every company will have its challenges with
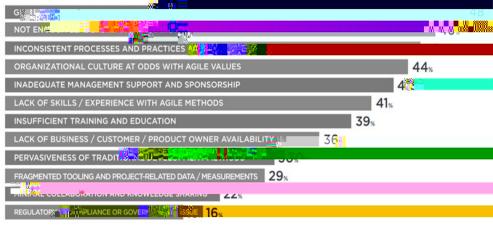

Figure 1: Agile Survey Challenges.

agile implementation, when selecting agile approaches to implement, the focus of the business areas should be carefully considered [5,7] propose 3 signi cant steps to scale up Agile:

Leading agile by being agile: According to [7], when leaders have not understood and implemented agile practices themselves, they will attempt to scale up agile the same way they have approached other change initiatives: through top-down plans and directives, which

need to be implanted by leadership and management teams. Agile principles and values can be implanted throughout the organisation through joint value workshops and active communication regarding change within the organisation.

- Changes in the business's operating architecture: e architecture is an important aspect that a ects productivity positively or negatively.   erefore, it is essential to make changes in this business area to align it with the business objective. Consequently, creating a common backlog with a single priority list is recommended to increase the productivity and product deliveries of the teams. In the case that not every team can adopt any task from the list either because of knowledge gaps or costs, the following suggestions can be implemented:

    o   e teams' focus should be changed to specialising in speci c business ows.   is would not require them to know all the product components but only a few. Consequently, a business ow-based backlog list can be created.

    o   Create statics teams within product families.

Conclusion