

An Integrated Framework for Design Thinking and Agile Methods for Digital Transformation

Kavitha Gurusamy¹, Narayanan Srinivasaraghavan^{1(✉)}, and Sisira Adikari²

¹ Australian Public Service, Canberra, ACT, Australia
Kavitha_rkumar@yahoo.com.au, sangeethaandnarayanan@gmail.com

² University of Canberra, Bruce, ACT 2601, Australia
Sisira.Adikari@canberra.edu.au

Abstract. This paper proposes a framework to deliver a faster workable project module in an innovative, dynamic environment. The increased demand in mobile and internet applications drive businesses to rethink their customer needs to a greater extent and undertake digital transformation to compete in the marketplace. The agile methods as an approach is gaining popularity to develop innovative solutions while Design Thinking leads to transformation, evolution and innovation, and to form new ways of managing current and future business opportunities. The proposed framework supports the evolving Digital Transformation by combining the benefits of Agile methods and Design Thinking. This paper contributes to the body of knowledge on Design Thinking and Agile methodology by providing a framework which integrates the core processes of Agile methods and Design Thinking.

Keywords: Digital transformation · Design thinking · Agile methodology

1 Introduction

In this digital era, consumers are highly reliant on mobile applications for their day to day activities. This drives businesses to rethink their changing customer needs and undertake digital transformation to compete with their business competitors. Most of the organizations have undergone Digital Transformation in some way to provide improved customer service and experience. This revolution attracted researchers around the world. Researchers created new business models for Digital Transformation [1] and implemented Digital Transformation using Agile methodology [2]. One of the key aspects of Digital Transformation is creativity and is based on networking ideas and mindset. The success of Digital Transformation relies largely on organisational, technical, strategic planning and a team with high competence, high commitment that respond well to delegation [3]. One of the biggest barriers to implementing the Digital Transformation is the confabulation and co-mixing of operations, corporate IT, product development & technical, strategic planning functionality [14]. Both iterative and agile methods were developed to overcome the various obstacles faced in sequential forms of project organization. As technology projects grow in complexity, it is difficult for users to define the long-term requirements. Projects developed by constant iterations

(agile) can provide an opportunity for clients/users to provide feedback to help refine their requirements [2, 3]. However, integrating Design Thinking into the agile iterations overcome the challenges posed by Agile methodology and can contribute to deliver constant continuous deliveries to the customers faster than the traditional approach.

1.1 Design Thinking

Design Thinking (DT) is the catalyst for innovative development or transformation. DT is a human-centric approach and defined in many ways by different authors. For example, Tschimmel [4] defined Design Thinking as a way of thinking which leads to transformation, evolution and innovation, to a new form of living and to new ways of managing business. DT is emerged from Cognitive Process. However, DT today is not only a cognitive process or a mindset but has become an effective toolkit for any innovation process, connecting the creative design approach to traditional business thinking, based on planning and problem solving [4]. DT integrates human, business, and technological factors in problem forming, solving, and design. Its human-centric methodology integrates expertise from design, social sciences, engineering, and business [5].

The understanding and acceptance of failures and mistakes are important elements of DT, which differentiates DT from the traditional way of thinking business [4]. Three major constraints of DT are desirability, feasibility and viability. The first stage of the design process is often about discovering which constraints are important and establishing a framework for evaluating them. Design thinker brings the constraints into a harmonious balance rather than a competent designer resolving the issue [6]. According to Brown [6] Designers are different to Design Thinkers. Designers have learned to excel at resolving one or two or even all three of the DT constraints. Design Thinkers, by contrast, are learning to navigate between and among them in a creative way. Design Thinkers shifted their thinking from problem to project [6]. Images play an important role in DT. Design thinkers usually apply sketches, drawings and material models to explore project problems and solutions [4].

Design thinking is an interplay between diverging exploration of problem and solution space and converging processes of synthesizing and selecting. Plattner et al. [5] related design thinking to the structures, cultures and processes of IT development – in particular with regard to the following aspects: • Building on Diversity, Exploring the Problem Space, Exploring the Solution Space, Iterative Alignment of Both Spaces (Fig. 1).

Building on Diversity: Having a team of individuals with diverse skills and communicating frequently and closely collaborating during the design phase.

Exploring the Problem Space: Developing a comprehensive understanding of the problem before the actual development process starts, by exploring various scenarios and important use cases by learning about the users and their specific needs.

Exploring the Solution Space: Promoting creative ideation and conceptualization process by considering a number of options and choosing a solution that best meets the needs of the users.

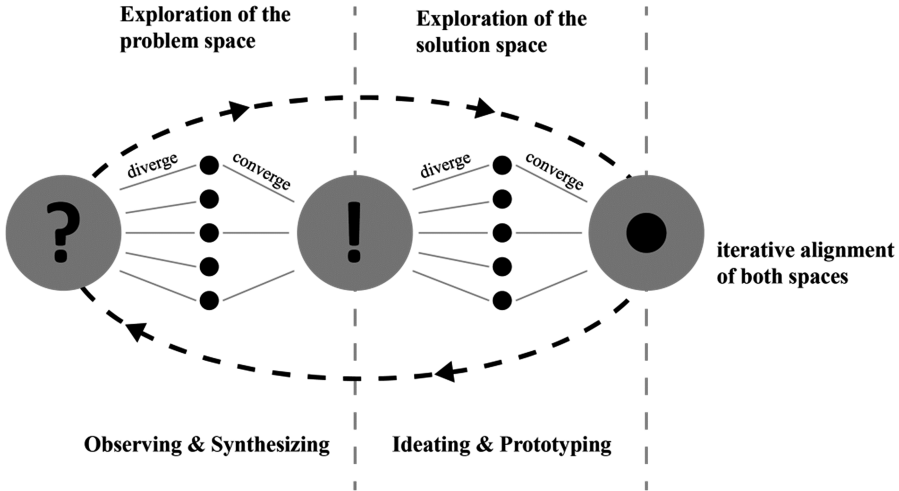


Fig. 1. Problem and solution space in design thinking [7]

Iterative Alignment of Both Spaces: Enabling a highly iterative development process by seeking and integrating feedback from within the team, users and experts throughout the development phase.

In both problem and solution space, there is room to explore multiple options through the divergent phase. The convergent space allows concentrating on one solution amongst many that drives to project solution.

1.2 Agile Methodology

Over the last decade, agile development approach has received much attention from researchers and practitioners as an approach to deal with project development in an innovative and dynamic environment. A core principle of Agile development is to satisfy the customer by providing valuable product on an early and continuous basis. The agile method aims to deliver a small set of features/working modules to the customers as quickly as possible in short iterations. This reduces much of upfront research and analysis required by traditional methodologies. However, in some cases, these fast deliveries do not necessarily need to satisfy user requirements in the end [8] due to lack of prior architectural design, research and poor documentation [9].

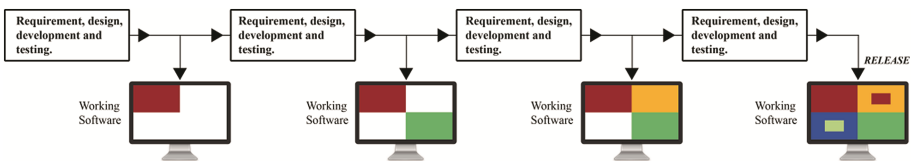


Fig. 2. Agile iterations [12]

Agile development processes are people and code-centric approach [10]. Agile development processes do not follow rigid milestone roadmaps, but instead by a set of rules that the team can easily act flexibly to adapt to unexpected events [5].

Agile methodologies such as Scrum and Extreme Programming (XP) advocates iterative development processes with the frequent communication with the customers in order to release a piece of workable module frequently. XP heavily focuses on development techniques whilst Scrum focuses on roles, processes and project management. In scrum project development, a scrum team consists of analysts, developers and testers work together in a centralized environment. All development processes and activities are transparent and visible. Scrum development activities are structured through a project backlog that consists a list of prioritized development activities [11, 12]. XP advocate continuously refactoring code to improve code quality and system architecture [13].

1.3 Research Motivation

This research focuses on helping business to take advantage of Digital Transformation in a cost effective manner. The aim is to produce an integrated framework using Design Thinking and Agile methodology to address the challenges posed by Digital Transformation. This research aims to deliver a better framework to produce a product that best meets customer needs with minimum costs, waste and time, and enables the business to achieve bottom line gains earlier than via traditional sequential approaches.

The rest of the paper is organized in the following order. Following the introduction, Sect. 2 provides commonality between the DT and Agile methodology whilst Sect. 3 reports the existing research on DT and Agile approach. Section 4 discusses the challenges posed by the Agile and DT methods. The proposed framework is discussed in Sect. 5. Finally, limitations of this research and future work are addressed with the conclusion.

2 Design Thinking vs. Agile Methodology

There are commonalities between DT and Agile methodologies. Hirschfeld et al., [10] identified commonalities with respect to DT and Agile methodologies which includes Wicked Problems: refers to problems that are not well understood and thus difficult to describe, becomes clearer when moving closer to the solution of the problem; Close Interactions: people interact closely with each other, exchange a lot of knowledge and opinions, which in turn supports making progress; Go for Feedback: iterative and incremental development is fundamental to agile approach [10]. Similar iteration is used in DT's problem and solution exploration [5]. Plattner et al. [5] discussed existing IT development approaches and their relation to DT in the context of agility. Their comparison with design thinking and agile shows some strong parallels: core features like “user-centricity”, “iterative learning and development processes”, and “extensive team communication” and identified that design thinking methodology has been already introduced to IT development [5].

3 Research on Design Thinking and Agile Approach

For all entrepreneurs, it is useful to apply DT when moving through a creative process of problem solving or when looking for opportunities and when facing challenges [4]. DT could be used in a range of fields. For example Simons et al. [14] adapted design thinking approach on Biotechnology Research and Development, which helped them to create innovative content repeatedly at a fast pace.

Simons et al. [14] adopted DT concepts in many ways. They used new perspectives from outside to gain an understanding of the problem and unexpected connections. Through radical collaboration, collected richest insights from team members. Further explored options and ideas early to reveal unexpected insights and gained evidence quickly to improve the decision making. Some research successfully integrated Agile approach with other research methodologies to enhance the power of the agile method. For example Hirschfeld et al., [10] Integrated Design Thinking methodology in Agile development process, and developed a model and used it to develop software in Virtual collaborated environment; Ferreira et al. [15], introduced User Interface Design in Agile software development; Fox et al. [8] analysed prior research that integrated User-Centric approach in Agile software development process, and identified that the integration added value to the software development process. Based on their findings, Fox et al. [8] tried to identify a common model with the integration of agile and user-centered approach and named them as Generalist, Specialist, and the Hybrid approach. The introduction of User-Centered Design specialists along with agile developers in each iteration improved the usability of the products. This confirms integration of agile and user-centered approach adds value to the innovative developments.

4 Challenges on DT and Agile Approach on Software Development

Agile breaks down the problem into small chunks that people can deal with and stay on track which removes options from the table. But DT tries to keep options on the table for as long as possible [5]. Agile reduces much of upfront research and analysis required by traditional methodologies. However, in some cases, this fast delivery doesn't satisfy user needs in the end. On the other hand, User-Centric approach spends a considerable amount of time for research and analysis prior to development, and takes longer to deliver the product [8].

Cao et al. [9] identified some of the challenges posed by the agile approach. During the development process, more emphasis is given on workable systems than documentation. This poses lack of formal architectural design and leads to irrecoverable architectural mistakes. Agile methods heavily rely on on-site customer input rather than predefined requirements documentation. This poses customer-related challenges as the customer's knowledge of the requirement for larger complex project is limited. Further, Agile methods heavily rely on tacit knowledge amongst the team members, so critical decisions may not be documented. This leads to developer related challenges. Agile works well for flat organizational structure but not for hierarchical structure. This introduces organizational challenges.

On a theoretical level, we have seen that design thinking can make valuable contributions to IT development and enhance the innovativeness of IT development. However, there are still unresolved challenges in applying DT successfully to IT development [5, 7]. The general concern about DT is too much time is spent on understanding the problem before actual development process starts. This reduces DT's viability when the project is under time pressure. Especially people reporting to higher hierarchy perceive DT as a risk and prefer secured milestone-based development process [7]. The proposed framework tries to address some of these challenges.

5 Proposed Framework

The proposed framework integrates DT and agile methodology for digital transformation. This framework extracts the benefits from both the methods and compliments the challenges posed by the methods. Adapting the DT concept on Agile transformation opens enhancement and can be applied in agile iterations.

The DT model shown above (Fig. 3) is from Hasso-Plattner Institute and is adapted for this research. The first phase *understand*, collects information about the topic or problem to be solved. *Observe* phase collects insights about the user needs by conducting interviews with the clients and research. The information collected from the first two phases are shared amongst the group and converted into a visual framework called *PointOfView*. In the *Ideate* phase an interdisciplinary team goes through the processes they have observed so far and design a solution with opportunities to change during the implementation. The core phase in the DT is the *Prototype*, and it turns ideas into an action. The last phase Test evaluates the requirements and user needs. All these processes are highly iterative and can be iterated from any phase a satisfactory product is delivered.

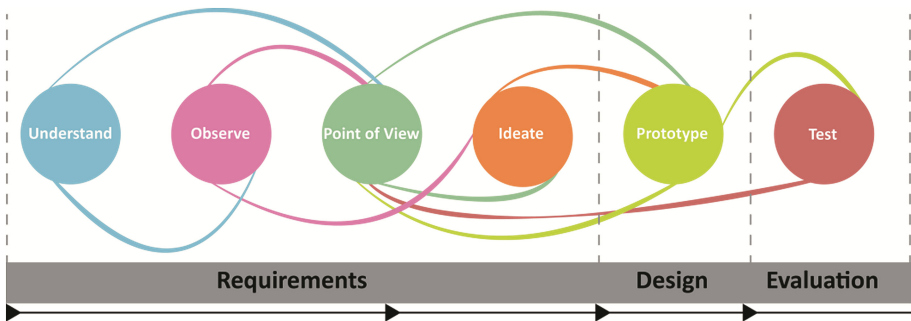


Fig. 3. Adapted design thinking process model [4]

In the proposed framework, the six DT processes are adapted into 3 phases called *Requirements*, *Design* and *Evaluation*. As the first 4 phases of the original DT model focuses on collecting information required starting an actual implementation, it is named as *Requirements* phase. In this phase, information is collected from different experts including but not limited to technical experts, engineers, end users and business analysts. The *Prototype* phase from original model is termed as *Design* because in this phase a

design prototype is created for real action. In the *Evaluation* phase the developed design will be evaluated to ensure it is ready to build the system. The adapted DT phases are going to be incorporated in proposed agile iterations.

Agile is a mindset; this can be applied to different situations as they arise. In Agile development, business representatives and developers work together daily throughout the project to produce workable modules. They prioritize the work and create backlogs for ongoing development. Agile development supports the practice of delivering a smaller workable product rather than delivering one product at the end of the project. The short deliveries and iterations allow the small chunks to be tested and improved during development.

The Agile Framework shown in Fig. 2 consists of 4 phases and these phases are iterated to produce small Workable Module (WM). Agile *Requirements* encompasses the task information that goes into determining the needs or conditions to meet for a new or altered product or project. In this phase the features that need to be built are defined, the quality measures are also defined to validate the workable modules. *Design* phase includes but is not limited to solution design, user interface design, and architecture design to build the system. In *Development* phase the development team splits the work between the developers and selects the prioritized jobs from the backlogs [8]. On completion, the workable module is sent to the next phase *Test*. Once implementation features are verified in the test phase, then the finished product is released to customers.

In the proposed framework (Fig. 4) the adapted 3 phase DT concepts *Requirements*, *Design* and *Evaluation* are applied in Agile's *Requirements* and *Design* phase. DT's divergent thinking is applied to identify multiple options to solve a problem while the convergent thinking provides a practical way of deciding among existing alternatives. The divergent phase explores the problem and solution space. The convergent phase drives us towards the solution.

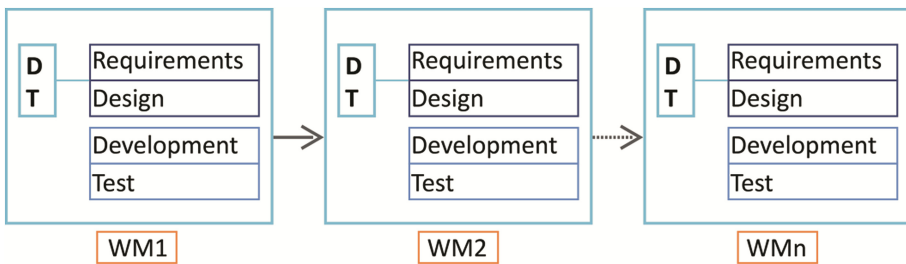


Fig. 4. Integrated design thinking and agile framework for digital transformation

The completed Requirements and Design phase outcomes will flow through to the agile Development and Test phases.

The challenges posed by the agile method such as lack of prior research are addressed in the proposed framework. In each agile phase, DT's exploration and convergent principles are applied which brings ideas from the multi-disciplinary expertise and provide room for prior research and analysis. Further, it provides the basis for envisioning and evaluating possible solutions. As these iterations are applied to the small module of the

project, the case of failure is taken as an advantage to improve it in the consecutive iterations. This concept is similar to Brown's [6] "Fail early to succeed sooner". As the proposed framework is highly iterative and the incremental process is driven by people with diverse background and expertise, the frequent deliverables are more likely to satisfy user needs.

6 Limitations and Future Work

This framework was developed based on existing theory and literature on DT and agile methodologies. Attention was given to prior research and findings were considered while developing the framework. However, the framework hasn't been tested in the real environment yet. A dedicated transformation team is important for long term success. However, the research hasn't focused on team management or how to improve the coordination amongst the team members and across various levels of the projects. Future research can address these issues. This research framework can be extended/adapted to any type and/or size of the organisation. This research is not focused on a specific Agile method to follow. Future research may focus on the type of organisations and specific Agile methods for Digital Transformation.

7 Conclusion

In this paper, we presented an enhanced framework based on Design Thinking and Agile concepts in the context of Digital Transformation. The main contribution of this paper is highlighted in the new approach in Digital Transformation and addressed the challenges posed by traditional approaches. This paper contributes the knowledge on DT research, Agile development and Digital Transformation.

References

1. Berman, J.S.: Digital transformation: opportunities to create new business models. *Strategy Leadersh.* **40**(2), 16–24 (2012)
2. Sriram, N.: *Agile IT organization design - for digital transformation and continuous delivery*. Addison Wesley, Reading (2015)
3. Baker, M.: *Digital Transformation - 4th Edition*. Buckingham Monographs (2014)
4. Tschimmel, K.: Design thinking as an effective toolkit for innovation. In: *Proceedings of the XXIII ISPIM Conference: Action for Innovation: Innovation from Experience*, Barcelona, pp. 1–20 (2012). ISBN 978-952-265-243-0
5. Meinel, C., Leifer, L.: Design thinking research. In: Plattner, H., Meinel, C., Leifer, L. (eds.) *Design Thinking: Understand - Improve – Apply*, pp. xiii–xxii. Springer, Heidelberg (2011)
6. Brown, T.: *Change By Design – How Design Thinking Transforms Organizations and Inspires Innovation*. HarperBusiness, New York (2009)
7. Lindberg, T., Meinel, C., Wagner, R.: Design thinking: a fruitful concept for IT development? In: Plattner, H., Meinel, C., Leifer, L. (eds.) *Design Thinking: Understand - Improve – Apply*, pp. 1–20. Springer, Berlin (2011)

8. Fox, D., Sillito, J., Maurer, F.: Agile methods and user-centered design: how these two methodologies are being successfully integrated in industry. In: AGILE 2008, Conference, pp. 63–72. IEEE Computer Society (2008). ieeexplore.ieee.org
9. Cao, L., Mohan, K., Peng, X., Peng, X., Ramesh, B.: A framework for adapting agile development methodologies. *Eur. J. Inf. Syst.* **18**(4), 332–343 (2009)
10. Hirschfeld, R., Steinert, B., Lincke, J.: Agile software development in virtual collaboration environments. In: Plattner, H., Meinel, C., Leifer, L. (eds.) *Design Thinking Understand-Improve-Apply*, pp. 197–218. Springer, Berlin (2011). ISBN 978-3-642-13756-3
11. Kautz, K., Johansen, T.H., Uldahl, A.: The perceived impact of the agile development and project management method scrum on information systems and software development productivity. *Australas. J. Inf. Syst.* **18**(3), 303–315 (2014)
12. GAO: Effective Practices and Federal Challenges in Applying Agile Methods GAO Reports. 7/27/2012, preceding, pp. 1–34 (2012)
13. Beck, K.: *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Boston (2000)
14. Simons, T., Gupta, A., Buchanan, M.: Innovation in R&D: using design thinking to develop new models of inventiveness, productivity and collaboration. *J. Commercial Biotechnol.* **17**(4), 301–307 (2011)
15. Ferreira, J., Noble, J., Biddle, R.: Agile development iterations and UI design. In: Agile Conference (AGILE), pp. 50–58. IEEE Computer Society (2007). ieeexplore.ieee.org