Design Thinking Techniques Selection in Software Development: On the Understanding of Designers and Software Engineers Choices

Lauriane Pereira[1], Rafael Parizi[1], Sabrina Marczak[1] and Tayana Conte[2]
[1]*Polytechnic School, Pontifical Catholic University of Rio Grande do Sul/PUCRS, Porto Alegre, Brazil*
[2]*Institute of Computing, Federal University of Amazonas/UFAM, Manaus, Brazil*

Keywords: Design Thinking, Software Development, Techniques Selection, Focus Group, Empirical Study.

Abstract: Design Thinking (DT) is a concept that promises increased innovativeness through a more user-centered approach. DT offers a mindset, working spaces, and techniques to support the generation of ideas and transform those into solutions. However, the selection of DT techniques is a complex endeavor since it needs to take into account the problem context and nature, user profile, among other characteristics. In addition, little is known about how do professionals make their selection. This paper reports on a focus group study with professionals working in software development. We used the Cynefin framework combined with the Double Diamond model to explore the process of selection of DT techniques for hypothetical scenarios. We found that the professionals need to respect the default domain to set their strategies and allow insights to emerge.

1 INTRODUCTION

Design Thinking (DT) groups a set of practices inspired by Design principles, using empathy and creativity to meet user needs and achieve goals (Weigel, 2015). DT offers a mindset, working spaces, and techniques to support the generation of ideas and transform those into solutions (Lindberg et al., 2011).

DT models, techniques, and frameworks (e.g., InnoDev, which combines DT with Scrum (Dobrigkeit et al., 2017)) have been proposed as mechanisms to support software development. For instance, Requirements Engineering is a discipline that explores DT to qualify the problem understanding and propose desirable solutions (Hehn and Uebernickel, 2018). However, using DT in software development configures a complex nature activity. It involves different stakeholders, multidisciplinary mindsets, and pressure for results generated by the competitive market (Hehn et al., 2020; de Paula et al., 2020).

The endeavor to select techniques to use through DT that fit the problem scenario and nature, user profile, stakeholder perspectives, and software team capabilities, to mention a few characteristics, is a complex task (Brenner and Uebernickel, 2016). Teams with little or no knowledge about DT may use techniques that are not suitable for their context or might not know the extent of what each technique or their combination has to offer (Souza et al., 2020). Parizi et al. (2020) identified that the selection of techniques is challenging even for experienced professionals.

DT techniques selection involves a process of choosing a subset of techniques from a set of alternatives based on a given criterion, and little is known about how do professionals make their DT techniques selection in software development, the long-term goal of this research is to characterize the decision making process of such selection to later provide computer-based support to it. Specifically, this paper reports on an exploratory focus group study that aimed to identify which techniques software engineers and designers working in software development would choose to solve hypothetical scenarios. We sought to answer the following research question: *How do designers and software engineers select techniques to support the use of Design Thinking in software development?*.

By using the Cynefin framework (Snowden and Boone, 2007) to inspire the definition of each of the scenarios and the Double Diamond model (Council, 2020), we learned that the professionals need to respect the default domain to set their strategies and allow insights to emerge. They also need to be aware that this domain change, depending on their decisions. Therefore, this paper brings the following contributions: i) exploration of the DT techniques selection in software development, ii) the viewpoint of designers and software engineers, and iii) usage of the Cynefin and Double Diamond to frame our study.

Pereira, L., Parizi, R., Marczak, S. and Conte, T.
Design Thinking Techniques Selection in Software Development: On the Understanding of Designers and Software Engineers Choices.
DOI: 10.5220/0010460403530360
In *Proceedings of the 23rd International Conference on Enterprise Information Systems (ICEIS 2021) - Volume 2*, pages 353-360
ISBN: 978-989-758-509-8

2 DESIGN THINKING TECHNIQUES SELECTION IN SOFTWARE DEVELOPMENT

DT can be defined from 3 perspectives (Brenner et al., 2016): (i) as a mindset, DT is focused on a strong orientation to discover the obvious and hidden customers' and users' needs and prototype the possible solutions; (ii) as a process, DT is composed of a micro-process in an innovation process and a macro-process in prototypes that must fulfill defined requirements and; (iii) as a toolbox, DT refers to applying numerous techniques (e.g., personas, brainstorming) from design, software engineering and psychology to explore and solve the problem-at-hand.

Literature reports a plethora of DT models (Park and McKilligan, 2018). One of the most well-known is the Double Diamond model (Tschimmel, 2012) which abstracts 2 main spaces: the problem space and the solution space. The first diamond (problem space) comprises the Discover space, representing the initial divergent phase where new opportunities and insights are sought and the Define space, which marks the consolidation of insights and looks to converge on ideas. The second diamond (solution space) is composed of the Develop space, representing the development of potential solutions for the identified ideas, and the Deliver space, which aims at the convergence into a solution upon validation. Although DT models such as the Double Diamond help conceptualize the understanding and solution proposal processes, these abstract working spaces require using a set of techniques to transform ideas into actions and results. Therefore, knowing the techniques and selecting the ones that best fit the situation can foster better project decisions and improve communication among multidisciplinary teams (Chasanidou et al., 2015).

The first step when considering the use DT is to decide on the toolbox, i.e., which techniques to apply (Carlgren et al., 2016). De Paula et al. (2020) claim that it is important to do some work upfront: investigate the stakeholders' profile, have an overview of their needs, understand the nature of the problem, etc and then use these data to support the selection of which techniques to use to leverage innovation and produce solutions according to the user's expectations (de Paula et al., 2020). Selecting such techniques seems like a challenging and complex endeavor.

Computing offers mechanisms for supporting DT techniques selection in software development. An example of a mechanism that assists professionals to select DT techniques is DTA4RE[1], a tool that recommends DT techniques to be used in software development (Souza et al., 2020). These recommendations are based on a static set of pre-defined questions (e.g., "Could you access users to validate your ideas?" or "Could you form a group to discuss and generate ideas?") that map which techniques can be used. Although DTA4RE consolidates and recommends techniques, it does not use any dynamic or intelligent behavior to do so, e.g. the nature of the problem or the user profile are not considered.

The Cynefin framework (Snowden and Boone, 2007), developed to support executives and leaders, has been used in several areas to help people to see things from new viewpoints, assimilate complex concepts, and address real-world problems and opportunities; including software development (Jantunen et al., 2019). The framework is organized into 3 universes: ordered, unordered, and disordered. Each universe is composed of different representations of the relationship between cause and effect (ordered and unordered), or no cause-effect representation (disordered). The ordered and unordered universes have 2 domains: simple and complicated, and complex and chaotic. In addition to the disordered universe, these domains can be summarized as follows: i. *simple*, a domain of best practices, where problems are well understood, and a solution requires minimal expertise; ii. *complicated*, a domain of good practices, where problems may be solved knowing the questions that need to be answered and how to obtain the answers; iii. *complex*, a domain of emergent solutions, where there are unknown problems and the final solution is only apparent discovered; iv. *chaotic*, a domain of novel solutions where the immediate priority is containment, and; v. *disorder*, a domain (or universe) in the center of the framework to represent that there is no knowledge on the scenario and the priority one is to move to a known domain.

3 RESEARCH METHOD

We conducted an exploratory focus group study (Morgan and Morgan, 1997) aiming to identify which techniques software engineers and designers working in software development would choose to solve certain hypothetical scenarios defined based on the Cynefin framework domains (Snowden and Boone, 2007), namely: simple, complicated, complex, and chaotic. Since DT is not an inherently approach from software development, we chose to explore the viewpoints of software engineers and designers working in software development. We had no preliminary assumptions of whether the distinct professionals would

[1] sites.google.com/site/dta4re/pagina-inicial

have followed distinguished selection rationals.

This study was attended by professionals with roles linked to software engineers such as Product Owners, Analysts, Developers, Testers, Technology Leaders, and Managers, and by professionals with roles linked to design of solutions, such as Designer themselves, Research Designers, UX/UI Designers, and Facilitators, who had to be fully allocated to a software team or contribute to one in order to participate in the study. We selected people from our network; mostly from the TECNOPUC (Technology Park located at PUCRS University) or previously connected to us through LinkedIn from previous studies or former co-workers[2]. Table 1 presents the profile of the 39 professionals, grouped by the sessions.

The sessions were organized by convenience, i.e., either to attend the participants' availability or to group them into a mixed and composite group of professionals. Before the session, each participant was invited to fill a questionnaire to indicate which DT techniques they have previously used and were familiar with. The list contained 24 techniques extracted from Souza et al. (2020). We collected this data to double-cross whether each participant has knowledge and experience of cited techniques during the session or whether they just complied with the discussion.

Each session was moderated by the first author, who has about 12 years of experience in software industry and almost 5 with DT's use in the role of a requirements analyst in 3 local companies. Each session lasted in average 1.5 hours and was comprised of 2 parts: a brief introduction of the session dynamic (10 minutes) to make the participants comfortable and level up their expectations as well as to present the Double Diamond DT model, and the focus group itself or the data collection activity (80 minutes).

The focus group itself reserved 20 minutes for the discussion of each of the 4 hypothetical scenarios: 4 minutes to present the scenario and clarify questions, up to 5 minutes for people to individually consider which techniques to use, 10 minutes for discussion, and 1 minute for the moderator to wrap-up. The overall, driven question upon each scenario's presentation was: How would you use DT to solve this scenario? I.e., which techniques would you use and for what purpose? Participants were asked to consider any DT techniques they could think of. We intentionally did not provide them with their responses from the previous questionnaire, but we let them free to consult the list or any other material. Besides, we asked them to position their selection within the 4 Double Diamond DT model working spaces. To help them recall the working spaces, we have the Double Diamond model illustrated in a projected slide.

Table 1: Focus Group Participants: the size of the organization (Org Size)- e.g. SE for small enterprise, ME for medium , LE for large and GLE for large global enterprise.

ID	Role	Org Size	Yrs at Org/DT	Previous Experience	Session
P01	Facilitator	SE	2/5	PO	1
P02	Facilitator	SE	3/3	PO, Dev, Mgr	1
P03	Lead UX/UI Designer	LE	13/4	PO, Dev, Designer, Mgr	1
P04	Developer Manager	GLE	8/1	PO, Dev, Mgr	1
P05	Manager	LE	5/3	Not mentioned	1
P06	Research Designer	LE	/10	Designer	1
P07	Experience Designer	SE	1/1	Not mentioned	1
P08	Experience Designer	ME	11/8	Dev, Designer, Mgr	2
P09	Marketing Analyst	SE	2/2	PO, Dev, Mgr	2
P10	Product Design Specialist	ME	1/4	Designer	2
P11	Support Analyst	GLE	5/2	Not mentioned	2
P12	UX Designer	LE	3/3	PO, Designer	2
P13	Software Developer	GLE	3/3	Dev	2
P14	Technology Director	SE	14/5	PO, Designer, Mgr	3
P15	Service Designer	ME	3/5	Designer	3
P16	Design Thinker	SE	4/4	PO, Mgr	3
P17	Product Manager	SE	1/3	Designer	3
P18	Technology Director	LE	2/10	PO, Dev, Mgr	4
P19	Design Evangelist	GLE	4/7	PO, Dev, Designer	4
P20	UX Research Analyst	GLE	1/3	PO	4
P21	Product Designer	LE	2/6	Designer	4
P22	Technology Leader	GLE	8/5	PO, Designer	4
P23	Business Analyst	LE	5/1	PO, Dev, Tester	4
P24	Product Designer	LE	4/10	PO, Designer, Mgr	5
P25	UX Leader	LE	3/3	Designer	5
P26	Innovation Director	SE	1/4	Designer	5
P27	Design Thinker	SE	2/2	PO, Mgr	5
P28	UX Consultant	SE	4/6	Designer, Mgr, Tester	5
P29	Service Designer	GLE	6/6	Designer	5
P30	UX Researcher	LE	3/6	Designer	6
P31	Lead UX Researcher	ME	1/2	PO, Designer, Mgr	6
P32	Innovation Facilitator	LE	13/1	PO, Dev, Designer, Mgr, Tester	6
P33	CEO	ME	6/4	Designer, Mgr	6
P34	UX Designer	LE	1/4	Designer	7
P35	Entrepreneur	GLE	3/4	PO, Dev, Mgr	7
P36	Scrum Master	LE	3/4	PO, Mgr, Tester	7
P37	Technology Consultant	SE	10/2	Not mentioned	7
P38	Business Analyst	LE	7/1	PO, Tester	7
P39	Entrepreneur	SE	4/8	PO, Designer, Mgr	7

Before starting the focus group, we presented the Cynefin framework and 4 scenarios (Pet-shop app, Transit App, E-Commerce App, and Recruitment process) for 2 professionals: a master student who has 7 years of experience in the industry and a Product Owner with 17 years of experience. Then, individually, we asked them to categorize each scenario for each Cynefin's domain. As a result, both professionals categorized the scenarios as shown in Figure 1.

The scenarios were presented from the simple until the chaotic domain, and they are defined as follows: In scenario 1 – Pet-Shop, this simple scenario introduced the wish to offer customers certain pet-shop store security to their dogs, and it was presented to the participants as follows: "We have received a client who wants to launch an application, envisioning security for pet owners. The pet-shop has a daycare, bathing, among others.". Since in simple domain is known and perceptible, predictable and reproducible, we understand that by presenting a client who knows the wished solution in a known type of business like pet-shop is easy to know how to proceed.

In Scenario 2 – Transit app, this complicated sce-

[2] Two of our co-authors work in the industry.

nario introduced an opportunity to find/discover a new launch in a transit app, and it was presented to the participants as follows: "In a competitive market, we have to launch a new feature in our transit app. 3 million users are using it. How do we proceed? We need to find a gap in the market.". Since in complicated scenarios the cause-effect is not perceptible, predictable and reproducible, we understand that by presenting a client who knows the wished solution, we believe that this kind of scenario is suitable because this the of business recently gained attention among people in many countries; however, many trends are happening around it.

Scenario 3 – E-Commerce, this complex scenario introduced an issue to solve about losing sales, and it was presented to the participants as follows "We have received a customer who is losing sales in his e-commerce. How do we proceed?". Since in complex scenarios the cause-effect is an unknown problem in an existing solution, we understand that by presenting an issue to solve, we believe that they need to think how to support the enterprise to react as soon as possible to avoid significant loss in their sales.

Scenario 4 – Recruitment process, this chaotic scenario introduced an issue to solve about the recruitment process, and it was presented to the participants as follows "We want to improve the recruitment process. How?". Since in chaotic scenarios the problems are inconsistent and imperceptible, we understand that by presenting recurring issue in many companies, we believe that they need to think how to structure what is happening in a complicated context that they do not have context.

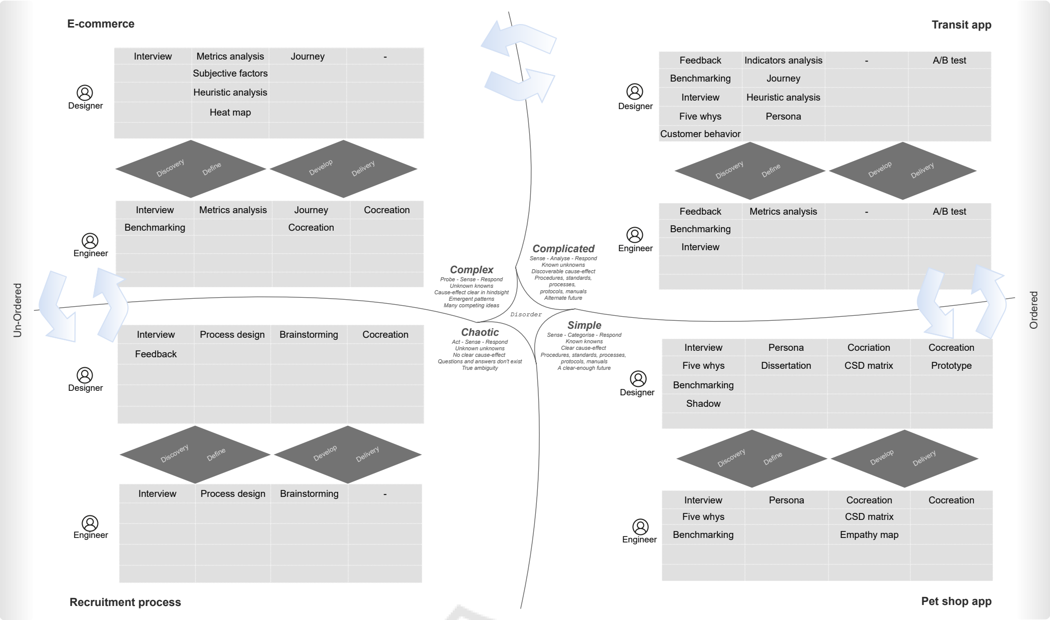
The disordered universe was not considered because if a problem reaches this universe, the professional needs to mitigate the uncertainties and changes for another domain. All sessions were voice and video recorded with previous consent from the participants. We used the Content Analysis technique (Krippendorff, 2018) to analyze all transcriptions. First, we transcribed all sessions after coding all questions (e.g., question 1 of session 1, ..., and 7 were analyzed before the next question). Then, we categorized all codes, grouping them by context.

4 RESULTS

The focus group aimed to identify which techniques software engineers and designers working in software development would choose to solve hypothetical scenarios, answering the following research question: *How do designers and software engineers select techniques to support the use of Design Thinking in software development?*. In the following, we present how the professionals chose the techniques for each scenario, considering the Double Diamond:

Scenario 1. Pet-Shop, Simple: In this scenario where a problem is known, perceptible, predictable, and reproducible, software engineers and designers selected the following DT techniques for each Double Diamond's working space: Discovery: the professionals mentioned interviews / questionnaires to collect the perspectives – *"making an interview with them [clients] or a small questionnaire to understand why the client wants more security (...)"* (P12). Also, they use "five whys" to comprehend the root of the needs – *"(...) "Five why" is suitable to understand why they want this solution, for example, Why do you want this App? Why do you believe that they want it? and other questions."* (P04) and benchmarking to analyze the competitors and market trends – *"You need to understand how the competitors solved this problem and how their clients are accepting it."* (P29). Define: The professionals cited persona to understand who has the need – *"To solve it [issue], we need to use personas to understand their needs. In this scenario, we have the shop owner as a persona, so we need to identify what he needs. Is it just earning money?"* (P02). Also, certainties, assumptions, and doubts (CSD) matrix can be used to evaluate the need – *"Validate if people want this App, then validate if it is a customer need, maybe the customer does not need it."* (P01). Develop: They mentioned the co-creation to evaluate and create the solution in a multidisciplinary team – *"What are their needs? Does the client want to improve his customer relationship? Does the client want to increase billing? I think I'll try doing a workshop with the people to check it."* (P33). Deliver: The professionals highlighted prototype to validate their solution – *"Maybe you can use a prototype to validate the idea with people."* (P06).

Scenario 2. Transit app, Complicated: In this scenario, the cause-and-effect is not immediately apparent to everyone. It also represents a trending topic requiring the professionals to seek solutions different from those already known in the market. For this scenario, software engineers and designers selected the following DT techniques for each Double Diamond's working space: Discovery: Feedback collection to look for opportunities/issues – *"Sometimes, the competitive differential is to know what the competitors are doing in a wrong way"* (P31). Also, interviews to explore more details in a deep way – *"(...) individual interviews with users, preferably the external users. It is also important to consider delighted people and people who are not satisfied with application usage."* (P27). Define: Metric analysis to ex-



E-commerce — Designer

Interview	Metrics analysis	Journey	-
	Subjective factors		
	Heuristic analysis		
	Heat map		

E-commerce — Engineer

Interview	Metrics analysis	Journey	Cocreation
Benchmarking		Cocreation	

Transit app — Designer

Feedback	Indicators analysis	-	A/B test
Benchmarking	Journey		
Interview	Heuristic analysis		
Five whys	Persona		
Customer behavior			

Transit app — Engineer

Feedback	Metrics analysis	-	A/B test
Benchmarking			
Interview			

Recruitment process — Designer

Interview	Process design	Brainstorming	Cocreation
Feedback			

Recruitment process — Engineer

Interview	Process design	Brainstorming	-

Pet shop app — Designer

Interview	Persona	Cocriation	Cocreation
Five whys	Dissertation	CSD matrix	Prototype
Benchmarking			
Shadow			

Pet shop app — Engineer

Interview	Persona	Cocreation	Cocreation
Five whys		CSD matrix	
Benchmarking		Empathy map	

Figure 1: Our Results.

tract and comprehend the data – *"We can confirm it using metric analysis, interpreting the data"* (P31). Also, Persona to evaluate the journey – *"You need to identify the persona to explore and work on his journey"* (P07). Develop: User journey was mentioned to identify opportunities – *"Doing a journey for different users, being possible to identify opportunities"* (P30). Delivery: A/B testing to understand the scenarios – *"A/B testing! Let us choose some customers, talk to them and verify if the solutions are working. In this case, we could have enough to validate it."* (P10).

Scenario 3. E-Commerce, Complex: In this scenario that the cause-and-effect is an unknown, flux and an unpredictable problem, requiring the professionals to avoid the sale loss, software engineers and designers selected the following DT techniques for each Double Diamond's working space: Delivery: This complex scenario represents a new problem in e-commerce. People also know this type of business because this form of commerce became popular in the last years. Also, there are tools to collect data and analyze customer behaviors. The professionals need to identify how to avoid the sale loss and what is happening in this context, then they can determine how to proceed with DT in this domain. The professionals proposed the following techniques: Metric analysis is used to understand the context – *"You could use metrics to understand where the people are, their gender, and their age. What is happening? It is important to understand customers' behavior six months ago or a year ago, such as your sales? When did your sales start to decrease? Which are the selling sectors impacted?"* (P02). Also, exploratory research to obtain more insights – *"You must combine metrics with exploratory research. This combination will bring the insights more in the big picture"* (P14) and benchmarking to analyze the competitors and market trends – *"You must know what is happening in the market."* (P02). Define: Co-creation workshop to understand different perspectives – *"It is important to discover the context of sales losing. One of the methods is a co-creation workshop to connect people, talk, and try to understand their different perspectives about the product and contexts."* (P28). Develop: User journey to explore the process and heuristic evaluation to identify the causes of falling sales – *"I think it has a simple solution, doing a heuristic analysis to understand it."* (P21). Also, a heat map to identify the application' flow – *"Identify where you are losing the sales, through a heat map."* (P21). Delivery: A/B testing to evaluate hypothesis – *"What was the loss quantity? you can set up A/B testing to identify why it is happening."* (P16).

Scenario 4. Recruitment process, Chaotic: In this scenario that there is no clear cause-and-effect relationship with high turbulence, software engineers and designers selected the following DT techniques considering each Double Diamond's working space: Discovery: It is important to understand the process, using interviews – *"First, you need to interview them, seeking to understand what happening is."* (P20). Define: Design process to comprehend the contact points among all stakeholders – *"It is a design situation, you must draw all processes and to map all* *contact points among them"* (P10). Develop: Journey map to identify the pain points – *"Once again, we return to the question: What is the journey? What is the current journey? What are the pains? Does it make sense to keep the current journey or build a new one? Where is the main pain?"* (P30). Also, a brainstorming to collect feedback – *"Using brainstorming, you can collect feedback to understand the points of contact among all involved people."* (P12) Delivery: Collect feedback to gather the perspectives about the solution – *"Focus on feedback collection. There are many perspectives, such as who is hiring, such as a manager, HR person, recruit, (...)"* (P12). There is also a disordered universe. We did not present this kind of scenario because there are many uncertainties about it.

5 DISCUSSION

Understanding professionals' choices when selecting DT techniques for software development can help obtain subsidies that support decision-making in this challenging and complex endeavor. Thus, in this study, different professionals were encouraged to present the techniques they chose to solve 4 hypothetical scenarios through a focus group.

When we presented the hypothetical scenarios, the professionals asked the following topics during the session' discussion: problem context, domain identification, user's needs comprehension, collected data understanding, business type categorization, innovation level, and competitive market aspects. Then, they highlighted these details as a criterion in their decision process for selection DT techniques. Also, the professionals emphasized that their decision making must take into account the scenario details.

For the pet shop app, a simple scenario in which a problem is known, perceptible, predictable, and reproducible; the professionals asked for details to understand the wished software solution and its needs. Given this, the professionals diverged and converged following the Double Diamond model, almost always selecting the same techniques, because the scenario was known by the professionals.

They have chosen techniques in all the Double Diamond working spaces because this kind of scenario is easier than others. It is also a predicted scenario, and the professionals distributed their techniques throughout the co-creation process without attempting in a specific working space. The co-creation is a collaborative development of something. In a simple domain, the professionals highlighted the need to assess the facts, categorize them, and apply a set of instruments, techniques, and methods to collect, evaluate, and validate the context and business.

For the transit app, a complicated scenario in which the cause-and-effect is not immediately apparent to everyone, the professionals highlighted the importance of understanding the consumer, and the competitors' to find a market gap. They chose the heuristic analysis to understand the journey. In addition, the professionals mentioned metrics analysis and A/B tests to evaluate and create hypotheses.

They have chosen more techniques in the problem space because they felt the need to explore the current context and new market trends to find a market gap, analyzing customer behavior and data. They mentioned they would use more techniques to discover the problem than work in the solution space. They did not mention any techniques in discovery space, since the problem understanding is the key to define any technique in solution space; they need to capture the real needs. In the complicated domain, it is necessary expertise because emerging concepts are necessary to explore the cause and effect, feeling the consumers, exploring the market, and investigating various options. There is a possibility that this kind of domain does not evolve; remaining undefined, perhaps it can become an unordered domain.

For the e-commerce app, a complex scenario in which the cause-and-effect is an unknown; all professionals asked many questions, because there is a high uncertainty in this scenario. They highlighted that if there were metrics, they could use these to analyze the loss of sales. Also, the subjective factors analysis and heat maps to understand what could be happening can support their decisions. The professionals have chosen more techniques in problem converging working space because this kind of scenario is required to analyze the journey to find users' real problems. It is important to extract the right answers in the complex domain because the flow is unpredictable. A solution can emerge and then this domain can become ordered. It is essential to respect the default domain to set the scenario and allow insights to emerge.

For the recruitment process, a chaotic scenario in which there is no clear cause-and-effect relationship with high turbulence; the professionals have known this kind of problem, however, they mentioned many uncertainties in the problem and its solution. The professionals mentioned the importance of knowing how the current process is through feedback collection and interviews and a process view with its pain points to think about the problems and its solution.

The professionals had many difficulties with problem and solution space because this kind of scenario has a high level of uncertainty and requires to explore more the needs to find the problem. They mentioned it was hard to indicate or select one technique to use in this domain. However, the professionals need expertise in the chaotic domain because they need to know how to proceed with the DT to reduce uncertainties. It is crucial to act quickly in the chaotic domain. Since it is not always possible to determine cause and effect, the search for the right answer could result in waste of time. It is also important to have attention in emerge hypothesis to evaluate the scenario in domain. They mentioned that the solution is a consequence of the problems, regardless of the techniques used.

The results also show that techniques' selection is related to rationale linked to the professional's certainty level. Simple and complicated (pet shop and transit app) domains are in an ordered universe. We perceived Designers and Engineers were more certain to select DT techniques. In these domains, it is possible to be more confident to make appropriate decisions, apply experience based on previews cases, establish structures to process, and systematically measure the findings using clear objectives.

However, complex and chaotic (e-commerce and recruitment process) domains are in an unordered universe, where there is no immediately apparent relationship between cause and effect. The professionals need more details to think in these domains. The questions were about how to explore the scenarios, through data, clear objectives, and context. Then, the first strategy is to discover the problem in its essence. These domains need more expertise to mediate DT because many opportunities for innovation can emerge. This situation needs knowledge, preview experiences, and an understanding composed by a diversity of opinions and insights.

Cynefin helped us to understand better how the software industry professionals made their decisions regarding the techniques' selection for the different domains. Also, the scenarios can move among the domains, depending on the level of certain, represented by the blue arrow in Figure 1. We identified that spaces could not determine the techniques used through DT (discover, define, develop, and deliver).

In this study, we noticed that the techniques could be applied in different working spaces (discover, define, develop, and deliver). It depends on the problem context, domain identification, user's needs comprehension, collected data understanding, business type categorization, innovation level, and competitive market aspects (based on presented hypothetical scenarios). Thus, it is worth mentioning that other applications categorized in the Cynefin domains may require the use of other techniques.

6 RELATED WORK

Studies discuss how to use DT in software development; they focus on understanding an issue or phenomenon. Our study conducted seven focus groups to identify how Designers and Engineers are adopting DT, understanding their decision making in selecting DT techniques for solving hypothetical scenarios. We have not found studies in this line.

However, other studies have discussed the use of DT in software development and techniques and purposes. Canedo et al. (2020) highlighted that DT practice in requirement elicitation could deliver product quality to the end-user since DT techniques could prevent failure in understanding requirements. This study presents how the professionals use the DT tools and the techniques more used by them (Canedo et al., 2020). Rauth et al. (2014) discussed how DT is used in large organizations, they identified that DT can be used to develop new ideas, a mindset, or a combination of mindset and methods (Rauth et al., 2014).

In line with these findings, we identified similar perspectives in our study too. Prasad et al. (2018) derived a framework from achieving customer satisfaction through the adoption of DT in agile-base projects, exploring how effectively the professionals use DT practices with the agile process (Prasad et al., 2018). In our study, we identified some characteristics mentioned by Prasad et al., like techniques used during DT and the perceived value when a project is delivered. Also, Dobrigkeit et al. (2019) claim the professionals' experience and how much they know about DT may influence how they consider DT usage in a software project (Dobrigkeit and de Paula, 2019).

7 FINAL CONSIDERATIONS

In this paper, we aimed to explore how software engineers and designers make the selection of DT techniques for different software development scenarios. We conducted seven focus group sessions, presenting four scenarios based on the Cynefin framework. Cynefin combined with the Double Diamond DT model helped us to comprehend in a better way how the professionals made their decisions for the different domains. We noticed that depending on the problem domain (e.g. simple and complex), the Designers and Engineers couldn't determine techniques by working spaces. The choice of techniques depends on the problem context, user's needs comprehension, characteristics, challenges, collected data, business type, innovation exploration, innovation level, feasibility and market competitive aspects.

Focus groups (Morgan and Morgan, 1997) has typical limitations of qualitative studies, mainly in the generalization of results (Singer et al., 2008). We counted on the cooperation of thirty-nine professionals who used DT for software development, which influenced the final results' generalization. Even if the generalization is not possible, these data are valid and complemented with other studies.

In our future work, through empirical studies in real case projects, we intend to explore in depth the strategies of professionals' decision-making process for selecting DT techniques. We aim to discover if the decisions are empirical, intuitive, rational, or heuristic-based, and to propose a computational mechanism capable of supporting these decisions.

ACKNOWLEDGEMENTS

The authors would like to thank all professionals. Also, we like to thank the financial support from the CAPES (process 175956/2013 and partially financed by code 001, CNPq (process 311494/2017-0), and FAPEAM (process 062.00150/2020).

REFERENCES

Brenner, W. and Uebernickel, F. (2016). *Design Thinking for Innovation: Research and Practice*, pages 3–21. Springer.

Brenner, W., Uebernickel, F., and Abrell, T. (2016). *Design Thinking as Mindset, Process, and Toolbox*, pages 3–21. Springer.

Canedo, E., Pergentino, A., Calazans, A., Almeida, F., Costa, P., and Lima, F. (2020). Design Thinking Use in Agile Software Projects: Software Developers' Perception. In *Proc. Int'l Conference on Enterprise Info Systems*, pages 217–224, Online Conf. Scite Press.

Carlgren, L., Rauth, I., and Elmquist, M. (2016). Framing Design Thinking: The Concept in Idea and Enactment. *Creativity and Innovation Mngt*, pages 38–57.

Chasanidou, D., Gasparini, A., and Lee, E. (2015). Design Thinking Methods and Tools for Innovation. In *Proc. Int'l Conference on Design, User Experience, and Usability*, pages 12–23, Los Angeles, USA. Springer.

Council, D. (2020). The design process: What is the double diamond? Available on designcouncil.org.uk. Accessed in January, 2020.

de Paula, T. R., Santana Amancio, T., and Nonato Flores, J. A. (2020). Design Thinking in Industry. *IEEE Software*, 37(2):49–51.

Dobrigkeit, F. and de Paula, D. (2019). Design Thinking in Practice: Understanding Manifestations of Design Thinking in Software Engineering. In *Proc. European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 1059–1069, Tallinn, Estonia. ACM.

Dobrigkeit, F., de Paula, D., et al. (2017). The Best of Three Worlds - The Creation of InnoDev a Software Development Approach that Integrates Design Thinking, Scrum and Lean Startup. In *Proc. Int'l Conference on Engineering Design*, pages 319–328, Vancouver, Canada. Design Society.

Hehn, J., Mendez, D., Uebernickel, F., Brenner, W., and Broy, M. (2020). On Integrating Design Thinking for Human-Centered Requirements Engineering. *IEEE Software*, 37(2):25–31.

Hehn, J. and Uebernickel, F. (2018). The Use of Design Thinking for Requirements Engineering: An Ongoing Case Study in the Field of Innovative Software-Intensive Systems. In *Proc. of the Int'l Requirements Eng. Conf.*, pages 400–405, Banff, Canada. IEEE.

Jantunen, S., Dumdum, R., and Gause, D. (2019). Towards New Requirements Engineering Competencies. In *Proceedings of the International Workshop on Cooperative and Human Aspects of Software Engineering*, pages 131–134, Montréal, Canada. IEEE.

Krippendorff, K. (2018). *Content Analysis: An Introduction to Its Methodology*. Sage, New York, USA.

Lindberg, T., Meinel, C., and Wagner, R. (2011). *Design Thinking: A Fruitful Concept for IT Development?*, pages 3–18. Springer Berlin, Germany.

Morgan, D. and Morgan, D. (1997). *Focus Groups as Qualitative Research*. A Sage university paper. SAGE.

Park, H. and McKilligan, S. (2018). A systematic literature review for human-computer interaction and design thinking process integration. In Marcus, A. and Wang, W., editors, *Design, User Experience, and Usability: Theory and Practice*, pages 725–740. Springer.

Prasad, W. R., Perera, G., Padmini, K. J., and Bandara, H. D. (2018). Adopting Design Thinking Practices to Satisfy Customer Expectations in Agile Practices: A Case from Sri Lankan Software Development Industry. In *Proc. of the Moratuwa Engineering Research Conf.*, pages 471–476, Moratuwa, Sri Lanka. IEEE.

Rauth, I., Carlgren, L., and Elmquist, M. (2014). *Making It Happen: Legitimizing Design Thinking in Large Organizations*. John Wiley & Sons.

Singer, J., Sim, S. E., and Lethbridge, T. C. (2008). *Software Engineering Data Collection for Field Studies*, pages 9–34. Springer.

Snowden, D. and Boone, M. (2007). A Leader's Framework for Decision Making. *Harvard Business Review*, 85:68–76, 149.

Souza, A., Ferreira, B., Valentim, N., Correa, L., Marczak, S., and Conte, T. (2020). Supporting the Teaching of Design Thinking Techniques for Requirements Elicitation Through a Recommendation Tool. *IET Software*, 14(6):693–701.

Tschimmel, K. (2012). Design Thinking as an Effective Toolkit for Innovation. In *Proc. Conference: Action for Innovation from Experience*, pages 1–20, Barcelona, Spain.

Weigel, L. (2015). *Design Thinking to Bridge Research and Concept Design*, pages 59–70. John Wiley & Sons.