The design risks framework: Understanding metacognition for iteration

Spencer E. Carlson, Daniel G. Rees Lewis, Leesha V. Maliakal, Elizabeth M. Gerber and *Matthew W. Easterday*, Segal Design Institute, Northwestern University, Evanston, IL, USA

Iteration is an important design process that novice designers struggle to follow. However, iteration is difficult to coach because we do not understand the underlying metacognitive knowledge required for effective iteration. We developed the Design Risks Framework, which helps researchers to identify the knowledge underlying three metacognitive processes that control iteration: focusing attention on key areas of the project, identifying project risks, and choosing iterative strategies to mitigate risks. We tested the framework over a 6-week period with 5 novice design teams and found that novices seemed to lack metacognitive knowledge of 49 criteria for identifying project risks. By using this framework to diagnose knowledge gaps and design coaching interventions, educators and managers can improve how novice designers iterate in design projects.

Keywords: design metacognition, design cognition, design knowledge, design process, reasoning

Iteration is an important design process that is challenging for novice designers (Atman, Chimka, Bursic, & Nachtmann, 1999; Atman et al., 2007; Crismond & Adams, 2012). Here, we define iteration as "a purposeful progression through stages of the design process," in which designers engage cognitive processes such as seeking information, defining the problem, and modifying a solution as needed to refine one's understanding of the problem and advance a solution (Adams & Atman, 1999; Adams, Turns, & Atman, 2003, p. 286). Iteration often involves repetition and transitioning between activities, but these processes alone do not fully capture the abilities that novice designers must learn to iterate effectively ("we don't want to simply direct students to 'transition more frequently!'"; Atman & Adams, 2000, p. 2). For this purpose, it helps to define iteration as the goal-directed process in which designers transition between, or repeat, design activities *as needed* to progress from an ill-defined problem to a design solution (Adams et al., 2003).

Corresponding author:
Spencer E. Carlson
s.c@u.northwestern.edu

Recent research in this journal argues that iteration results from designers' metacognitive monitoring and control of design process and calls for a deeper

www.elsevier.com/locate/destud
0142-694X *Design Studies* **70** (2020) 100961
https://doi.org/10.1016/j.destud.2020.100961

understanding of this metacognition (Ball & Christensen, 2019). Unfortunately, we do not yet understand the metacognitive knowledge that drives effective iteration or what metacognitive knowledge novice designers lack that makes iteration challenging. This creates a difficult practical problem for design coaches: how to coach novice designers in iteration when it is unclear why they are struggling in the first place?

To answer the call to improve understanding of design metacognition (Ball & Christensen, 2019), we propose the Design Risks Framework (DRF): a novel theoretical framework which highlights the metacognitive processes that control iteration so that researchers can identify the specific knowledge structures underlying those processes.

The Design Risks Framework enables researchers to explain a designer's (or design team's) iteration in terms of the application or absence of specific metacognitive knowledge. We demonstrate this capability in two ways. First, we review extant studies of iteration in architecture, software design, and industrial design to generate novel hypotheses about the metacognitive knowledge that designers applied to iterate in these studies. Second, we present our analysis of 5 case studies of novice design teams planning over the course of 6-week-long design projects. We used the framework to diagnose the specific metacognitive knowledge these particular design teams failed to apply in specific planning decisions—in this case, 49 criteria for identifying project risks. In the discussion, we summarize follow-up work, already published, in which these insights enabled us to design more effective coaching for the novice teams (Rees et al., 2018). This affirms the practical utility of the framework for informing coaching.

1 Iteration

Iteration is the strategic management of the design process to refine one's understanding of the problem and advance a solution (Adams & Atman, 1999; Adams et al., 2003). As such, iteration is a defining characteristic of effective design processes (Adams et al., 2003; Crismond & Adams, 2012; Guindon, 1990; Jin & Chusilp, 2006; Schön, 1983; Schön & Wiggins, 1992; Wynn & Eckert, 2017).

Prior research on iteration focuses on understanding the process of iteration and how ideas evolve in this process (Adams & Atman, 1999; Adams et al., 2003; Atman & Adams, 2000; Atman et al., 1999, 2007; Chusilp & Jin, 2006; Guindon, 1990; Jin & Chusilp, 2006; Wynn & Eckert, 2017). Much prior research on iteration has used protocol studies—in which a designer talks aloud while solving a design problem—to document the iterative strategies that designers employ to make progress in design problems across diverse domains (Adams & Atman, 1999; Chusilp & Jin, 2006; Dorst & Cross, 2001;

Gero & Mc Neill, 1998; Guindon, 1990; Jin & Chusilp, 2006; Schön, 1983; Wynn & Eckert, 2017). For example, Guindon (1990) documents how software designers iterated to explore highly ambiguous aspects of the solution, while Dorst and Cross (2001) document how industrial designers iterated to maintain the coherence between an evolving solution concept and problem definition.

A designer's expertise can largely be conceptualized in terms of their ability to apply these iterative strategies, which can be specified as useful sequences of transitions between cognitive processes such as accessing information, clarifying information, evaluating the solution, changing the problem definition, changing the solution, and selecting a solution (Adams & Atman, 1999; Adams et al., 2003). In protocol studies comparing novice designers with more experienced designers, the experienced designers were more likely to spend time iterating, use more iterations, transition through more steps in an iteration, use iterations that coupled problem scoping with solution development, and express knowledge of iterative strategies (Adams et al., 2003; Atman et al., 1999, 2007; Crismond & Adams, 2012).

Prior research reveals that effective iteration requires the ability to draw from a repertoire of strategies for moving among design activities to advance the design process. However, we do not yet understand the metacognitive knowledge structures that enable effective iteration or what metacognitive knowledge novice designers lack that makes iteration challenging.

2 Design metacognition

While metacognition is central to how designers manage their iterations (Adams et al., 2003; Schön, 1983), it remains an underexplored topic in design cognition (Ball & Christensen, 2019). In a recent review of research on design cognition, Ball and Christensen (2019) summarize research on the many cognitive strategies that designers use in the iterative design process. These strategies include framing, analogies, and mental simulation (Ball & Christensen, 2019). They argue that "a major concern is to understand how designers *select* and *deploy* design strategies to navigate through the inherent 'uncertainty' that pervades real-world design problems because of their 'ill-defined' (Simon, 1973) or 'wicked' (Rittel & Webber, 1973) nature" (Ball & Christensen, 2019, p. 36).

Consistent with decades of research in cognitive science (e.g., Greene & Azevedo, 2007; Winne & Azevedo, 2014), Ball and Christensen (2019) point out that understanding how designers select and deploy these strategies is fundamentally a question about metacognition—the processes that continually monitor and control cognition to choose the right strategies for an evolving task (Ackerman & Thompson, 2017; Ball & Christensen, 2019; Winne & Azevedo, 2014). They conclude that:

> Metacognition has been an overlooked aspect of information processing in design, despite being central to understanding every aspect of a designer's ongoing activity in progressing from an initial ill-defined design problem to an all-encompassing design solution. We argue that viewing design cognition through a metacognitive lens is critical to advancing an integrated understanding of the way in which strategies change over the design time-course in response to fluctuating feelings of uncertainty. That said, understanding the metacognitive basis of the dynamic aspects of design cognition is in its infancy, with many core questions still needing to be addressed (Ball & Christensen, 2019, p. 52).

We begin to answer this call by focusing on the source of designers' uncertainty, which Ball and Christensen (2019) propose triggers designers' efforts to metacognitively control the iterative design process. What makes designers uncertain? How do designers know there are important gaps in their understanding of the problem and solution so they can choose effective design strategies? We expect that the ability to feel this uncertainty is something that must be learned rather than a universal human trait. Research from the learning sciences shows that novices in a domain initially display overconfidence in their ideas and must learn to assess their ideas more accurately before they can reach mastery (Ambrose, Bridges, diPietro, Lovett, & Norman, 2010).

Unfortunately, we do not understand the metacognitive knowledge that enables experienced designers to recognize uncertainty, which means we do not know what knowledge novice designers must learn to recognize uncertainty so they can metacognitively control their design iterations. To begin addressing this problem, we propose the Design Risks Framework: a novel theoretical framework which highlights the metacognitive processes that control iteration so that researchers can identify the specific knowledge structures underlying those processes in particular design domains.

3 *Design Risks Framework*

The Design Risks Framework identifies three metacognitive processes for monitoring and controlling iteration: *focusing attention* on key areas of the project, *identifying risks* in those areas, and *choosing strategies* to mitigate those risks. Each process relies on different underlying knowledge structures (see Figure. 1). Note that these processes can become automated with experience, meaning that designers may or may not be actively aware of these processes or the tacit knowledge involved. By using the framework to analyse this knowledge, researchers can understand the knowledge that enables effective iteration and develop coaching interventions that support this knowledge to improve how novices iterate.

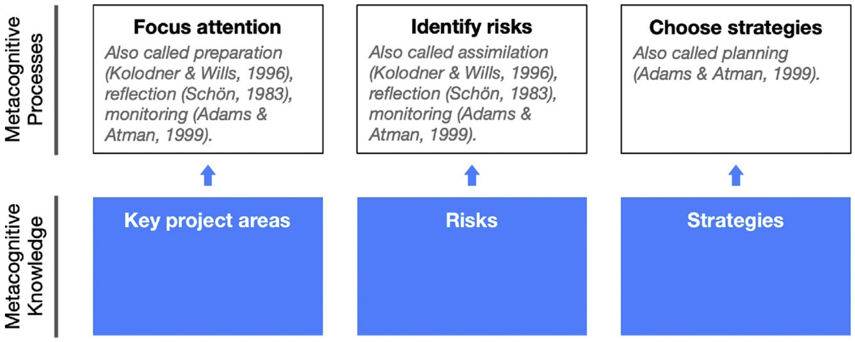

Figure 1 The Design Risks Framework identifies three metacognitive processes—focusing attention, identifying risks, and choosing strategies—that align with processes in existing models. The framework builds on these existing models by identifying the underlying metacognitive knowledge involved in each process

We developed the framework by reviewing existing models of design cognition that seem to discuss elements of metacognition (described below), identifying the minimum set of logically necessary processes for the sake of parsimony, and inferring the kind of knowledge that might be used in each process by incorporating insights from design practice and cognitive science research.

3.1 Focusing attention

To monitor and control iteration, designers first focus their attention on their knowledge of key areas in their project. To do this, designers rely on schemata of problems and solutions in a given design domain. Schemata are knowledge structures with slots that specify the key components of a situation or design solution, which can be filled with data from a particular design situation to make sense of it (Rumelhart, 1980). For example, some designers schematize user needs as having three slots: a "job" or task the user must do, a "pain" or reason the task is difficult, and a "gain" or benefit that could be gotten by making the task easier (Osterwalder, Pigneur, Bernarda, & Smith, 2014). Schemata can vary widely in their level of abstraction and content (Rumelhart, 1980), so designers might have broad schemata for understanding design tasks (such as a schema with slots for the problem, solution, and key stakeholders), or finer-grained schemata for understanding specific elements of the situation (such as a schema for the user's need with slots for job, pain, and gain), or subcomponents of a solution (such as a schema for designing a personal computer with slots for the motherboard, power supply, graphics card, and so on). The designers use these schemata to understand the issues in a particular project and determine where to focus their attention when reflecting on internal knowledge about the project or when reviewing an external representation of the problem or solution.

Because many domains can be implicated in a single design problem (Schön & Wiggins, 1992), each designer may use a combination of schemata—based on their background knowledge of various domains—to understand the different issues in a particular project. For example, when designing a personal computer it might help to have a schema for user needs to ensure the computer is desirable, a schema for the technical components to ensure the computer is functional, and a schema for the manufacturing process to ensure the computer can be produced at scale.

Focusing attention is a *meta*cognitive process because metacognition is processing information about cognition (Winne & Azevedo, 2014), and focusing attention involves the designer reviewing internal and external representations (information) of their *knowledge* of the project (cognition).

Some existing models of iteration include a version of this process, such as preparation (Kolodner & Wills, 1996), while other models include focusing attention as one element of a broader process, such as monitoring (Adams & Atman, 1999; Ball & Christensen, 2019) or reflection (Schön, 1983). The Design Risks Framework further extends these domain-general process models of iteration by identifying the underlying knowledge structures driving this process. Specifically, the framework directs researchers to look for the schemata that guide where designers focus their attention.

3.2 *Identifying risks*

Second, designers identify project risks—specific uncertainties that could lead to failure. We draw this definition of risk from design practice (e.g., Blank & Dorf, 2012; Constable, 2014), not from research on risk management, which uses the same word for a different concept. For example, if a toy designer has incomplete or incorrect knowledge about national consumer safety standards, there is a risk of designing toys that violate these standards, harming children, and placing the designer in legal jeopardy. An experienced toy designer has knowledge of this risk, including specific criteria for evaluating their knowledge about safety standards to identify whether this risk is present. Expert designers have metacognitive knowledge of many risks in the domains where they have experience, enabling them to identify many aspects of their knowledge of the project (i.e., internal and external representations of the problem and solution) that need to be refined and tested.

Some existing models of iteration seem to include a version of this process, such as assimilation (Kolodner & Wills, 1996), while other models seem to include identifying risks as one element of a broader process, such as monitoring (Adams & Atman, 1999; Ball & Christensen, 2019) or reflection (Schön, 1983). Building on these existing models that indicate the importance of identifying risks as a general metacognitive process, in DRF we contribute a framework that directs researchers to look for designers' knowledge of how to recognize specific risks, and how those risks threaten the project.

3.3 Choosing strategies

Third, designers choose iterative strategies to mitigate the risks they have identified. For example, much has been made of how designers use a repertoire of representations to understand different aspects of the problem and solution (Schön, 1983; Schön & Wiggins, 1992). Choosing to use a certain representation is one example of choosing a strategy. Other examples include choosing to interview users or choosing to build and test a low-fidelity prototype. For example, to mitigate the risk of designing a toy that violates the child safety standard for avoiding choking hazards, a designer might choose the strategy of building a low-fidelity prototype to measure whether it is larger than an average child's windpipe, thereby reducing uncertainty that could lead to failure.

This process requires designers to have metacognitive knowledge of many iterative strategies, and how those strategies enable one to clarify, organize, and test their knowledge of a project in order to mitigate certain risks.

Some existing models of iteration include a version of this process, such as planning (Adams & Atman, 1999), while other models imply the existence of this process by including a subsequent process of *implementing* strategies, such as action (Schön, 1983), and strategic control (Ball & Christensen, 2019; Kolodner & Wills, 1996). Building on these existing models that indicate the importance of choosing strategies as a general metacognitive process, the Design Risks Framework directs researchers to look for designers' knowledge of specific strategies and how those strategies can help to mitigate specific risks.

3.4 The Design Risks Framework across disciplines

To understand how this framework helps researchers to explain iteration in terms of the specific metacognitive knowledge underlying these three processes, consider the following examples from architecture, industrial design, and software design. We use the DRF to make new conjectures—initial theory building claims that can be validated in future work (Glaser & Strauss, 1967)—about the metacognitive processes and metacognitive knowledge that seemed to enable these designers to iterate effectively. Note that these conjectures are our own interpretation of the examples, rather than that of the original researchers or the designers themselves.

First, take Schön's (1983) classic example of Quist and Petra, two architects discussing how to design a school on the slant of a hill. Viewing this episode through the Design Risks Framework, both Quist and Petra seemed to ***focus***

their attention on the fit between the buildings and the site. Petra seemed to ***identify a risk*** that she did not know how to make the buildings look aesthetically coherent with the irregular hillside. This knowledge gap created a risk that Petra's building would not be commissioned or enjoyed, which could threaten her reputation and livelihood. In an attempt to mitigate this risk, she seemed to ***choose a strategy*** of sketching to explore one potential solution—butting the buildings up against the contours of the landscape. Reflecting on Petra's sketches, Petra and Quist ***focused their attention*** on the fit between buildings and site once more, and they still ***identified the risk*** of aesthetic incoherence—Petra's model of the solution idea still did not align with her model of the geography. In response, Quist ***chose a strategy*** of conducting a new sketching experiment to explore an alternative solution—conceiving a geometric shape and imposing it on the landscape, rather than trying to design a shape that fits within the landscape.

Using the Design Risks Framework, we focus specifically on the metacognition driving these iterative cycles. To be sure, the architects also had craft knowledge that influenced their cognition *within* phases of the design process—for example, this knowledge influenced the kind of solutions they conceived (for Petra, fitting the buildings within the landscape; for Quist, imposing geometric discipline on the landscape). A typical reading of their protocol would focus on this knowledge, because it is the explicit content of their discussion. This is important, but it *does not help us to explain iteration*. In contrast, the DRF raises questions about how these architects knew to iterate *between* design phases (conceiving a potential solution, conducting sketching experiments to test it, etc.), and helps researchers to explain this iteration by looking for the *metacognitive* knowledge governing how the architects focused their attention, identified risks, and chose iterative strategies. In this case, the DRF enabled us to explain the designers' iteration by saying that they had metacognitive knowledge including (a) schemata defining key aspects of the problem, which included a slot for the fit between the building and the site; (b) the risk of aesthetic incoherence; and (c) the strategy of sketching experiments, and that they used this knowledge to exercise metacognitive control over their design iterations.

To see that researchers can apply this framework across different design domains, consider a second example from Guindon's (1990) protocol study of experienced software designers tackling a novel problem of designing software to control a lift. Interpreted through the DRF, we can say that the designers in this study seemed to ***focus their attention*** in the problem space on certain scenarios (e.g., imagining the lift moving from floor to floor). Then, they seemed to ***identify risks*** created by gaps in their knowledge of the problem and solution. For example, one designer noted a gap in his knowledge of the solution—"what if you press up at the floor but, once in the lift, you press a down button?"—which implied a risk that the software might fail to reconcile these conflicting inputs and take passengers in the wrong direction (Guindon, 1990, p. 330). To mitigate this risk, the designer seemed to ***choose a strategy*** of conceiving and evaluating a partial solution to deal with this button pressing issue.

Third, consider the protocol that Dorst and Cross (2001) collected from an experienced industrial designer who was designing a garbage disposal system for a train. Through the lens of the Design Risks Framework, it seems that, at one point, the designer ***focused his attention*** on the train's toilet system, which he recognized as the current solution for disposing of waste on the train, and therefore important to understand. At that moment, he ***identified the risk*** that he did not know how the toilet system worked, and therefore his garbage disposal system might simply be duplicating a pre-existing function of the toilets. To mitigate this risk, he ***chose the strategy*** of gathering information about how the toilets disposed of sewage (which, given this was a laboratory study, involved asking the experimenters for more information). This strategy allowed him to learn that the train currently dumped sewage directly onto the tracks below. He then redefined the problem around developing a better system to handle both trash and sewage.

In each of these three studies, researchers analysed these protocols to understand the process of iteration and how ideas evolve in that process—but not the metacognition that controls iteration (Dorst & Cross, 2001; Guindon, 1990; Schön, 1983). By applying the Design Risks Framework to these examples, we can identify specific metacognitive knowledge that played a critical role in directing each iteration. First, the designers knew to focus their attention on specific aspects of their projects: the aesthetic fit between building and site, the behaviour of the lift software when passengers change their minds, and the schematic of the toilet system. Second, the designers had knowledge of key risks: the risk of not knowing how to achieve aesthetic coherence, the risk of not knowing how the lift software would deal with conflicting inputs, and the risk of wastefully duplicating the toilet system. Third, the designers knew strategies to mitigate these risks: by conducting sketching experiments to test high-level solutions, by conceiving a new software component, and by requesting information about the toilet system.

These examples demonstrate how researchers can apply the framework to make conjectures about the knowledge that enables designers' iterative, metacognitive control of the design process. However, this framework may also illuminate why *novice* designers struggle to iterate in a way that design instructors view as productive—by highlighting differences between where novices and instructors focus their attention in the problem and solution, differences in what novices and instructors identify as important risks, and differences in what strategies they choose. By highlighting these metacognitive processes, the Design Risks Framework may provide a framework for diagnosing key metacognitive knowledge that novice designers lack, providing better guidance for coaching iteration.

4 Method

To develop empirically grounded conjectures about gaps in novice designers' metacognitive knowledge, we applied the Design Risks Framework to 5 case studies of novice design teams completing full-time, 6-week-long design projects. We focused on the teams' weekly iteration planning meetings. Because the designers externalized their reasoning to communicate within their teams, we were able to observe where they focused their attention, what risks they identified, and what iterative strategies they chose. Beginning with these three categories from the DRF, we iteratively elaborated a coding scheme of key metacognitive knowledge that was absent in the design teams' reasoning, hindering their ability to plan effective iterations.

4.1 Context

For this study, we investigated 5 teams (4–5 designers each) participating in a 6-week extra-curricular undergraduate summer program at a university design institute. Teams worked 36 h/week on campus. Each team worked with a local client to design products and services to address a real-world challenge. The teams' challenges were: improving airport accessibility for autistic travellers, reducing air travel related wheelchair breakages, improving accommodations for people with dementia, increasing first responder support for youth mental health in an exceptionally violent neighbourhood, and reducing teen depression. This study was part of an ongoing decade-long research project on improving support for novice designers (Rees et al., 2018; Rees Lewis, Gerber, Carlson, & Easterday, 2019; Rees Lewis, 2018).

The study involved 21 undergraduates from 18 to 22 years old (12F, 10M). Some participants had experience working on design projects in their undergraduate coursework, while others were working on design projects for the first time. Participants were majoring or double-majoring in engineering (15), natural sciences (3), social sciences (3), art (3) and journalism (3), and included 4 first-, 12 second-, and 5 third-year students. These novice designers may or may not be representative of other novice designers; this is not a problem for our analysis because we did not intend to generalize our empirical findings about these designers to other designers (Small, 2009). Instead, the purpose of this analysis was to develop conjectures about the metacognitive knowledge of *particular* novice designers, using the Design Risks Framework.

Each week, the design teams completed a 2-h iteration planning workshop led by members of the research team. In these workshops, teams discussed their plans with the aid of 2 poster-sized templates called the Design Canvas and Iteration Plan. The Design Canvas was a template for representing the design problem and solution (Figure 2), and the Iteration Plan was a template for representing an iterative strategy (Figure 3). These tools, along with the structure of the workshop, guided teams to externalize their reasoning in discussion and record it on the tools as they moved through the metacognitive processes of planning an iteration.

In this analysis, we assessed the teams' metacognition, as described below, to articulate our own tacit metacognitive knowledge as experienced designers and design instructors. To perform this kind of analysis, researchers must have design expertise. The members of our research team have over 50 combined years of experience designing educational technologies, online communities, social services, consumer software, and toys, across the public sector, private sector, and academic design-based research. All members of our research team are either university design faculty or doctoral students in design-intensive programs, and work in an interdisciplinary, design-intensive research lab. Across the team, we have 3 doctoral, 4 master's, and 4 bachelor's degrees that involved formal training in design.

4.2 Data collection and analysis

We used the grounded theory methodology to take an exploratory, qualitative approach to analysing these case studies. The grounded theory research methodology (Corbin & Strauss, 2014) is used to *generate*, rather than *validate*, new theoretical models grounded in data that explain some phenomenon (in this case, that phenomenon is design metacognition).

Data collection focused on teams' reasoning during the weekly planning workshops. Each week, one researcher observed each team and recorded field notes of their discussion. We also digitally captured images of the Design Canvas and Iteration Plan at the end of the workshop. Thus, we followed Winne's (2010) suggestion to develop conjectures about metacognition by analysing observable representations of metacognitive events (in this case, speech and writing).

As they collected data, each researcher tagged data with low-level codes and grouped codes into categories according to grounded theory coding practices (Charmaz, 2003; Corbin & Strauss, 2014). The categories in our coding scheme represent a schema of risky areas of teams' projects (Table 1), and the codes in our coding scheme represent specific criteria for identifying risks in each of those areas (Table 2).

As themes emerged while coding data, each researcher wrote *analytic memos* summarizing the key project areas and risks the novices seemed to be ignoring, along with evidence from observations and images of teams' Design Canvases. Writing analytic memos is a grounded theory strategy in which researchers

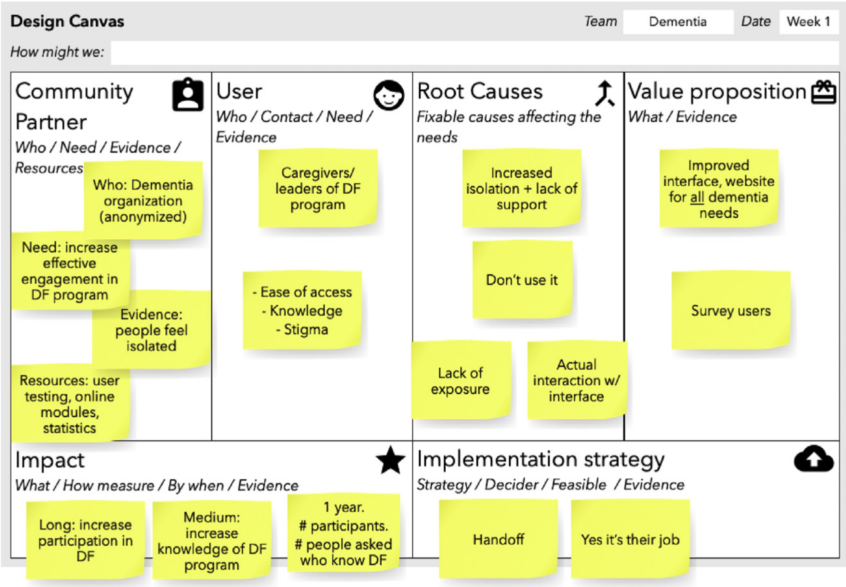

Figure 2 During the weekly planning workshop, design teams used this Design Canvas template, which we provided, to represent their knowledge of key areas in their project. This example comes from the dementia team in week 1 (reproduced for readability)

begin assembling coded data into tentative conjectures to explain the phenomenon under investigation, enabling researchers to reflect on their understanding and decide what additional data are needed (Charmaz, 2003; Corbin & Strauss, 2014). Next, we gathered as a team to share and discuss our individual analytic memos. We held these team meetings at least once a week, following data collection and individual analysis. We used these meetings to synthesize the insights from individual analytic memos, enabling the lead researchers to write synthesis memos that summarized the research team's conjectures about the metacognitive knowledge students needed to learn to plan more effective iterations.

Finally, we continued to develop empirical grounding and refine these conjectures by conducting *theoretical sampling*—a grounded theory strategy in which researchers intentionally collect and analyse additional data to refine or refute the conjectures that emerged from previous rounds of analysis (Charmaz, 2003; Corbin & Strauss, 2014). We did this by redesigning the Design Canvas to include new boxes (based on newly emerging categories in the coding scheme) that prompted the design teams to explicitly discuss new areas of the project where we conjectured that the novices were ignoring potential risks.

Because we already had some expectations about the metacognitive knowledge that novices would need to learn, informed by the Design Risks Framework and our previous experience as design instructors, we did not use classical

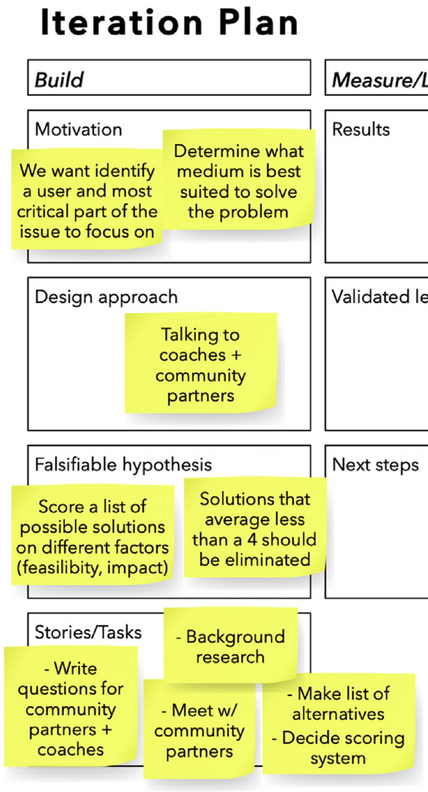

Figure 3 During the weekly planning workshop, design teams used this Iteration Plan template to represent their plan for iterating. This example comes from the dementia team in week 1 (reproduced for readability)

Table 1 Supported by the Design Canvas, novice design teams focused their attention on their knowledge of 10 key areas in their design projects

Areas of Projects	*Description*
Client	A person at a partner organization with relevant expertise (e.g., non-profit) that connects designers with resources (e.g., information, access to users). May also implement solution, if they find it helpful.
User Access Plan	Plan for accessing users (see Users row) to learn about their needs.
Demoing Plan	Plan for regular feedback from the client.
Desired Impact	The social impact that designers intend.
User	Person who will use the proposed solution. Users have *needs,* which have 3 components: a “job” (a task users must complete), a “pain” (a challenge they face in that task), and a “gain” (the benefit they will attain if they can complete the task). By satisfying user needs, designers can entice users to adopt the proposed solution (and thereby promote the Desired Impact).
Root Causes	A causal model of the fixable causes explaining why user need, client need, and desired impact are unmet.
Value Proposition	A proposed solution and how it will overcome root causes to satisfy user, client, and desired impact.
Existing Solutions	Existing solutions and why they are inferior.
Implementation Strategy	Explains who will implement the solution and how. Often involves a hand-off to the client.
Impact	Evidence of desired impact.

Understanding metacognition for iteration

Table 2 Novice design teams struggled to apply 49 criteria for identifying risks across 10 key areas of their design projects

Project Areas	*Criteria for identifying risks*	*Risk to the project*
Client	1. No contact with a real person at a partner organization 2. No identified client need (either as a concrete "job," "pain," and "gain," or a clearly measurable social impact goal) 3. Need isn't reasonable (it conflicts with data and/or common knowledge) 4. No supporting evidence (including both the content *and* the source of supporting data)	If designers do not understand the client's needs, there is a risk of designing a solution that the client does not want.
User Access Plan	5. No plan 6. Plan won't achieve access to intended user 7. Plan probably won't work (based on common knowledge and any evidence) 8. No supporting evidence (including both the content *and* the source of supporting data)	If designers do not have a plan to access users, there is a risk they will be unable to check whether they understand the users' needs and are making progress toward a solution the users want.
Demoing Plan	9. No plan 10. Plan doesn't involve demoing to intended client 11. Plan doesn't involve demoing every 1–2 weeks 12. Plan probably won't work (based on common knowledge and any evidence) 13. No supporting evidence (including content *and* source)	If designers do not have a plan for demoing, there is a risk they will be unable to check whether they are making progress toward a solution the client wants.
Desired Social Impact	14. No defined desired impact 15. Desired impact is not a *social* impact 16. No identified challenges preventing desired impact 17. Challenges aren't believable (given data and common knowledge) 18. No objective way to measure impact 19. No baseline measurement or goal 20. No deadline for reaching the goal 21. No supporting evidence (including content *and* source)	If designers do not understand the desired social impact of their project, there is a risk they will misconstrue the root causes of the social problem and design an ineffective solution *and* there is a risk they will be unable to judge whether their solution made an impact.
User	22. No identified user 23. User doesn't matter to the client 24. No identified user need (including a job, pain, and gain) 25. User need is not well-supported by data and common knowledge 26. No supporting evidence (including content *and* source)	If designers cannot articulate a user need that is supported by evidence, there is a risk they will misconstrue the root cause(s) of that need and design ineffective solutions.
Root Causes	27. No defined causal chains that link the obstacles to satisfying the user's need, client's need, and desired impact each back to a fixable root cause 28. Missing obstacles to satisfying user need, client need, and/or desired impact 29. Causal chains are not credible relationships between individual variables 30. Causes not reasonable given common knowledge and available data 31. No supporting evidence (including content *and* source)	If designers have not identified the fixable root causes of a problem, there is a risk that their solutions will be ineffective and rejected by users or the client.

(*continued on next page*)

Table 2 (*continued*)

Project Areas	*Criteria for identifying risks*	*Risk to the project*
Value Proposition	32. No value proposition (including solution, features, and justification to root causes and user need) 33. Value proposition doesn't address a fixable root cause and user need from the root cause analysis 34. No argument for how solution's features will overcome fixable root cause to address need 35. Not specific enough to guide building and testing prototypes 36. Not specific enough that testing it would yield decisive results 37. No/weak evidence that the solution is desirable 38. No/weak evidence that the solution is effective 39. Evidence doesn't specify both content *and* source of data	If designers cannot explain and provide evidence of how their solution will solve the user's problem, there is a risk that it will not.
Existing Solutions	40. No identified existing solutions (or very strong argument that none exist) 41. No reasonable argument (based on data and common sense) that existing solutions are inferior 42. Existing solutions not actually relevant to the intended user and need 43. No supporting evidence (including content *and* source)	If the solution is inferior to existing solutions, there is a risk that the user or client will not adopt it.
Implementation Strategy	44. No defined implementation strategy (including defining the resources needed to execute it) 45. Implementation strategy will probably implement the solution in a way that does not work for the client or does not achieve desired impact 46. No credible arguments (supported by common knowledge and data) that implementation strategy is feasible 47. No supporting evidence (including content *and* source)	If designers do not know how they will build and diffuse the solution—or if they lack evidence that their strategy will work—there is a risk of designing something that is never implemented.
Impact	48. No believable argument that desired impact was achieved 49. No supporting evidence (including content *and* source)	Even if a solution is implemented, there is a risk that it may not make the intended social impact for unforeseen reasons. By measuring impact, designers ensure their solution has worked.

grounded theory methodology, in which researchers begin with a completely blank slate (Glaser & Strauss, 1967). Rather, we used a variant of grounded theory in which initial data collection and analysis were guided by pre-existing *sensitizing concepts* (Charmaz, 2003), or expectations about what might be important. The only difference between this variant of grounded theory and classical grounded theory is that the first round of data collection involves theoretical sampling rather than being completely open-ended. In this case, our sensitizing concepts were the Design Risks Framework and the initial boxes in the Design Canvas and Iteration Plan which we intended would provoke discussion among the novices about the project areas we expected would be important. Nevertheless, we remained open to unexpected findings as required in grounded theory (Charmaz, 2003; Corbin & Strauss, 2014), and this enabled us to refine the focus of our analysis over time.

Specifically, we narrowed our focus over time to analysing the risks that design teams failed to identify. We did because the teams usually *focused their attention* on key project areas (likely due to the Design Canvas; Table 1), but they often failed to *identify risks*, so we could rarely evaluate how well they *chose strategies* to mitigate risks. Interestingly, there is extensive prior research on designers' strategies (Adams & Atman, 1999; Ball & Christensen, 2019), but very little research that explicitly investigates designers' knowledge of risks. This revealed a useful property of the framework, when applied to novice designers: it focused our attention on the aspects of metacognition where these particular novice designers began to struggle. Conducting this research might, therefore, enable us to develop future coaching strategies that would address the root problem rather than downstream symptoms.

After developing the coding scheme through grounded theory as described above, we conducted a second phase of analysis to test inter-rater reliability. The purpose of this analysis was to test whether two researchers, given the same data, could independently and reliably use the categories in the coding scheme to identify risks. The coding scheme for this analysis is provided in Table 2. The risky project areas (Table 2, Column 1) are the category codes in our coding scheme that we used to test inter-rater agreement. We used the criteria for identifying risks (Table 2, Column 2) as sub-codes to reason about when to apply the category codes. Researchers applied these codes to Design Canvases from five of the planning workshops (17% of planning workshops). Researchers coded images in this phase of analysis to ensure that both raters were provided with exactly the same data to code. We chose this set of five so that all teams and all weeks of the project were represented, because different coding issues might arise across teams and project stages. Two researchers successfully achieved inter-rater agreement (Cohen's kappa = .79; Landis & Koch, 1977) by independently coding teams' Design Canvases using the coding scheme (Table 2).

5 *Findings*

We applied the framework to these case studies to demonstrate both the complexity of the metacognition that controls iteration, and the framework's practical utility for diagnosing specific areas where coaching might help novice design teams to iterate more effectively. Applying the framework revealed that the novice design teams struggled to identify risks in their projects because they struggled to apply 49 different criteria for evaluating their knowledge across these 10 areas (Table 2).

To illustrate how teams struggled to apply diverse criteria for identifying risks, we provide several representative examples from one team's reasoning in week 3 of their project. This team—the Hospital Team—had risky assumptions about many important areas of their project. We focus on 3 key areas: their client, their analysis of the root causes of the problem, and their solution idea (also called their value proposition). Our team identified that risks in each of these areas might have blocked the team from making their desired impact of reducing teen depression, yet the team failed to identify these risks while planning.

5.1 The Hospital Team's client

The Hospital Team was tasked to "work on teen depression", which their client, a local hospital, had identified as an important local issue. However, our observations and images of the team's Design Canvas suggest that, by week 3, the team had little information about what social impact outcomes the hospital wanted to achieve. This was risky because the team expected the hospital to implement their solution.

5.1.1 Vague assertions about client's needs

The first set of risks in the team's understanding of the client came from vague claims about the client's needs. For the client's needs, the team wrote and discussed three ideas: "reduce stigmas," and "raise awareness about adolescent depression" (Figure 4). These are vague statements because it is very different if the hospital wants to raise awareness to promote enrolment in a new treatment program the hospital is offering, versus a public health outreach program raising awareness so adolescents understand they are not alone. These vague statements create a risk of having incorrect assumptions about what kinds of solutions the hospital is willing and able to implement. For example, by further investigating the hospital context, the team might learn that the hospital already has depression treatment programs, but has trouble getting adolescents to enrol. Alternatively, the team might learn that the hospital lacks treatment programs but wants to develop a new outreach program to support adolescent mental health. These are two distinct situations that imply different solutions (e.g., a publicity campaign vs. a curriculum). If the team assumed the hospital wanted a curriculum when the hospital really wanted a publicity campaign, they would spend time developing an ineffective solution that they could have spent making impact.

5.1.2 Vague evidence about the client's needs

The second set of risks in the team's understanding of the client comes from vague evidence of the client's needs. Our field notes and images of the team's Design Canvas suggest that the team noted and discussed two pieces of evidence for their information about the client's needs: "suicides" and "local hospital feedback; 2014 state-wide survey" (Figure 4). This evidence is ambiguous.

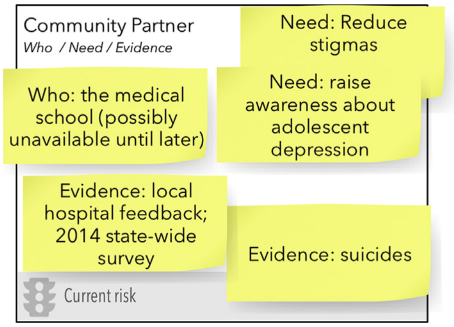

Figure 4 Community partner section of the Hospital Team's Design Canvas template in week 3 (reproduced for readability)

It might suggest that the hospital is willing to fund initiatives to reduce suicides. Alternatively, it might only suggest that suicides have been happening in the community, or that hospital administrators are concerned about suicides (but may not be prepared to fund mental health initiatives). The source of this evidence is also unclear, which creates a risk of misinterpreting the evidence. It is different if the team has evidence from a powerful hospital executive that the hospital wants to fund anti-suicide initiatives, versus if the team has evidence from local news clippings that the community is experiencing a wave of suicides. In the first case, the team could design resource-intensive solutions. In the second case, the team might first need to convince hospital administrators to fund mental health programs. These are different directions for the team's design project, and without learning more by gathering more specific evidence about the client's needs, the team risks choosing an unproductive direction. In addition to this risky ambiguity, it is unclear what the team learned from "local hospital feedback" and the "2014 statewide survey". The survey respondents might have been hospitals, patients, or community members, and they might have reported many different things that would each have different implications for the design project, so it is important for the team to understand this evidence in a more specific way.

It is entirely possible that the Hospital Team used these notes to index a more elaborate shared understanding of their evidence that would answer each of these questions. However, our field notes suggest that they did not. When the Hospital Team was working to add this evidence to their Design Canvas, they named each source of evidence without elaborating on how it supported their claims about the client's needs.

5.1.3 Evidence does not support team's assertions about client's needs

Third, the team's evidence did not support their assertions about the client's needs. The team wrote and discussed that the hospital's needs were "reduce stigma" and "raise awareness about depression." In part because the team's evidence is ambiguous, it begs the question how “suicides” and “local hospital feedback; 2014 state-wide survey” indicate that the hospital has an organizational goal to “reduce stigma” and “raise awareness about depression.” Evidence from field notes confirms that the team made a logical leap here; over their first three weeks of planning sessions in their project, the team never wrote down or discussed evidence that the hospital would be willing and able to implement the solution they wanted to design. Instead, the team confused evidence that hospital staff were *aware* of the problem of teen depression and suicides with evidence that the hospital was interested in *mobilizing* resources to address this problem. They seemed to assume that the hospital would want to reduce stigma and raise awareness about depression simply because the team saw it as the hospital’s job to care about public health.

5.2 The Hospital Team’s root causes and value proposition

Two other risky areas in the Hospital Team’s project in week 3 were their analysis of the root causes of adolescent depression and their value proposition (Figure 5). A value proposition is a sentence template that designers use to express how their solution addresses the root causes of the problem; in other words, it is the solution idea plus justification (Osterwalder et al., 2014). Value propositions serve as hypotheses that designers can test against data to validate or reject their solution idea. The team’s value proposition was that “mental health reading log for middle school students” would help form a habit of “communication between students + parents” (Figure 5). We identified three risks related to the team’s value proposition and root cause analysis. First, the team’s proposed solution did not seem to address the root causes they identified in their root cause analysis. Second, the team’s evidence did not support the efficacy or desirability of their proposed solution. Third, the team’s analysis of root causes does not explain what the root cause of adolescent depression is. These issues revealed a risk that the Hospital Team would design something that does not make impact—either because it was not used, or because it did not effectively address the root causes of adolescent depression.

5.2.1 Value proposition does not address root causes

First, the Hospital Team’s proposed solution did not seem to address the root causes they identified in their root cause analysis. These root causes included: parents being reluctant to communicate with children about emotions, parents having stigmas around mental health issues, and parents lacking mental health education (Figure 5). In light of these root causes, it is not clear that the team’s proposed solution should work: why would assigning mental health-related readings to middle schoolers help those middle schoolers to form a habit of communicating about emotions with their parents (as stated in the team’s value proposition), if the parents are unwilling and ill-prepared to engage in this communication (as stated in the root cause analysis)? Readers may be surprised that the novice designers lost track of this seemingly obvious

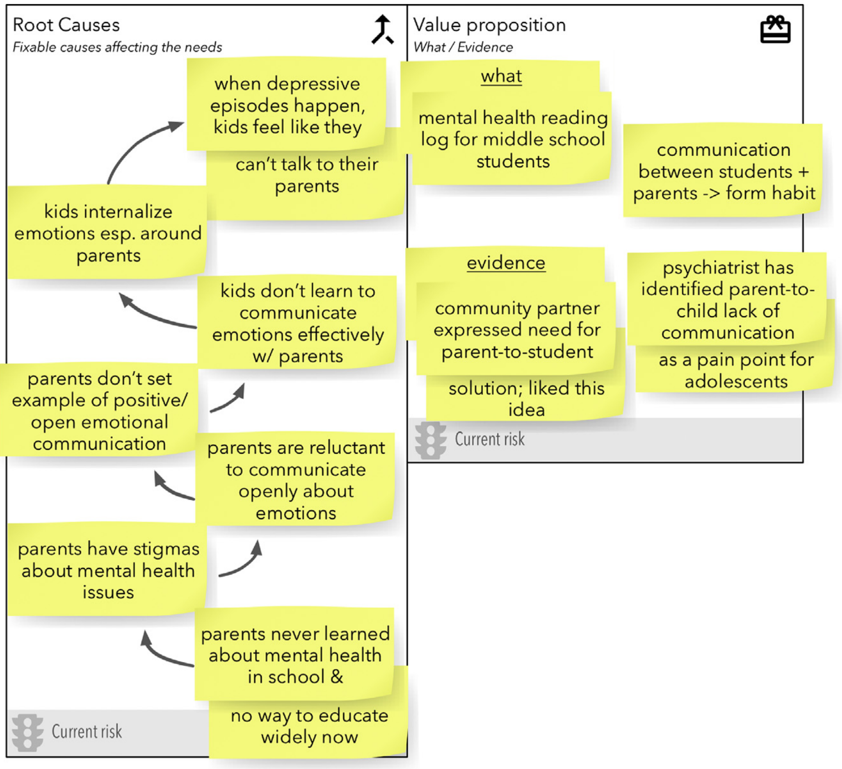

Figure 5 Root causes and value proposition sections of the Hospital Team's Design Canvas template in week 3 (reproduced for readability)

misalignment (proposing a solution that does not address the problem). Yet consider how common it is for novice researchers to propose a research question that does not align with their literature review, methods, and/or findings. The sheer complexity of design problems (including product design problems and designing research studies) means that designers face a constant challenge to maintain coherence between many complicated and evolving ideas; in light of this, it makes sense that novices (and even experienced designers) often find misalignments in their thinking like this one. What matters is that novices learn to identify and address the risks created by misalignments.

5.2.2 Evidence does not support the value proposition

A second risk was that the team's evidence did not support their value proposition. Specifically, the team mentioned no evidence in their discussion or on their Design Canvas that their proposed solution would be effective or desirable to adopt. The evidence from the client (what they call the "community partner") seemed promising: "community partner expressed need for parent-to-student solution; liked this idea." This suggests that the client might implement this solution (assuming it aligns with the hospital's organizational goals, which remained a risky assumption as we discussed above). However, the evidence from the psychiatrist has more to do with defining the problem than validating a solution. Furthermore, the team had no evidence that their mental health reading logs would increase emotional communication between middle schoolers and their parents—or that middle schoolers would even be willing to use the logs (which seems unlikely; it is not intuitively obvious that middle schoolers would want to do this reading, and this design team never wrote down or discussed any evidence suggesting otherwise, based on their Design Canvas and our field notes of their planning discussions). This lack of evidence is a source of risk because it indicates the solution may not make the desired impact of reducing teen depression.

5.2.3 Root cause analysis does not explain root causes of teen depression

Third, the team's analysis of root causes does not identify a root cause of teen depression. The causal chain they discussed while planning and recorded on their Design Canvas (Figure 5) ended with the statement "when depressive episodes happen, kids feel like they can't talk to their parents". The fact that depressed teenagers avoid talking with their parents does not explain why they become depressed in the first place. The team might believe that depressive episodes are inevitable, and what matters is how one deals with them. Even so, it is also ambiguous how the team believes the lack of parent-child communication affects teen depression. For example, it could be that the team believes talking about depression directly helps people to overcome depressive episodes. Alternatively, the team might expect that talking with parents would lead to the parents arranging professional therapy, which would help the teens overcome their depressive episodes. These are two different theories that imply the need for completely different solutions—the first might require a solution that scaffolds in-depth reflective discussion while the second might only require helping parents become aware of the teen's depressive episode. In summary, the team had an ambiguous model of what causes teen depression, which made it ambiguous which solutions should be effective, which created a risk of spending time developing a solution that would not reduce teen depression.

We focus on discussing these specific areas of the Hospital Team's reasoning in week 3 of their project to illustrate the granularity of the metacognitive knowledge required to identify risks effectively. We have shown how the Hospital Team struggled to identify risks because they did not apply specific criteria for evaluating their knowledge of their client, root cause analysis, and value proposition. In total, we identified 49 criteria for identifying risks based on our findings from across all 5 novice design teams we studied (Table 2). The complexity of this body of knowledge helps us to understand why novice designers find iteration so difficult, and suggests many, specific areas where coaching might help novice design teams to iterate more effectively.

6 Discussion

6.1 Theoretical implications

Our findings begin to answer a recent call in this journal to understand better the metacognitive underpinnings of the design process (Ball & Christensen, 2019). By using the Design Risks Framework, we developed conjectures about the knowledge structures involved in monitoring and controlling design iterations. Specifically, we found that novice design teams seemed to lack metacognitive knowledge of 49 different criteria for identifying project risks, the second metacognitive process in the Design Risks Framework. The teams' difficulty identifying risks seemed to prevent the teams from meaningfully choosing iterative strategies to mitigate those risks and advance their projects.

This lack of key metacognitive knowledge helps to explain the extant finding that novice designers struggle to iterate effectively (Atman et al., 2007, 1999). Yet, metacognitive knowledge falls beyond the scope of prior theoretical models of iteration (Ball & Christensen, 2019). Prior research has defined the process of iteration (Adams & Atman, 1999; Atman & Adams, 2000), expert-novice differences in iteration (Atman et al., 2007, 1999), and how ideas evolve through iteration (Dorst & Cross, 2001; Guindon, 1990; Jin & Chusilp, 2006; Schön, 1983). The Design Risks Framework adds to these existing models by enabling us to make plausible, empirically-consistent conjectures about the specific, underlying knowledge that enables effective iteration and distinguishes experienced designers from novice designers.

This framework reflects a theoretical contribution that expands our understanding of iteration and design metacognition by highlighting the role of metacognitive processes and knowledge in controlling iteration. The framework highlights three metacognitive processes for monitoring and controlling iteration: *focusing attention* on key areas of the project, *identifying risks* in those areas, and *choosing strategies* to mitigate those risks. The framework directs researchers to analyse the metacognitive knowledge structures underlying these processes: schemata that guide where designers focus their attention, criteria designers use to identify risks, and knowledge of which iterative strategies mitigate certain risks. In doing so, this framework enables researchers to explain a designer's (or design team's) iteration in terms of the application or conspicuous absence of specific metacognitive knowledge.

Note that we do not necessarily expect all of the specific schema slots (Table 1; used in the "focus attention" phase of the DRF) and risks (Table 2; used in the "identify risks" phase of the DRF) identified in this study to generalize to all designers or design problems—nor are they meant as a comprehensive set of all metacognitive knowledge that might be useful in approaching the design problems in these case studies. Design problems can be framed in different ways that implicate different domains of knowledge (Ball & Christensen, 2019; Buchanan, 1992), and we expect this will hold true for metacognitive knowledge as well. We expect that different designers will have developed different metacognitive knowledge (different schemata and risk criteria) to cope with the unique design challenges they have encountered in their experience, but designers may also develop similar metacognitive knowledge to cope with similar challenges that arise across domains. For example, policy designers might focus their attention using a schema including slots for the broader legal framework and the state of political willpower, and identify risks using corresponding criteria such as constitutionality and political feasibility—issues that did not emerge in our case studies. But they might also focus their attention and identify risks related to root causes and implementation strategies—issues that did emerge from these case studies. Using the DRF and perhaps building on the initial set of metacognitive knowledge identified here, future researchers and design instructors can map this metacognitive knowledge across a wide range of designers and design problems.

6.2 Implications for coaching iteration

While this study did not investigate coaching directly, by highlighting the metacognitive knowledge that seems to control iteration, the framework generates a more complete picture of how instructors might coach iteration. To see this, consider the primary coaching strategy implied by existing models: recommending iterative strategies.

Several existing models define iteration as applying iterative strategies—sequences of transitions between cognitive activities—to refine one's knowledge of the problem and advance a solution (Adams et al., 2003; Jin & Chusilp, 2006). This implies that the underlying knowledge required to iterate is knowledge of iterative strategies and when to apply them. Based on these models, some have suggested coaching iteration by recommending appropriate iterative strategies to designers (Chusilp & Jin, 2006).

To be sure, choosing strategies is an important driver of iteration and a major component of the DRF. However, our findings demonstrate there is other underlying knowledge−knowledge of key project areas and risks—that may be even more fundamental and important to coach before one can meaningfully assess and coach how novice designers choose iterative strategies. When we tried to analyse how well the novice design teams chose strategies, we found it was difficult to do because the teams were unable to first identify risks. To evaluate a choice of strategy, a designer must evaluate whether it would plausibly mitigate certain risks. But if designers do not have a clear idea of the risks they are trying to address, it simply does not make sense to ask whether they have chosen appropriate strategies for mitigating those risks. This reveals that much of the work involved in coaching iteration may not be directed toward suggesting iterative strategies. Instead, effective coaching may often need to look upstream to improve how novice designers are attending to key areas of the project and identifying risks.

Related research supports this hypothesis (Rees et al., 2018). In that study, we evaluated a coaching strategy that targeted the metacognitive knowledge required for focusing attention and identifying risks—something that would not have been possible without the DRF. We provided novice design teams with a Design Canvas to help them focus their attention on key project areas, and a checklist of criteria to help them identify project risks. We found that these design teams planned more iteratively *and* iterated better in their projects, compared with another group of novice design teams who instead received 2 h of face-to-face coaching each week to help them plan their next iteration (Rees et al., 2018; Rees Lewis, 2018). This suggests that targeted coaching based on this framework can be more effective than unstructured face-to-face coaching. It might be most effective to combine these instructional strategies by providing teams with a Design Canvas and risks checklist to support planning discussions within their teams while also providing a point of departure for coaching discussions with instructors.

6.3 Limitations

In this study, we used grounded theory methods to develop empirically grounded conjectures about the metacognitive knowledge that enables effective iteration. While grounded theory methods are appropriate for developing plausible conjectures in new, underexplored areas (such as design metacognition; Ball & Christensen, 2019), these methods did not validate or prove that the conjectures are true (Corbin & Strauss, 2014). Thus, validation research is needed. This might include correlational studies that measure the relationship between specific kinds of metacognitive knowledge proposed in the Design Risks Framework and iterative behaviour across different designers. Researchers might also conduct experiments in which a random subset of novice designers are taught metacognitive knowledge to see if this changes their iterative behaviours in design tasks where that knowledge is useful.

7 Conclusion

Previous research suggests that novice designers iterate in a haphazard rather than strategic way (Crismond & Adams, 2012), but existing models of iteration do not identify the metacognitive knowledge that managers and educators should coach to help novice designers iterate better (Ball & Christensen, 2019). Using the DRF, researchers can perform a new kind of analysis to understand the specific metacognitive knowledge underlying iteration where novice designers need support. This enables researchers to design coaching tools that draw attention to these abilities (e.g., Rees et al., 2018). Managers can use these tools to focus on coaching the critical metacognitive elements of iteration where novice designers need support (e.g., Zhang, Easterday, Gerber, Rees Lewis, & Maliakal, 2017). By using this framework as the theoretical underpinning to analyse metacognitive knowledge and design coaching for iteration, researchers may enable managers and educators to better support the efforts of design teams tackling complex design challenges across design domains and industries.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank Bruce Sherin, Kristine Lu, Eureka Foong, Jamie Gorson, and the Delta Lab for feedback on earlier drafts of this paper. We are grateful to Alex Sher for supporting the project. This work was funded by U.S. National Science Foundation grants IIS-1530883 and IIS-1320693.

References

Ackerman, R., & Thompson, V. A. (2017). Meta-Reasoning: Monitoring and control of thinking and reasoning. *Trends in Cognitive Sciences, 21*(8), 607−617. https://doi.org/10.1016/j.tics.2017.05.004.

Adams, R. S., & Atman, C. J. (1999). Cognitive processes in iterative design behaviour. *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference* 11A6/13−11A6/18. https://doi.org/10.1109/FIE.1999.839114.

Adams, R. S., Turns, J., & Atman, C. J. (2003). Educating effective engineering designers: The role of reflective practice. *Design Studies, 24*(3), 275−294. https://doi.org/10.1016/S0142-694X(02)00056-X.

Ambrose, S. A., Bridges, M. W., diPietro, M., Lovett, M. C., & Norman, M. K. (2010). *How learning works: Seven research-based principles for smart teaching* (1st ed.). Jossey-Bass.

Atman, C. J., & Adams, R. S. (2000). Characterizing engineering student design processes: An illustration of iteration. *Proceedings of the Annual Meeting of the American Society of Engineering Education Conference, Session, 2330*, 143.1−143.11, (St. Louis, MO).

Atman, C. J., Adams, R. S., Cardella, M. E., Turns, J., Mosborg, S., & Saleem, J. (2007). Engineering design processes: A comparison of students and expert practitioners. *Journal of Engineering Education, 96*(4), 359−379. https://doi.org/10.1002/j.2168-9830.2007.tb00945.x.

Atman, C. J., Chimka, J. R., Bursic, K. M., & Nachtmann, H. L. (1999). A comparison of freshman and senior engineering design processes. *Design Studies, 20*(2), 131−152.

Ball, L. J., & Christensen, B. T. (2019). Advancing an understanding of design cognition and design metacognition: Progress and prospects. *Design Studies, 65*, 35−59. https://doi.org/10.1016/j.destud.2019.10.003.

Blank, S. G., & Dorf, B. (2012). *The startup owner's manual: The step-by-step guide for building a great company* (1). K&S Ranch Press.

Buchanan, R. (1992). Wicked problems in design thinking. *Design Issues, 8*(2), 5. https://doi.org/10.2307/1511637.

Charmaz, K. (2003). Grounded theory: Objectivist and constructivist methods. In N. K. Denzin, & Y. S. Lincoln (Eds.), *Strategies of qualitative inquiry* (2nd ed.). Sage.

Chusilp, P., & Jin, Y. (2006). Impact of mental iteration on concept generation. *Journal of Mechanical Design, 128*(1), 14–25. https://doi.org/10.1115/1.2118707.

Constable, G. (2014). *Talking to humans success starts with understanding your customers*. Retrieved from. https://s3.amazonaws.com/TalkingtoHumans/Talking+to+Humans.pdf.

Corbin, J. M., & Strauss, A. L. (2014). *Basics of qualitative research: Techniques and procedures for developing grounded theory* (4th ed.). SAGE.

Crismond, D. P., & Adams, R. S. (2012). The informed design teaching and learning matrix. *Journal of Engineering Education, 101*(4), 738–797. https://doi.org/10.1002/j.2168-9830.2012.tb01127.x.

Dorst, K., & Cross, N. (2001). Creativity in the design process: Co-evolution of problem–solution. *Design Studies, 22*(5), 425–437. https://doi.org/10.1016/S0142-694X(01)00009-6.

Gero, J. S., & Mc Neill, T. (1998). An approach to the analysis of design protocols. *Design Studies, 19*(1), 21–61. https://doi.org/10.1016/S0142-694X(97)00015-X.

Glaser, B. G., & Strauss, A. L. (1967). *The discovery of grounded theory: Strategies for qualitative research* (1st ed.). Aldine.

Greene, J. A., & Azevedo, R. (2007). A theoretical review of Winne and Hadwin's model of self-regulated learning: New perspectives and directions. *Review of Educational Research, 77*(3), 334–372. https://doi.org/10.3102/003465430303953.

Guindon, R. (1990). Designing the design process: Exploiting opportunistic thoughts. *Human–Computer Interaction, 5*(2–3), 305–344. https://doi.org/10.1080/07370024.1990.9667157.

Jin, Y., & Chusilp, P. (2006). Study of mental iteration in different design situations. *Design Studies, 27*(1), 25–55. https://doi.org/10.1016/j.destud.2005.06.003.

Kolodner, J. L., & Wills, L. M. (1996). Powers of observation in creative design. *Design Studies, 17*(4), 385–416. https://doi.org/10.1016/S0142-694X(96)00021-X.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159–174. https://doi.org/10.2307/2529310.

Osterwalder, A., Pigneur, Y., Bernarda, G., & Smith, A. (2014). *Value proposition design: How to create products and services customers want*. Hoboken, NJ: John Wiley & Sons.

Rees, L., Gorson, J. S., Maliakal, L. V., Carlson, S. E., Gerber, E. M., & Easterday, M. W. (2018). Planning to Iterate: Supporting Iterative Practices for Real-world Ill-structured Problem-solving. *Rethinking Learning in the Digital Age: Making the Learning Sciences Count, 1*, 9–16.

Rees Lewis, D. G. (2018). *Beyond Problems on a Platter: Creating Tools for Teaching Planning in Real World Design* [Doctoral Thesis]. Northwestern University.

Rees Lewis, D. G., Gerber, E. M., Carlson, S. E., Easterday, M. W., et al. (2019). Opportunities for educational innovations in authentic project-based learning: understanding instructor perceived challenges to design for adoption. *Educational Technology Research and Development, 67*(4), 953–982. https://doi.org/10.1007/s11423-019-09673-4.

Rittel, H. W. J., & Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy Sciences, 4*(2), 155–169. https://doi.org/10.1007/BF01405730.

Rumelhart, D. E. (1980). Schemata: The building blocks of cognition. In R. J. Spiro (Ed.), *Theoretical issues in reading comprehension* (1st ed.). (pp. 33–58) Routledge. https://doi.org/10.4324/9781315107493-4.

Schön, D. A. (1983). *The reflective practitioner: How professionals think in action*. New York: Basic Books.

Schön, D. A., & Wiggins, G. (1992). Kinds of seeing and their functions in designing. *Design Studies, 13*(2), 135–156. https://doi.org/10.1016/0142-694X(92)90268-F.

Simon, H. A. (1973). The structure of ill structured problems. *Artificial Intelligence, 4*(3), 181–201. https://doi.org/10.1016/0004-3702(73)90011-8.

Small, M. L. (2009). "How many cases do I need?": On science and the logic of case selection in field-based research. *Ethnography, 10*(1), 5–38. https://doi.org/10.1177/1466138108099586.

Winne, P. H. (2010). Improving measurements of self-regulated learning. *Educational Psychologist, 45*(4), 267–276. https://doi.org/10.1080/00461520.2010.517150.

Winne, P. H., & Azevedo, R. (2014). Metacognition. In R. K. Sawyer (Ed.), *The cambridge handbook of the learning sciences* (2nd ed.) (pp. 63–87). https://doi.org/10.1017/CBO9781139519526.006.

Wynn, D. C., & Eckert, C. M. (2017). Perspectives on iteration in design and development. *Research in Engineering Design, 28*(2), 153–184. https://doi.org/10.1007/s00163-016-0226-3.

Zhang, H., Easterday, M. W., Gerber, E. M., Rees Lewis, D., & Maliakal, L. (2017). *Agile Research Studios. Orchestrating Communities of Practice to Advance Research Training* 220–232. https://doi.org/10.1145/2998181.2998199.