Scientific Research Publishing

Journal of Software Engineering and Applications, 2018, 11, 299-317
http://www.scirp.org/journal/jsea
ISSN Online: 1945-3124
ISSN Print: 1945-3116

"*I'm Just Guiding You*": An Exploration of Software Design Mentorship within a Software Engineering Firm

Leshell Hatley[1], Maha Al-Freih[2], Brenda Bannan[3]

[1]Department of Mathematics & Computer Science, Coppin State University, Baltimore, USA
[2]Department of Educational Technology, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia
[3]Instructional Technology/Learning Technologies Design Research, George Mason University, Fairfax, USA
Email: lhatley@coppin.edu, mmalforaih@pnu.edu.sa, bbannan@gmu.edu

How to cite this paper: Hatley, L., Al-Freih, M. and Bannan, B. (2018) "*I'm Just Guiding You*": An Exploration of Software Design Mentorship within a Software Engineering Firm. *Journal of Software Engineering and Applications*, 11, 299-317.
https://doi.org/10.4236/jsea.2018.116019

Received: February 16, 2018
Accepted: June 26, 2018
Published: June 29, 2018

Abstract

The case study presented here uses an interpretivist (qualitative, humanistic) approach to illustrate and describe a range of interactions and behaviors that occur during design meetings where mentoring and design simultaneously occur within a software engineering firm, during a portion of the design phase for a software project. It attempts to examine the interaction between two design team members (one novice and one expert) and describes how these observations intersect with the theoretical and applied literature and actual design processes. Taking cues from two theoretical descriptions of the design process, the study presented here suggests that modes and models of mentorship should be added, when applicable, as a descriptive portion of the design process.

Keywords

Software Design, Mentorship, Design Meeting, Collaboration

"*Design is what links creativity and innovation. It shapes ideas to become practical and attractive propositions for users and customers. Design may be described as creativity deployed to a specific end.*"

-Sir George Cox
Former Chairman, Design Council
2005

1. Introduction

Overall, design is a multifaceted complex construct implying a variety of activities toward the manifestation of artifacts—from raising issues to be addressed, to questioning improvements and solving problems, all the way to the act of creating, and sometimes even re-creating [1] [2]. Additional words, such as creativity and invention (as the quote above illustrates), have also been used to describe the complex elements and processes of design [3]. Efforts to advance our understanding of the design process and the need to understand it—as it happens (*in-situ*) have given rise to many design research methodologies, concepts, perspectives, and frameworks for examining such a complex process [1] [4] [5] [6]. With these, one of the major shifts of focus in design studies over the past decade or so has been from individual to collaborative design processes and activities. These activities have included understanding how designers communicate with each other and/or with clients while during the design period [3] [7] [8], what are collaborative working styles and performance implications of such collaborative efforts [9], and how and what types of tools can be used to support designers who are co-located and/or intergenerational [10].

This paper describes a study that took place regarding the nature and process of software design to add to this literature on collaborative design processes. One of the objectives was to examine the interaction between software design team members and how observations of a software design team intersect with the theoretical and applied literature and actual design processes. The study described here explores the collaborative nature between two software developers of a local software engineering firm, established ten years ago, as they work together to design a cloud-based (data-intensive, web-based) software tool. We have given this company the fictitious name "Code Theory". Here, we attempt to explore the communication and interaction patterns that occur when designing software, specifically between an expert software engineer and one at a novice level, where design and design mentoring occur simultaneously.

2. Need for Research

Software engineering (software *design*, programming, and testing) is a globalized and fast-paced phenomenon, where education and training typically begin within secondary and postsecondary education levels. Software engineering students graduate and begin working for small, medium, and large-scale companies as they attempt to apply their problem-solving and programming skills and contribute to the design and development of a variety of software products. There are volumes of articles describing how students learn the design process and how they apply design thinking principles in school [11] [12] [13] [14] [15]. Yet, there is a gap in the literature regarding "*on-the-job*" software design training and mentoring, particularly after a student graduates and has started employment. If companies expect to be successful in this technology-driven, fast-paced industry, it is imperative for new hires to sufficiently contribute to the design and innovative nature of their products. However, as Begel & Simon [16] describe in a recent study of the struggles of new college graduates in their first software development jobs, many new hires begin with significant challenges.

After observing new hires for 85 hours on-the-job at Microsoft, they found that these new hires "got stuck" when communicating, collaborating, providing technical expertise at times, cognitively handling the massive amounts of information shared regarding the code for their projects, and orienting to their new work environments.

This case study extends this work and is intended to illuminate the communication patterns that occur when a newly hired employee is paired with a more experienced employee to undergo software design. Here, one engineer is profoundly more experienced and happens to be the founder of the company, while the other software engineer is a relatively novice software developer in comparison, who was employed at the company for a little over two years, although this was not his first job. Observing the social, cognitive, and collaborative design behaviors of two software engineers as they work together to design a cloud-based software tool they envisioned provides a breaking of point for empirically understanding the benefits, challenges, and implications of mentorship and training in software engineering firms, *in-situ*.

This paper begins with a description and illustration of the cognitive processes that occur when design takes place. The social and collaborative interactions that occur when a group of designers work together are also described and illustrated with two theoretical postulates from the literature. Example studies which explore the collaborative nature of software design follow. Participants and design conditions along with data collection and analysis details of this case study are then presented followed by a description of our findings and their implications. This paper ends with report on the limitations of the study and plan for future work.

3. Design as a Cognitive Process

The question of how designers think and work, that is, what designers understand about design and how they go about the act of designing based on this understanding, has been a matter of much discussion and writing in design research [17] [18]. To this end, design cognition is one of the major themes that has been investigated in design studies and research regarding designers of all ages [10] [19] [20]-[28]. The question most asked is: What happens *in the min*d when a creative idea takes shape? Even more, Warr and O'Neill [3] explain that the larger the number of ideas generated, the greater the probability of achieving an effective solution. Thus, the more creative one is in design, the more useful and usable the result will be. They go on to define creativity as having three essential elements: creative process, creative people, and creative products [3]. Here, we focus on the creative process.

So, what is happening in the mind when one creates or designs? Gabora [19] points out that the creative process has a long history of being defined in stages: 1) preparation (where the creator gathers relevant data), 2) incubation (when the creator unconsciously continues to work on the problem at hand), 3) illumination (when the knowledge and theories begin to connect and ideas begin to consciously take shape), and 4) verification (when an idea has been worked out, takes form, and can be communicated to and evaluated by oneself and others). This process can be represented by **Figure 1** below, and taken together, can be described as "ideation" when undergoing a design project.

Koestler [27] proposes that as a designer (or group of designers) progresses through these stages, they deliberately connect previously related and/or unrelated "matrices of thought" to produce ideas. Gabora [19] likens the exploration of these "matrices of thought" to the ideas in one's mind. These matrices of thought can also be combined with elements of the designer's environment, [3], past experiences, and/or technical knowledge over the course of time. **Figure 2** is a more robust representation of this.

4. Design as Social and Collaborative Processes

Gennari and Reddy [28] describe the [collaborative] design process as, "human activity, involving communication and creative thought among a group of participants". Thus, design is a social process that potentially entails thinking and working in multidisciplinary teams and across various perspectives. Essentially, groups of designers follow the same general cognitive process described above as they work together, as illustrated in **Figure 3**.

Due to the social nature of groups, however, a group or teams of designers undergo even more social and cognitive processes as they brainstorm (ideate), problem-solve, evaluate, and make decisions. These processes focus on ideas related to the design project itself (**Figure 4**) as well as social nature of group dynamics such as communicating, coordinating, and collaborating (**Figure 5**). Designers continuously impact one another's cognition as they share ideas related to the design project. **Figure 4** illustrates the conditions under which this sharing occurs related to shared ideas of a web-based software product [26].

Even more so however, a group or teams of designers undergo social events when communicating. These may include, but are not limited to, personality, attitude, considerable framing, reframing, negotiation, decision-making, and even conflict and conflict resolution [3] [8] [27] [30] and can be seen in **Figure 5**.

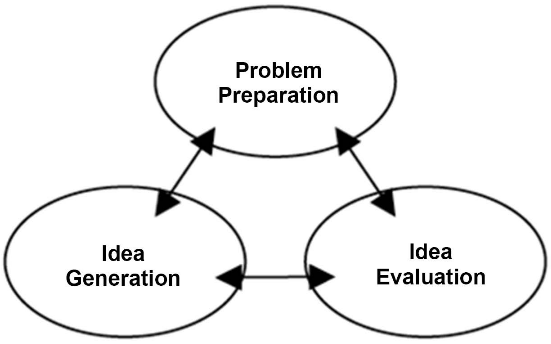

Figure 1. A generic creative process model [11].

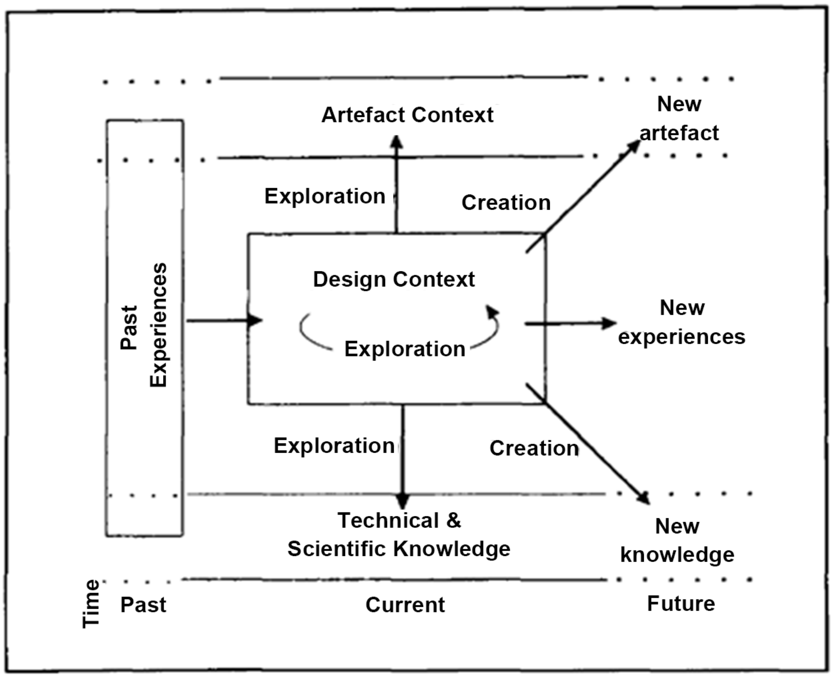

Figure 2. Representation of a designer's knowledge exploration during the design process [22].

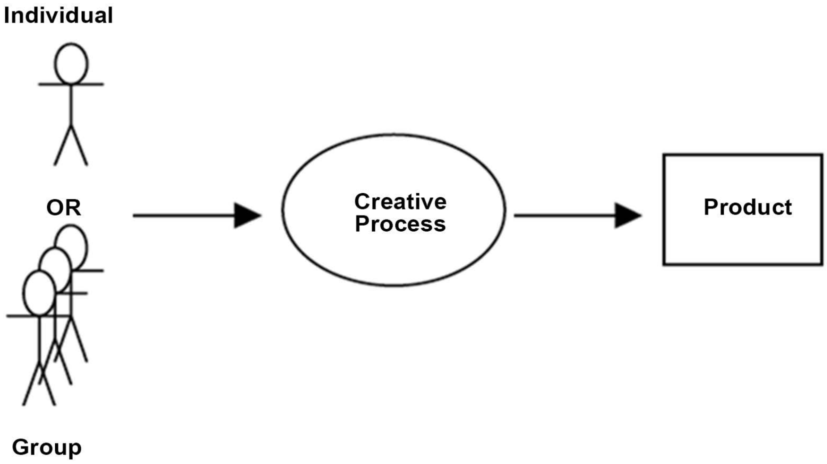

Figure 3. The components of creativity (design) [3].

Below we explore selected research studies that seek to uncover how design teams navigate the above cognitive and social events when designing software.

5. Research on the Social and Collaborative Elements of Design Teams

Goldschmidt and Rodgers [18] compared the thinking processes, methods and approaches of three different groups of designers during a design project, ID (Industrial Design students), ARCH (Architecture students), and DPHD (Design PhD candidates). Data sources included submitted designs and self-reported attitudes, main focus points, and sequence and duration of design activities.

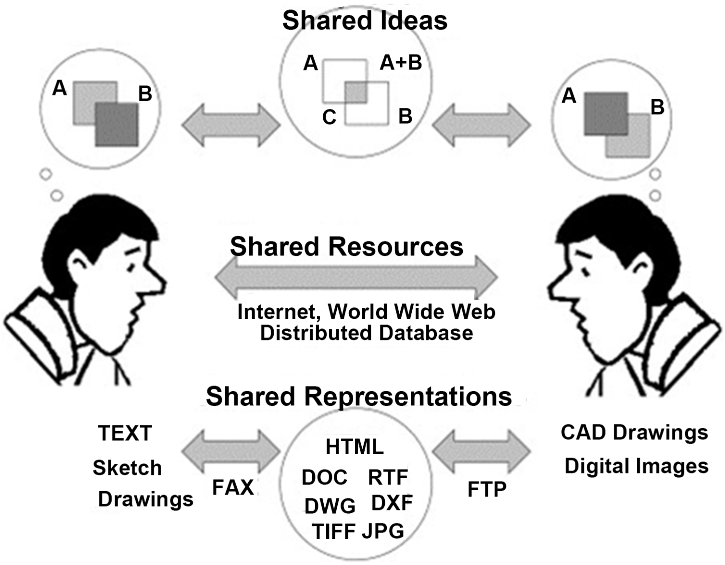

Figure 4. Group communication regarding a design project [29].

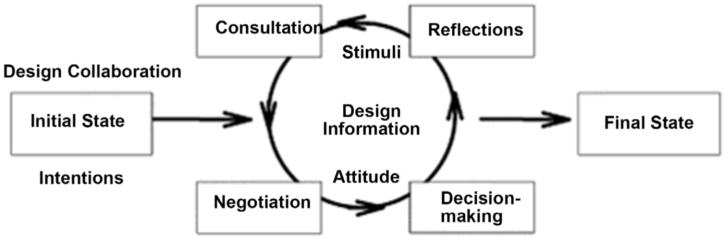

Figure 5. Design team communication events [27].

The researchers concluded that there were some commonalities and differences between the two undergraduate groups but the main differences were between the two undergraduates and the PhD student groups. Both sets of undergraduate design students (ID and ARCH) spent an average of close to one third of their time on "thinking about solutions and sketching them" whereas the DPHD students spent far less time on this activity. The researchers attribute these differences to the fact that DPHD students were more experienced and that their design was aimed towards a service, as opposed to a product, which requires more problem-solving cycles.

Tang *et al.* [5] examined the cognitive aspects of software engineering design. Specifically, the researchers aimed at investigating whether design planning, problem-solution, co-evolution, and reasoning techniques are factors that influence design decision-making and postulate that designers use these techniques in different ways during design sessions. In this exploratory study, two design teams from Adobe Systems Incorporated (Team A) and AmberPoint (Team M) were observed as they built a traffic simulation program. In order to understand how design teams reason with their designs, two coded protocols and a decision map were used to analyze the meeting. Preliminary evidence showed that design decision-making that is based on good design planning, minimal context switching, well-organized problem exploration and good reasoning supports effective software design.

Another study that investigates the role of cognition in design is a study conducted by Dong, Kleinsmann, and Deken [31]. The researchers argue that one of the challenges in design studies is accessing the collective cognitive structures of design teams, and in turn understanding the relationship between team cognition and performance. They point out the difficulty in mapping team mental model measurement methods from existing research to design teams and argue for the need for measurement methods that are in harmony with the type of knowledge required to complete the team's task.

The analysis of team communication revealed that the quality of team mental models and good enactment of these models are both indicators for the quality of team mental models. Specifically, the authors concluded that the enactment of the team mental model is more likely to be effective only when the team mental model shows emergence of sharedness and accuracy in relation to the dynamic referent model. The latent semantic analysis provides insights into whether all team members have mental models that are aligned with the team mental model, while reflective practice analysis enables us to determine if the team mental model is ultimately deployed into goal-directed behavior.

Ke and Im [26] studied team-based computer-game design efforts by middle school students of diverse backgrounds and abilities to explore the nature of their collective design actions and cognitive processes. Several collaborative themes emerged, namely: 1) collective exploration of design constraints during problem framing, 2) aggregation of identity, experience, and memory for collective solution generation, and 3) development of coalition and task interdependence during design execution.

6. Design & Mentorship

Like design, mentoring can also be difficult to define due to the many types and purposes of mentoring, varying contexts and lengths of time within which it occurs, and expected outcomes. However, Psychologist E. Paul Torrance [32] mentioned the organizational culture of mentoring when he said, "What is cultivated in a culture is developed there." By this, he suggested that mentoring is a "process in which one's peers and superiors (or leaders) take it upon themselves to cultivate a mentoring culture and therein develop, motivate, guide, and even protect emerging young minds in their push to achievement" [33]. Hester and Setzer [33] suggest, "The leader as mentor, no matter what the level within the organization, is charged with contributing to creative achievement. The mentor-protégé relationship, without the support of the culture of which it is a part, hangs on the slender thread of personal relationships and individual commitment only. Mentoring, to be effective, cannot be an island unto itself." This is the notion of mentorship we use here. This study helps to uncover when and where it occurs as a mentor and protégé design together.

7. Research Design

7.1. This Case Study: Goals and Research Questions

The purpose of this study was to investigate the design processes of a collaborative design team at a local software engineering firm. Specifically, we observe the interactions between an expert and a novice designer during design meetings as a means of exploring how they navigate designing a cloud-based tool they envisioned while in a software designer mentor-protégé relationship. The research question we attempt to answer is "How do software engineers interact when designing a software application, when their expertise levels vary?"

7.2. This Case Study: Qualitative (Interpretivist) Approach

We find that our approach to this project is interpretivist which is derived from an ontological belief that reality is socially constructed and cannot be separated from the mind of the individual [34]. The interpretivist tradition also acknowledges that these individually constructed realities are also socially constructed as these perspectives interact with the language and thought of the wider society. These ideas align with our purpose for conducting this study as we use naturalistic data collection methods typically used within the interpretivist tradition to access the perspective of several members of the social group (*i.e.* design team) and understand common patterns and themes of thought and action for that group [34].

7.3. This Case Study: Data Collection Methodology

According to Yin [35], a case study design should be considered when: 1) the focus of the study is to answer "how" and "why" questions; 2) you cannot manipulate the behavior of those involved in the study; 3) you want to cover contextual conditions; or 4) the boundaries are not clear between the phenomenon and context [36]. Furthermore, Yin asserts that case study design arises out of the need to understand complex social phenomenon in real-life context. Because collaborative design is a multifaceted process that is difficult to understand out of the context in which it occurs, the flexibility of a case study design seemed appropriate for our purposes [37].

Hence, a descriptive single-case study design [35] was employed in this study to provide an in-depth description of the interactions and behavior of designers during a software design meeting. The unit of analysis for this study was the interactions and conversation between the two members of the design team.

7.4. Participant Selection

We recruited employees from web design and software engineering firms to participate in this study. Due to constrained project schedules, traveling distance, and the timeline for this project, only one company matched our scheduled work plan. This software engineering company has already begun the design of an in-house project and welcomed participation. We were not aware of the dynamics of this design team, until our first meeting for observation.

Experienced Software Designer (aka EXPERT): EXPERT is a 38-year-old male with 18 years experience at the time of this study as a software engineer. He has created and sold two startup companies and founded, Code Theory, seven years before this study took place.

Novice Software Designer (aka NOVICE): NOVICE is a 29-year-old male with 5 years experience at the time of this study as a software engineer. Most of his work experience was as a consultant.

Both have undergraduate degrees in Computer Science.

7.5. This Case Study: Data Collection Sources and Procedures

Data sources for this study included an online questionnaire, observations, and one focus group. Online informed consent forms were provided to members of the design team and participants provided an initial electronic signature and a confirmation consent code of "YES". The same design team members who completed the online questionaire, were observed in a team meeting, and participated in a focus group. The observations and focus group were recorded via still image, audio, and video. Design team members were given a chance to confirm and/or clarify research notes via a focus group after the initial round of data analysis occurred, to confirm design practices, interactions, and procedures. The design team was also provided with a final copy of the design report. Detailed description of the procedures is provided in the following sections.

Online questionnaire. Both design team members received a pre-observation questionnaire online (hosted by Google Forms) before the observation took place to understand individual design preparation and practices before a design team meeting occurred.

Observations. An observation of one design team meeting was conducted and lasted for one hour and fifteen and a half minutes (1:15:36). This meeting took place during a business day at 1 pm within a local bakery and soup restaurant, a company tradition for meeting space since its founding. Both researchers took observation notes of the design space before the meeting started, while project design is in progress, and of the state of the design space upon completion of the design meeting. Still and video images of the design space were recorded at these time periods as well.

Focus group. A focus group of all participating and consenting design team members took place 5 days following the observation to gain more insight into individual and collective roles throughout the design process. An initial set of questions were prepared prior to starting the focus group, however, some questions emerged because of the what was observed during the design meeting as well as the responses and discussion that takes place during the focus group.

Data collected including observation notes, still images, audio, and video recorded meetings and focus group are kept confidential by the researchers. All participant and site names were converted to pseudonyms to protect the privacy of the participants.

7.6. This Case Study: Data Analysis Method

Data analysis was based on previous work in design studies of team collaborations. Most notably, we took cues from Cross and Cross' [38] and Kvan's [39] work. Kvan [39] provided a solid roadmap to follow which explains what really happens when designers collaborate, **Figure 6**. He points to Gero and McNeil's [40] assertion that design consists of a series of distinct events that occupy discrete and measurable periods of time, which is typically remarkably short, and that the level of expertise affects the way these events are completed. Designers work together for moments and then divide up and go their separate ways. Based on the results of this individual work, design direction and understanding may take a different turn. The occurrences of these distinct events are illustrated in **Figure 6**, a model of design collaboration. Moreover, Kvan [39] found that collaboration happens in multiple forms: 1) mutual collaboration, where designers are busy working with each other, 2) exclusive collaboration, in which designers work on separate parts of the problem, negotiating occasionally by asking for advice or feedback from the other, and 3) dictator collaboration, where the designers decide who is "in charge" and that person leads the process. Dictator collaboration seems to be most fitting when mentorship and design training might occur, so we attempted to use these distinctions to identify when these three types of collaboration occurred, if at all, paying attention to dictator collaboration to signal when EXPERT might be mentoring NOVICE.

Additionally, Cross and Cross [38] recorded and analyzed a three-person team design session for the following social occurrences: 1) roles and relationships (RR), 2) planning and acting (PA), 3) information gathering and sharing (IGS), 4) problem analyzing and understanding (PAU), 5) concept generation and adopting (CGA), and 6) conflict avoiding and resolving (CAR). We coded these instances within the transcription of the interaction initials specified above. We then identified which of these occurrences represented roles undertaken when EXPERT played the mentor role vs. when he played the collaborator role.

7.7. This Case Study: Data Analysis Procedure

As mentioned previously, because of the exploratory nature of this study, inductive analysis of the data to determine categories and themes seemed more suitable and in line with our purpose. Data analysis began in the field as observation notes were recorded, initial analytical insights emerged, tentative interpretations and themes were generated during team discussions immediately following the observation, and hypothesis were confirmed during subsequent focus group with design team members. All data were analyzed inductively line-by-line in search for categories and themes that emerged from data [41].

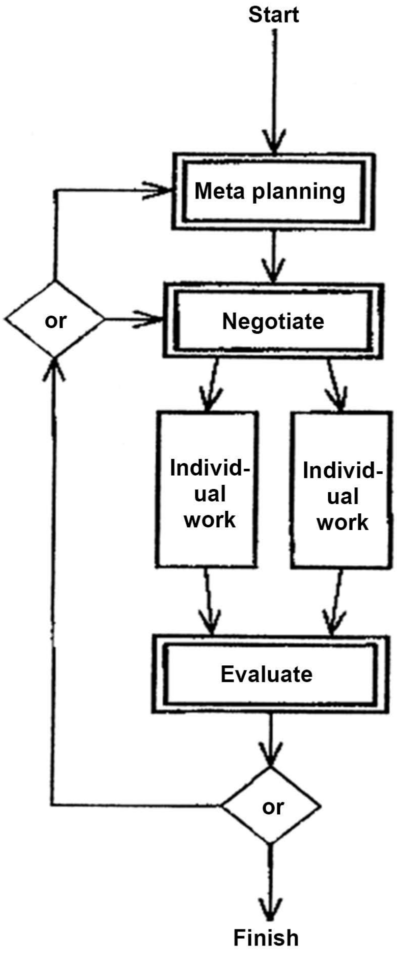

Figure 6. A model of design collaboration [39].

Following the discussion of these initial insights and the follow-up focus group with participants, we read through the transcripts, highlighted meaning chunks, underlined some interesting terms, and jotted down memos and questions that jumped to mind while reading. A second reading of the transcript and analysis of the images captured followed this and initial codes, or what Attride-Stirling [42] refers to as basic themes, were created. These initial codes were derived from participants' own responses and were not established prior to the analysis. Initial labels were then developed by grouping codes with similar attributes and properties together. Finally, these initial labels that together present an argument or a position within the context of the analysis and with the research questions in mind were grouped together to form the emergent themes [42]. Within each iteration, themes were compared to the actual data, which resulted in either expanding or collapsing of themes. Continued review of the literature resulted in the location of observation research done very similar to this work, which resulted in theme categories that heavily matched our emergent themes [42]. We then adopted and used these themes instead, as data analysis continued.

8. This Case Study: Research Environment

8.1. Description of Software Design Project Underway

This design team had begun the design process of this cloud-based software application approximately 6 weeks before our observations began. They met after and during work hours, when convenient, as this project was a tool they envisioned to help their clients, and not one commissioned by clients. They had an internal deadline in mind, which coincided with the release of an Amazon software service, but only worked on it when time allowed. Both designers already had a high-level understanding of the design to date and the design meeting we observed began to flesh out more details. EXPERT already had an idea of how to implement these details, but opted to use elements of this project as a teaching tool, as we later found out in our focus group. A statement of by EXPERT supports this during our observation—"***I am just guiding you. You do it and I'll let you know what you're doing wrong.***"

8.2. Design Space

Internal design projects such as this, emerged from employees and were not commissioned by a client and could therefore not be focused on during work hours or at client sites. Therefore, for projects they envisioned themselves, employees of Code Theory would often meet at a local restaurant to have their design meetings. EXPERT and NOVICE set up their laptops and worked side by side. When asked if this was their normal setup, they confirmed and referred to a common software design practice (within Agile Programming) called "pair programming", which is a setup where two programmers work on one computer. In this case, one computer displayed design notes, while the other displayed the code in question. This side-by-side setup can be seen in the image of Figure 7.

9. This Case Study: Preliminary Findings

9.1. Resulting Collaboration Design Events

As the design meeting progressed, the designers often moved from low-level to high-level design vocabulary as it related to the project. When discussing low-level design elements, they were writing code bits and contemplating literal command names, and when discussing high-level elements, they were thinking of how the user would interact with the tool being designed and the elegance of execution of the anticipated code.

Combining Cross and Cross' [38] social occurrences with Kvan's [39] distinction of design events, there were a total of 183 distinct design events observed, depicted in Table 1. Most of these events were of the type PAU: problem analyzing and understanding (84% or 46%) and 34% or 40% of the CGA events, which occurred during dictator collaboration events. These dictator collaboration events entailed EXPERT using questions to elicit more thought from NOVICE and/or wanting to understand NOVICE's rationale about the idea or

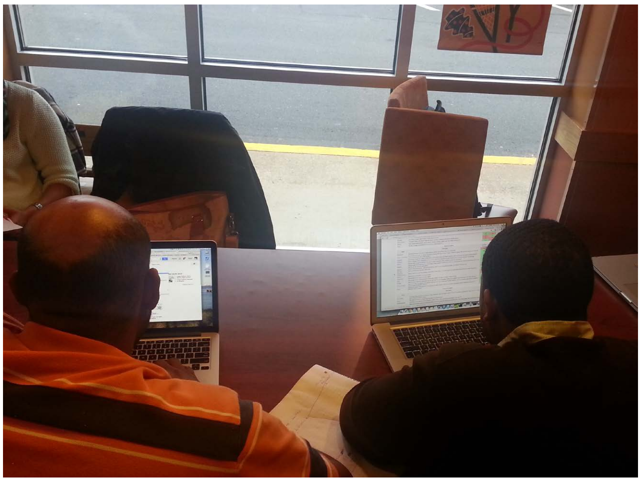

Figure 7. EXPERT and NOVICE sitting side-by-side, pair programming.

Table 1. Type and number of design events, with Mentor Design Event and Mentor Collaboration Type newly added.

Type of Design Event [38]	Collaboration Type [39]	Occurrences
Roles and Relationships (RR)	Mutual	16
Planning and Acting (PA)	-	0
Information Gathering and Sharing (IGS)	Exclusive	19
Problem Analyzing and Understanding (PAU)	Dictator	84
Concept Generation and Adopting (CGA)	Dictator	34
Conflict Avoiding and Resolving (CAR)	Mutual/Dictator	7
***Mentoring (M)**	**Mentor/Dictator**	**23**
Total		183

approach he presented at the time of the design event. Some examples of these questions and elicitations include statements like the following:

"*I don't understand this.*"

(*repeatedly stating this while point to the object of discussion, a diagram, a piece of code, etc.*)

"*I don't understand… I'm lost. I still don't understand.*"

"*What is a config file here? → How does it work? →How will we use it?*"

"*What are the inputs and how with they* [*the user*] *issue the command?*"

"*But let's start back further, how do you plan on...*"

"*Alright, this is too low-level, let's back up.*"

"*Forget the drawing, let's start over.*"

At times, NOVICE issued a few PAU events, but this was rare—"What is confusing, this portion of the drawing or the way the drawing was created?" and some CAR: conflict avoiding and resolution events (7), such as "I thought we agreed."—to which EXPERT responded, "Good point." However, then NOVICE would often agree with EXPERT. But it was not clear if he was avoiding conflict or agreeing because he understood. This relates to the CGA: concept generation and adoption events. As a result of EXPERT using the dictator collaboration events as "teaching moments", he would often ask NOVICE to close his laptop and would further explain a concept. NOVICE would indicate his understanding, as if to adopt or agree, and the design session would continue. These explanations occurred 34 times and **Figure 8** illustrates those occurrences when EXPERT asked NOVICE to close his laptop and listen.

There were 19 IGS: information gathering and sharing events. Most of these were initiated by either designer who simple began searching for a programming term or practice, or referencing a concept to be reminded of from a past design drawing. Once found, they would resume the design conversation and include the found understanding in the conversation. There were 16 RR: role and relationship events, where it as clear EXPERT was "in-charge". These events confirmed and were initiated by EXPERT's dictator collaboration events. This initiation came from statements like the following:

"*OK, what's the command? Write it.*"

"*I don't understand. Draw it out.*"

Figure 9 shows different instances when both EXPERT and NOVICE draw and used boxes to illustrate, explain, and/or share a design idea.

Several (23) M: Mentor collaboration events occurred as well. These were evident from statements made by EXPERT to NOVICE, some of which are listed below:

"*Design is assigning intelligence to code.*"

"*This is design, don't take it personal.*"

"*Design is about drawing boxes, not writing code.*"

"*That is not the way I would do it, so let's back up.*"

"*What I was looking for was….*"

(referring to the way he would have preferred an answer to one of his questions)

At times NOVICE confirmed his protégé status with statements like "I feel like I'm back in school." As a result, we introduce a new design event to the Cross and Cross's [38] list called the Mentor design event, and a new design collaboration type called Mentoring to Kvan's [39] Model of Design Collaboration.

9.2. Discussion

It was clear that many of the design collaboration events which occurred during this design meeting indicated that EXPERT was in-charge and often directed portions of the design discussion and many design decisions as teaching or mentoring moments or caused the direction of the conversation to elicit more thought and understanding from NOVICE. Initial analysis might suggest these moments fit the description of dictator collaborative events. However, the role and actions of the designer in charge reach beyond simply controlling the activities of the collaboration. They extend instead into gestures of teaching,

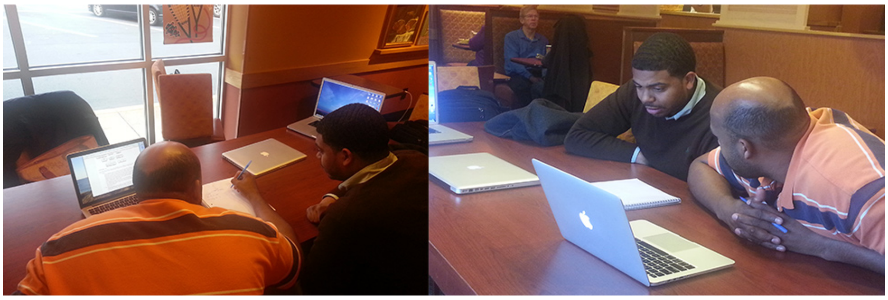

Figure 8. Design events where EXPERT asks NOVICE to close his laptop and listen.

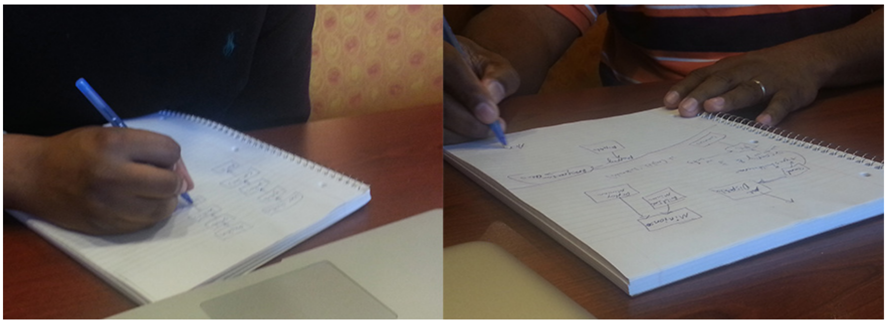

Figure 9. Both NOVICE (left) and EXPERT (right) draw boxes as they design. "*Design is about drawing boxes, not writing code.*"—EXPERT.

nurturing, mentoring, and guiding, as stated by EXPERT during one of the collaborative events and therefore suggested in the title of this article. As such data analysis suggests a fourth collaborative event be added to Kvan's Model of Design Collaboration and that is Mentoring. However, unlike the three types of collaborative events introduced by Kvan [39], the Mentoring relationship is optional. Therefore, we have chosen to represent it with a dotted-line at the various points in the diagram where it can occur as can be seen in Figure 10. Following from this, we also suggest a seventh potential type of collaboration be added to Cross and Cross' [38] list of social occurrences in design and that is Mentoring (M).

9.3. This Case Study: Conclusion

In summary, design is a social and collaborative process and receives an abundance of attention from researchers to understand how it is done, what interactions support it, and how it can be optimized. As software technology becomes more and more ubiquitous, it is crucial to understand and optimize the software design process. This is especially important to inform more smooth transitions from training to become a software engineer (design, code, test) to being hired with the objective of applying the software engineering skills learned. As can be seen by the case study presented here, guiding and mentoring plays a huge role in the interactions between novice and expert software engineers. So much so, that this interaction should be distinctively present in the models and descriptions of collaborative software design events and within collaborative design

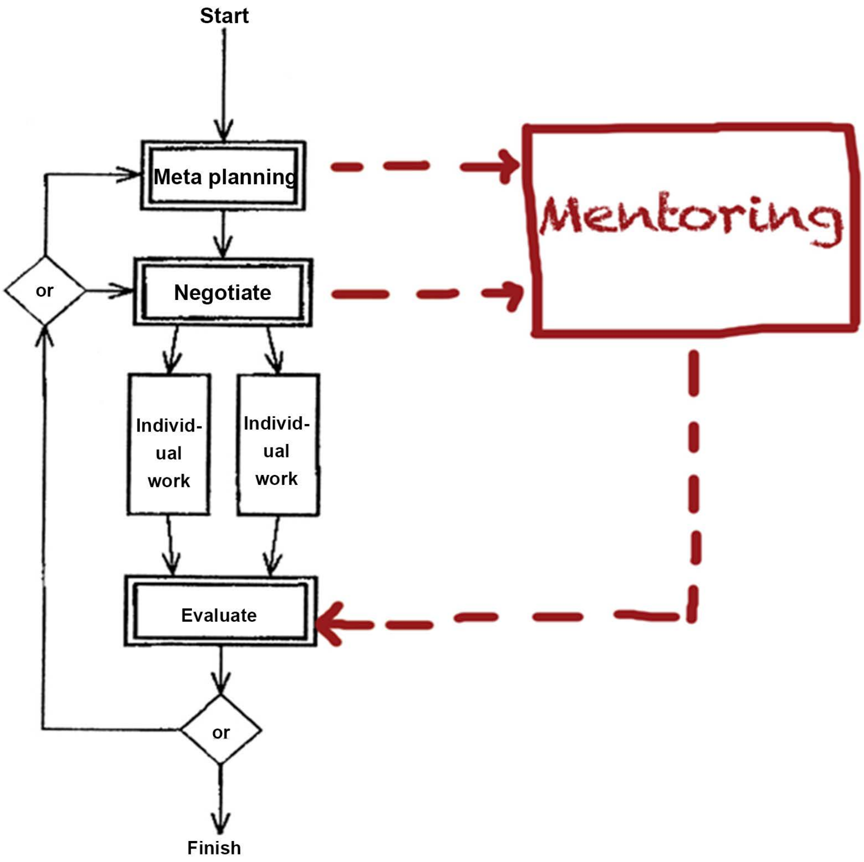

Figure 10. A model of design collaboration with mentoring.

interactions more broadly. The data analysis described above provides a foundation for this addition and paves a way for more future work in this area.

9.4. This Case Study: Limitations and Constraints

As mentioned previously, due to time constraints we were only able to find one firm that had a scheduled group meeting at the time we were ready to collect data. This limitation also affected the amount of data we could collect. We were only able to conduct one observation and one focus group in addition to the data from the online questionnaire and have enough time to analyze and synthesize our findings. Finally, as is the case with case study designs, the findings and explanations reported here do not necessarily reflect the behavior and interactions of other entities even though it may be suggestive of what may be found in similar organizations and design teams.

9.5. Implications and Future Work

Empirical evidence of software design teams adds to the literature on how design teams function in general. Uncovering the details of this functioning when mentoring is involved is a profound subject of exploration. We have only scratched the surface here and would like to continue observing this and other design teams from Code Theory. Our recruitment efforts for this study exposed us to many additional design companies, some who focus on web design, while others focus on animation and graphics. We would like to observe these teams as well and compare our findings. Additionally, Gero and McNeil's [40] suggested the design events (e.g. making a design decision) may take very discrete events lasting shorts amounts of time, mere seconds, and that experts may take half as much time as a novice to conduct the same event. We would like time to compare these events in continued data analysis. Mentorship is extremely important to the culture, development, and success of individuals and the companies for which they work. It is our hope to further this exploration and contribute to their growth and achievement.

References

[1] Friedman, K. (2003) Theory Construction in Design Research: Criteria: Approaches, and Methods. *Design Studies,* **24***,* 507-522. https://doi.org/10.1016/S0142-694X(03)00039-5

[2] Leitner, M., Innella, G. and Yauner, F. (2013) Different Perceptions of the Design Process in the Context of Design Art. *Design Studies,* **34**, 494-513. https://doi.org/10.1016/j.destud.2013.01.006

[3] Warr, A. and O'Neill, E. (2005) Understanding Design as a Social Creative Process. *Proceedings of the* 5*th Conference on Creativity & cognition* ACM, New York, 118-127. https://doi.org/10.1145/1056224.1056242

[4] Luck, R. (2012) Kinds of Seeing and Spatial Reasoning: Examining User Participation at an Architectural Design Event. *Design Studies,* **33***,* 557-588. https://doi.org/10.1016/j.destud.2012.06.002

[5] Tang, A., Aleti, A., Burge, J. and van Vliet, H. (2010) What Makes Software Design Effective? *Design Studies,* **31**, 614-640. https://doi.org/10.1016/j.destud.2010.09.004

[6] Wiltschnig, S., Christensen, B.T. and Ball, L.J. (2013) Collaborative Problem-Solution Co-Evolution in Creative Design. *Design Studies,* **34***,* 515-542. https://doi.org/10.1016/j.destud.2013.01.002

[7] Détienne, F. (2006) Collaborative Design: Managing Task Interdependencies and Multiple Perspectives. *Interacting with Computers,* **18***,* 1-20. https://doi.org/10.1016/j.intcom.2005.05.001

[8] McDonnell, J. (2012) Accommodating Disagreement: A Study of Effective Design Collaboration. *Design Studies,* **33**, 44–63. https://doi.org/10.1016/j.destud.2011.05.003

[9] Sonnenburg, S. (2004) Creativity in Communication: A Theoretical Framework for Collaborative Product Creation. *Creativity and Innovation Management,* **13***,* 254-262. https://doi.org/10.1111/j.0963-1690.2004.00314.x

[10] Walsh, G., Druin, A., Guha, M.L., Bonsignore, E., Foss, E., Yip, J.C. and Brown, R. (2012) DisCo: A Co-Design Online Tool for Asynchronous Distributed Child and Adult Design Partners. *Proceedings of the* 11*th International Conference on Interaction Design and Children*, ACM, 11-19. https://doi.org/10.1145/2307096.2307099

[11] Doboli, A., Umbarkar, A., Subramanian, V. and Doboli, S. (2014) Two Experimental Studies on Creative Concept Combinations in Modular Design of Electronic Embedded Systems. *Design Studies,* **35***,* 80-109. https://doi.org/10.1016/j.destud.2013.10.002

[12] Linsey, J.S., Green, M.G., Murphy, J.T., Wood, K.L. and Markman, A.B. (2005) Collaborating to Success: An Experimental Study of Group Idea Generation Techniques. *Proceedings of ASME Design Theory and Methodology Conference*, Long Beach, CA.

[13] McLaren, S.V. and Stables, K. (2008) Exploring Key Discriminators of Progression: Relationship between Attitude, Meta-Cognition and Performance of Novice Designers at a Time of Transition. *Design Studies,* **29***,* 181-201. https://doi.org/10.1016/j.destud.2007.10.001

[14] Perttula, M. and Sipila, P. (2007) The Idea Exposure Paradigm in Design Idea Generation. *Journal of Engineering Design,* **18***,* 93-102. https://doi.org/10.1080/09544820600679679

[15] Teo, L.K.C. and Waugh, R.F. (2010) A Rasch Measure of Fostering Creativity. *Creativity Research Journal,* **22***,* 206-218. https://doi.org/10.1080/10400419.2010.481534

[16] Begel, A. and Simon, B. (2008) Struggles of New College Graduates in Their First Software Development Job. *Proceedings of the* 39*th SIGCSE Technical Symposium on Computer Science Education*, Portland, OR, 12-15 March 2008, 226-230. https://doi.org/10.1145/1352135.1352218

[17] Adams, R.S., Daly, S.R., Mann, L.M. and Dall'Alba, G. (2011) Being a Professional: Three Lenses into Design Thinking, Acting, and Being. *Design Studies*, **32**, 588-607. https://doi.org/10.1016/j.destud.2011.07.004

[18] Goldschmidt, G. and Rodgers, P.A. (2013) The Design Thinking Approaches of Three Different Groups of Designers Based on Self-Reports. *Design Studies*, **34**, 454-471. https://doi.org/10.1016/j.destud.2013.01.004

[19] Gabora, L. (2002) Cognitive Mechanisms Underlying the Creative Process. *Proceedings of the Fourth International Conference on Creativity and Cognition*, Loughborough University, Loughborough, 13-16 October 2002, 126-133. http://www.vub.ac.be/CLEA/liane/papers/CandC/CandC.htm

[20] Liedtka, J. (2015) Perspective: Linking Design Thinking with Innovation Outcomes through Cognitive Bias Reduction. *Journal of Product Innovation Management*, **32**, 925-938. https://doi.org/10.1111/jpim.12163

[21] Lu, C.C. (2015) The Relationship between Student Design Cognition Types and Creative Design Outcomes. *Design Studies*, **36**, 59-76. https://doi.org/10.1016/j.destud.2014.08.002

[22] Sonnenwald, D.H. (1996) Communication Roles that Support Collaboration during the Design Process. *Design Studies*, **17**, 277-301. https://doi.org/10.1016/0142-694X(96)00002-6

[23] Strimel, G.J., Kim, E., Cantu, D. and Grubbs, M.E. (2017) Examining Design Cognition of Secondary and Post-Secondary Engineering Students. *Council on Technology and Engineering Teacher Education Proceedings*.

[24] Grubbs, M.E., Strimel, G.J. and Kim, E. (2018) Examining Design Cognition Coding Schemes for P-12 Engineering/Technology Education. *International Journal of Technology and Design Education*, **3**, 1-22. https://doi.org/10.1007/s10798-017-9427-y

[25] Council on Technology Teacher Education and International Technology and Engineering Educators Association (2016) Characterizing Design Cognition of High School Students: Initial Analyses Comparing Those with and without Pre-Engineering Experiences.

[26] Ke, F., and Im, T. (2014) A Case Study on Collective Cognition and Operation in

Team-Based Computer Game Design by Middle-School Children. *International Journal of Technology and Design Education*, **24**, 187-201. https://doi.org/10.1007/s10798-013-9248-6

[27] Koestler, A. (1964) The Act of Creation. Dell, New York.

[28] Gennari, J.H. and Reddy, M. (2000) Participatory Design and an Eligibility Screen Tool. *Proceedings of the AMIA Symposium*, Los Angeles, 4-8 November 2000, 290-294.

[29] Chiu, M.-L. (2002) An Organizational View of Design Communication in Design Collaboration. *Design Studies*, **23**, 187-210. https://doi.org/10.1016/S0142-694X(01)00019-9

[30] Matthews, B. and Heinemann, T. (2012) Analysing Conversation: Studying Design as Social Action. *Design Studies*, **33**, 649-672. https://doi.org/10.1016/j.destud.2012.06.008

[31] Dong, A., Kleinsmann, M.S. and Deken, F. (2012) Investigating Design Cognition in the Construction and Enactment of Team Mental Models. *Design Studies*, **34**, 1-33. https://doi.org/10.1016/j.destud.2012.05.003

[32] Torrance, E.P. (2002) The Manifesto: A Guide to Developing a Creative Career. Ablex, West Westport, CT.

[33] Hester, J.P. and Setzer, R. (2013) Mentoring: Adding Value to Organizational Culture. *The Journal of Values-Based Leadership*, **6**, Article 4. http://scholar.valpo.edu/jvbl/vol6/iss1/4

[34] Glesne, C. (2011) Becoming Qualitative Researchers: An Introduction. 4th Edition, Pearson, Upper Saddle River, NJ.

[35] Yin, R.K. (2003) Case Study Research: Design and Methods. 3rd Edition. SAGE, Thousand Oaks, CA.

[36] Baxter, P. and Jack, S. (2008) Qualitative Case Study Methodology: Study Design and Implementation for Novice Researchers. *The Qualitative Report*, **13**, 544-559. http://www.nova.edu/ssss/QR/QR13-4/baxter.pdf

[37] Fidel, R. (1984) The Case Study Method: A Case Study. *Library and Information Science Research*, **6**, 273-288.

[38] Cross, N. and Cross, A.C. (1995) Observations of Teamwork and Social Processes in Design. *Design Studies*, **16**, 143-170. https://doi.org/10.1016/0142-694X(94)00007-Z

[39] Kvan, T. (2000) Collaborative Design: What Is It? *Automation in Construction*, **9**, 409-415. https://doi.org/10.1016/S0926-5805(99)00025-4

[40] Gero, J.S. and McNeill, T. (1998) An Approach to the Analysis of Design Protocols. *Design Studies*, **19**, 21-61.

[41] Moghaddam A. (2006) Coding Issues in Grounded Theory. *Issues in Educational Research*, **16**, 52-66. http://www.iier.org.au/iier16/moghaddam.html

[42] Attride-Stirling, J. (2001) Thematic Networks: An Analytic Tool for Qualitative Research. *Qualitative Research*, **1**, 385-405. https://doi.org/10.1177/146879410100100307