Supporting Remote Real-Time Expert Help: Opportunities and Challenges for Novice 3D Modelers

Parmit K. Chilana[1], Nathaniel Hudson[2], Srinjita Bhaduri[3], Prashant Shashikumar[4] and Shaun Kane[5]

[1,4] Simon Fraser University
Burnaby, BC, Canada
{pchilana, pshashik}@sfu.ca

[2] Ross Video
Ottawa, ON, Canada
me@nathanielh.com

[3,5] University of Colorado-Boulder
Boulder, CO, USA
{srinjita.bhaduri, shaun.kane}@colorado.edu

***Abstract*—We investigate how novice 3D modelers can remotely leverage real-time expert help to aid their learning tasks. We first carried out an observational study of remote novice-expert pairs of 3D modelers to understand traditional chat-based assistance in the context of learning 3D modeling. Next, we designed MarmalAid, a web-based 3D modeling tool with a novel real-time, in-context help feature that allows users to embed real-time chat conversations at any location within the 3D geometry of their models. Our user study with 12 novices who used both MarmalAid's real-time, in-context chat and an external chat tool to seek help, showed that novices found the real-time, in-context chat to be more useful and easier to use, and that experts asked for fewer clarifications, allowing the novices to ask more task-related questions. Our findings suggest to several design opportunities to utilize and extend the real-time, in-context help concept in 3D modeling applications and beyond.**

Keywords—real-time help; in-context help; software learnability; 3D modeling

I. INTRODUCTION

The proliferation of 3D printing, virtual reality (VR), and augmented reality (AR) applications has sparked wide interest in learning 3D modeling. Although beginner-friendly modeling tools, instructional materials, and training programs are increasingly available, learning 3D modeling can still be daunting [17,21]. Prior work has documented many usability and learnability problems in using complex 3D modeling software, including dealing with confusing terminologies, creating complex shapes, and interacting with unfamiliar 3D geometry [4,27,37].

To develop 3D modeling skills, some newcomers seek help directly from experts, rather than learning from static videos or tutorials [21]. A novice 3D modeler may attend workshops at the local library, where they can receive one-on-one assistance from a workshop instructor throughout the modeling process [21]. Other learners may join maker spaces and online communities to learn modeling techniques from their peers [30,36,42]. By working with experts, modelers can experience "over-the-shoulder-learning" [39] and can ask targeted questions that reflect their particular software, task, and 3D model.

Despite the benefits of one-on-one help, it is rarely available outside of formal learning environments. Although novices can seek help from remote experts by posting a question on a discussion forum or asking a friend over email, it can be difficult for the expert to provide help without direct access to the user's task, and without the ability to quickly ask follow-up questions [8]. Furthermore, when requesting asynchronous remote help online, hours or even days may pass before a response arrives [3].

In this paper, we investigate how remote, real-time expert can be designed for 3D modeling tasks and how this form of help is used and perceived by novice 3D modelers. We carried out an observational study of 6 novice-expert pairs where each novice built a 3D model while asking questions to a remote expert via web chat. We found that although the novice users benefited by directly interacting with an expert, they had trouble explaining the visual context of their modeling tasks, often prompting clarification questions from experts.

To support for remote, real-time communication with experts, and drawing inspiration from modern contextual help approaches [9,16,20,29], we designed *MarmalAid*, a web-based 3D modeling application with an embedded *real-time, in-context chat feature* (Fig. 1). MarmalAid allows users to easily share work-in-progress with experts by establishing a shared visual context, and enables users to start real-time, in-context conversations that target specific areas of their 3D model. To evaluate MarmalAid, we recruited another set of 12 novice 3D modelers to compare MarmalAid's real-time, in-context chat feature with an external chat application. Our results show that the majority of the participants found the real-time, in-context chat to be more useful and easier to use than the external chat. Experts asked for fewer clarifications when using the real-time, in-context chat feature, enabling the novices to ask more task-related questions. Participants' qualitative feedback confirmed their strong preference for MarmalAid's real-time, in-context chat.

The main contributions of this paper are: 1) the design of a real-time, in-context chat feature for enabling remote one-on-one synchronous communication between novice and expert 3D modelers; and 2) empirical insights into how novice 3D modelers use and perceive remote expert help via real-time, in-context chat vs. an external chat application and how it affects their 3D learning tasks. Although MarmalAid's help features were designed to support 3D modeling tasks, we reflect on how the lessons learned from this work can generalize to other creative design tasks and how the real-time, in-context help approach can augment other forms of software learning and troubleshooting.

II. RELATED WORK

A. Learning by Demonstration

There is a long history of HCI research exploring learning to use feature-rich software by demonstration. Some approaches have investigated the use of animated steps (e.g., [19, 31]), interactive, guided tutorials (e.g., [12, 15, 23]), and even complete application workflows and editing histories (e.g., [18]). Other work has explored video-based demonstrations and mixed multimedia forms of assistance (e.g., [7,16,33]). These approaches

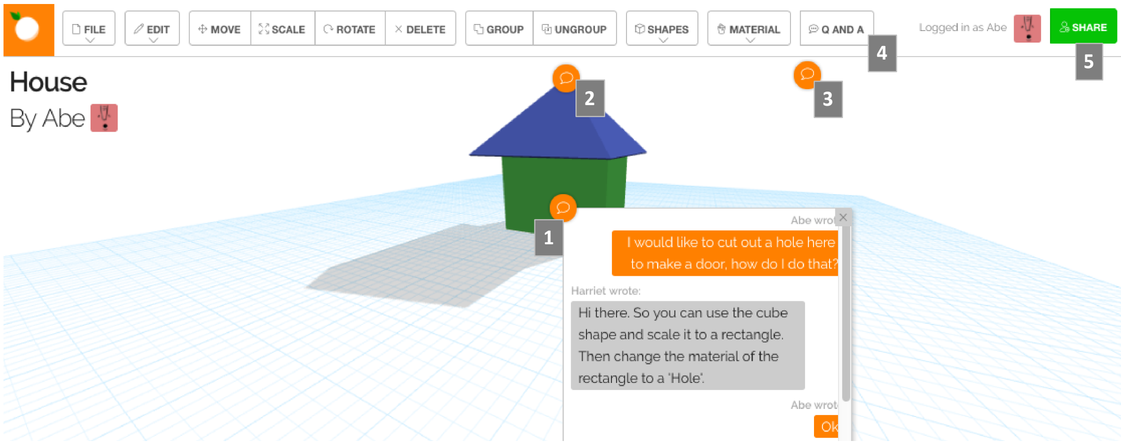

Fig. 1. MarmalAid user interface: (1) users can anchor chat windows to request remote assistance on any part of their model; (2) by clicking on the comment icon, the chat window collapses but remains attached; (3) document-level chat windows can also be added without a model-specific anchor; (4) the Q and A icon is used to create a new chat window; (5) the Share feature can be used to obtain a model-specific URL for sharing.

enable novices to follow steps created by expert users or application designers rather than scouring through static help materials. Despite the benefits of learning by demonstration, novices face a number of challenges in following the steps of experts [41], often giving up and seeking one-on-one help [21].

Recent research has explored how to enhance instructional materials with community contributions and embedded Q&A discussions (e.g., [5,24,40]). Our key motivation for MarmalAid was to investigate how one-on-one synchronous help can be offered within the application, in context of the users' tasks. We see in-context real-time chat features as complementing other forms of learning by demonstration and community-based help.

B. *In-Application Contextual Help*

HCI researchers have also long explored methods for providing contextual help, such as by attaching help messages to specific UI labels and controls within an application. Some of the prominent contextual help approaches include text and video-based tooltips [16], *Balloon Help* [11], pressing F1 over UI elements, choosing a command name to see animated steps [38], among others. These approaches usually require the help content to be pre-authored at design time, making the content limited to explaining the functionality of a specific UI widget or feature. In response to this limitation, recent work has explored the idea of crowdsourced contextual help, with systems such as *LemonAid* [9], *IP-QAT* [29] and *HelpMeOut* [20] where the in-context help can be dynamically created and maintained by the user community based on their actual application needs.

Although MarmalAid is inspired by these existing forms of contextual help, it offers two fundamental differences: 1) there is no pre-authored help content or ability to see previous community-authored questions—instead, MarmalAid provides help that is customized for a user's specific task and questions; and, 2) MarmalAid's real-time help can be directly anchored to part of a 3D model's geometry, raising the possibility of attaching Q&A to a broader set of anchors and referents within an application. As we will discuss in our findings, study participants asked proportionally more domain-specific questions (and fewer UI questions) when using Marmalaid's in-context help.

C. *Synchronous Remote Help*

Another area of HCI research that relates to the present work is the exploration of synchronous remote help-giving in learning and troubleshooting contexts. For example, Crabtree et al. [10] compared help given at a library help desk to help given at a call center of a printer manufacturing company, and found that users lacked precise technical knowledge of the machine that they used and had limited understanding of how to move from their current state to a solution. Similarly, studies of remote troubleshooting in the domain of home networking [34,35] have shown that users seeking help via telephone often struggle in providing precise descriptions of their issues, making it difficult for support specialists to understand and diagnose their problems.

Given the difficulty of diagnosing user- reported issues remotely, even when assistance can be provided in real-time, some have argued that shared visual context is necessary to facilitate conversational grounding during Q&A [14]. MarmalAid builds on this idea by providing a shared visual space for Q&A and by enabling users to spatially anchor questions to specific points of interest. Our empirical findings complement prior studies of remote help-giving and provide insights into how real-time remote help may be used during collaborative 3D modeling.

III. OBSERVATIONAL STUDY OF NOVICE-EXPERT PAIRS

The goal of our initial observational study was to better understand how novice users interact with a remote expert when learning how to create 3D models, and to identify the challenges faced by both novices and experts during this remote activity. Novices created 3D models using Autodesk *Tinkercad*, a web-based 3D modeling application, and talked with the expert via the *Slack* team chat application.

We recruited 10 participants from the local community via email to student lists on campus and posts to nearby makerspaces. Participants were all between the ages of 21 to 56. Based on self-report, we classified six participants (2 female) as novices, and four participants (1 female) as experts. Novice participants had little to no prior experience in 3D modeling, while the experts were familiar and/or proficient in using several CAD tools, such as *SketchUp*, *Blender*, *Rhino*, *Maya*, and *Solidworks*. Participant occupations included student and factory worker.

Each study session included one novice participant and one expert participant. Each novice only participated in one study session; two experts participated in one session only, and two experts participated in two study sessions. After an initial introduction, both novice and expert were placed in separate rooms. Each participant was seated at a computer that was running the Tinkercad and Slack applications. Expert participants were instructed to answer the novice's questions via Slack's text chat. Study sessions took place at a university research lab. Each session lasted one hour. Participants were compensated $15.

A. Procedure

Each novice participant completed two 3D modeling activities: an initial tutorial that introduced Tinkercad, followed by a self-guided 3D modeling task.

At the start of the study session, the facilitator introduced the novice participant to the goal of the study session. The novice then watched a 3-minute video tutorial that demonstrated the basics of Tinkercad, and then tried to reproduce the models shown in the tutorial using Tinkercad for a total of 10 minutes. Expert participants were given the option of completing the Tinkercad tutorial but were not required to do so.

After completing the tutorial, novice participants were shown an image of the 3D model and were instructed to recreate that model in Tinkercad. The test model was a set of 3D letter shapes; this model was designed so that novice participants would need to use all of Tinkercad's basic features, including placing 3D primitives (e.g., box, cylinder), changing the size and orientation of primitives, and combining primitives to produce more complex shapes. Novices were given 30 minutes to complete as much of the 3D modeling task as possible and were instructed to ask the expert for help via Slack chat. During this activity, they were asked to think-aloud so that the research team member could better observe what the novice user was doing.

Each novice and expert participant filled out a demographic questionnaire before beginning the study. At the end of the study, both participants were interviewed by the facilitator, and both participants filled out feedback questionnaires.

We captured participants' computer screens, and recorded audio of the think-aloud activity and follow-up interview. We also collected the 3D models created in Tinkercad and logged the Slack conversations between pairs of participants. Analysis of the collected data was done collaboratively by the research team using affinity diagrams and inductive analysis. Together, we identified the challenges that novice participants experienced in creating 3D models, observed patterns of help-seeking behavior between novices and experts, and analyzed feedback from participants about this type of collaborative learning activity.

B. Key Findings and Implications

Our analysis of observations, chat logs, and subjective feedback suggested that novice participants were positive about the idea of working remotely with an expert as they learned how to create 3D models but encountered some key challenges.

Challenges in Formulating Questions: The chat logs from the study showed that novices experienced difficulties in framing their questions, and in describing the current state of their 3D model. When novices encountered a problem in creating their model or manipulating Tinkercad's view, they often lacked the language to describe their problem. As a result, the expert participant struggled in understanding what the novice was asking and ended up asking multiple follow-up questions to understand the novice's help needs.

For example, one participant wished to move Tinkercad's camera to a specific location, but had difficulty describing exactly how they wished to position the camera, writing:

...the default camera view centers the camera over the center of the workplane. I'd like to be able to look directly down from the top left corner of the workplane so I can get a better idea of how my objects are positioned...(U04)

Expert participants asked follow-up questions in all six study sessions. In the above example, the expert asked several follow-up questions to clarify the novice's intent. In another example, the expert participant did not understand what the novice wished to do; asking a follow-up question resulted in the novice providing a clearer description of the goal:

U08: *How do I change the angle of a shape?*
Expert: *What exactly do you mean?*
U08: *To rotate in the 3D plane rather than just 90° up or down*

Post-activity interviews confirmed that both novices and experts struggled to communicate about specific problems. Five of the six novices explicitly mentioned the difficulty in articulating questions. One participant noted that she was aware that 3D modeling had specific terminology, but had difficulty articulating questions because she did not know that terminology:

...[Tinkercad] had some explicit terminology that made sense to me but was not immediately transferred to the expert…so, there were few rounds of back and forth clarifications... (U02)

The novices' difficulties in articulating questions, and experts in answering questions, suggests that there is an opportunity to improve remote help by providing more shared points of reference between the novice and expert.

Challenges in Conveying the Visual Context: In traditional, collocated help sessions, the expert can directly observe the novice's work; this shared visual context can help the expert to understand the novice's goals and can enable the expert to provide more contextually relevant feedback. For our observational study, we focused on understanding how the novice and expert participants working remotely might create workarounds to compensate for the lack of shared visual context.

Novice participants were shown how to capture screenshots and share them via Slack. Experts sometimes asked for a screenshot to help them understand what the novice was asking about. The exchange below shows how a participant had difficulty finding a menu item to change color and how the expert asked for a screenshot of the novice's view:

Expert*: if you click on the shape, the dropdown menu has a color select menu*

U06: *For some reason, I don't see it*

Expert: *hmmm can you send me a screenshot of the window?*

Expert: *...[After seeing the screenshot] you can click on the red circle above "solid" to open the color menu*

In some cases, novice participants sent screenshots even without any prompting from the expert, such as when asking about Tinkered features that they did not know the names of or when they did not know how to describe specific 3D modeling features like addition or subtraction of shapes. Overall, four of the six participants used screen-shots.

While sharing screenshots enabled the participants to create a shared visual context, this feature was something of a workaround and did not naturally fit into these participants' workflow. Several novice participants noted that creating and sharing screenshots required additional effort and offered suggestions for improving the process of sharing visual context:

> *Something more automated could have been better...Or have a playback for the model novice is building and the expert can look at it. Or like have a split-screen where expert can see in real-time and give feedback. (U01)*

Overall, our observations and interviews suggest that shared visual context is helpful in both asking for help and offering help. However, using external tools to create and share screenshots added some friction to the 3D modeling task. These findings suggest that enriching the shared visual context between novices and experts and integrating the Q&A process into the 3D modeling workflow, could provide many of the benefits of in-person help without requiring an in-person expert. This insight helped inform the design goals for MarmalAid, a new 3D modeling tool that explicitly supports real-time remote help.

IV. MarmalAid: Design of Real-Time, In-Context Help

Based on findings from our observational study, we designed MarmalAid to explore the benefits and challenges of offering real-time, in-context help within a new 3D modeling application. The design of MarmalAid was driven by two key goals: providing real-time, back and forth discussion of specific 3D modeling problems, and creating a shared visual context between two remote users, such as a novice and an expert.

A. Creating 3D Models

MarmalAid is designed to enable novices to create simple 3D models. MarmalAid's modeling tools are based upon those offered by Tinkercad. MarmalAid features a basic constructive solid geometry (CSG) modeling system in which predefined shapes (e.g., cubes, spheres, cylinders,) can be transformed by rotation, scaling and translation, and can be combined via addition or subtraction to create more complex shapes. To add holes and negative space to models, primitives can be marked as a *hole*; when a hole is combined with other primitives, the shape of the hole primitive is subtracted from the compound object.

B. Sharing the Visual Context

As novice participants in our observational study found the process of sharing screenshots to be cumbersome, MarmalAid allows a user to share their view of the 3D model in real time with a remote expert. MarmalAid offers novice users two mechanisms for sharing their models with others. At any point, a novice user can click the "Share" button (Fig. 1.5) and enter an expert's email address. This action will generate an email to the expert that contains a link to the active MarmalAid document. MarmalAid users can also share their current project by simply copying the document URL and pasting it into an email, chat message, or any other communication tool.

The expert can open the shared link to see the novice user's work in progress and tagged areas of the 3D model in the MarmalAid online editor. The expert's view is a live version of the novice's model: changes made by the novice will be shown to the expert in the shared view, and vice versa. The expert can further explore and edit the shared model and participate in the in-context chat conversations. These sharing features enable a remote expert to provide feedback about the novice participant's work, and about the current 3D model, without the need to install additional software or send 3D model files back and forth.

C. Adding Real-Time, In-Context Chat Conversations

MarmalAid's core help feature is its real-time, in-context help chat system. MarmalAid allows novice users to create real-time chat sessions with experts that are embedded directly into MarmalAid's user interface (UI), enabling a user to seek help without switching to a different work context.

Geometry-specific Conversations: Since our observational study showed that novice 3D modelers may have difficulty articulating questions because they lack the appropriate terminology, MarmalAid allows users to tag a specific part of a 3D model when asking a question. This feature allows the novice user to direct the expert's attention, even when the novice cannot clearly describe their problem (Fig. 1.1).

To activate the geometry-specific chat conversation, the user clicks on MarmalAid's "Q and A" button (Fig. 1.4), and then clicks any point in the 3D model to create a new chat window (e.g., Fig. 1.1). The chat window anchors to the particular point on the object, so that both users can see the chat in the same location. Chat windows are anchored to the 3D model and remain at the same location even if the camera is moved, or if the object itself is rotated, translated, or scaled. Users can minimize chat windows when not in use, and re-open them later (Fig. 1.2). Users can create multiple chat windows in the same document.

Document-level Conversations: If a user is not sure where to ask a question, they can still attach a chat window outside of the geometry of the 3D model, within the MarmalAid document window (Fig. 1.3). Users can discuss the whole model or ask general questions through a single chat window in the sidebar of the MarmalAid document window.

In-Application Notifications: To support collaborative editing, MarmalAid tracks all actions and chat messages made on the document and notifies all users when a comment has been added or the 3D model has been modified (via an in-app notification).

Note that the current version of MarmalAid assumes that the user knows who they want to share their model with and does not support matchmaking with new experts. Future versions of MarmalAid may provide increased support for sharing help requests on community-based sites, such as discussion forums.

V. Implementation of MarmalAid

MarmalAid is an entirely web-based application designed to run on any modern desktop web browser without any need to install software. MarmalAid has been tested in Safari 9.1.1, Chrome 55, Firefox 47.0.1, and Internet Explorer 11.0.

MarmalAid's server-side component is written using Python 3's Flask web framework and stores data in a SQLite database. This server-side component manages user accounts and 3D model files, controls permissions for viewing and editing models, and sends notifications. MarmalAid's server uses the

Socket.IO library for real-time communication between the server and clients.

MarmalAid's user interface and 3D modeling tools are implemented in HTML5 and JavaScript and run directly in the browser. MarmalAid's 3D viewport is implemented using the Three.js library and is rendered via WebGL, which allows for responsive hardware-accelerated 3D in the browser. Geometric manipulations of the 3D models, such as moving, scaling, and combining objects, is handled by the CSG.js (constructive solid geometry) library. MarmalAid's web client uses Socket.IO for real-time communication and Require.js to manage loading of the various component modules.

VI. User Study

To evaluate MamalAid's real-time, in-context help features, we conducted a user study in which we observed how novice 3D modelers work with a remote expert while learning 3D modeling. For this study, we created two variants of MarmalAid: one that included all of MarmalAid's integrated chat features (*In-Context Chat*), and another that included MarmalAid's shared view, but relied on an external chat application (*External Chat*).

The goals of this study were to assess the strengths and weaknesses of interacting with an expert while using real-time, in-context help; and, to investigate whether help-seeking strategies differ when help is presented in-context vs. in externally.

We recruited 12 participants (6F), aged 19-38, from a large university campus. Participants included undergraduate students, graduate students, and administrative staff from a range of departments (e.g., Computer Science, Engineering, Business, and Biology). We pre-screened participants to ensure that they did not have prior 3D modeling experience. The expert role was taken by a member of the research team. We decided to use the same expert for all participants and conditions to maintain consistency across the responses and the type of help provided.

A. Study Design

We used a within-subject design to minimize the impact of the known high variation among participants. To eliminate order effects, we randomized the order of the In-Context Chat and External Chat conditions using a Latin Square. Following the study task, participants were interviewed by the researchers.

Similar to our observational study, participants were asked to create 3D models to match a reference image. Their tasks were to construct two 3D models of castles (Fig. 2), which we determined to have approximately equal difficulty through pilot tests. The two castles were presented in random order.

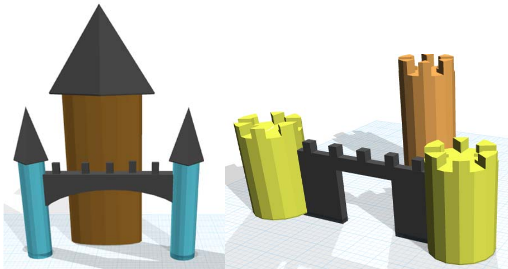

Fig. 2. Castles used in the two task conditions

For the External Chat condition, participants communicated with the expert via an external application. We chose *HipChat*, a web-based chat tool, as it had the required features but is not widely used. As in the observational study, novice participants were encouraged to ask questions of the expert through HipChat and were able to type questions and share screenshots (Fig. 4).

Our study considered the following measures for each participant: 1) Task performance, including the amount of the reference model the participant was able to complete; 2) help-seeking strategies, including how participants asked questions and had conversations with the expert; and 3) participants' subjective assessment of the two MarmalAid variants.

We conducted the study in an enclosed lab using a desktop computer running Windows 7 and the Chrome web browser.

B. Procedure

Each study session comprised four parts. For the first 10 minutes, participants completed the consent form and a demographic questionnaire, and followed a printed tutorial document that described MarmalAid's key features.

Next, participants completed the 3D modeling task for the two conditions (presented in counterbalanced order). Prior to the In-Context Chat condition, we showed participants how to use MarmalAid's Q&A feature and how to share their view with the expert. Prior to the External Chat condition, we showed participants how to use HipChat and how to share screenshots with the expert. The expert, portrayed by a member of our research team, was available throughout both of the task conditions. Participants were given 20 minutes to complete each modeling task.

Finally, participants answered questions about their experience in a 10-minute follow-up interview. The total study session took about 60-70 minutes. We compensated each participant with a $20 Amazon gift card.

C. Data Collection and Analysis

Throughout the study session, we recorded the participant's screen and audio recorded their interview responses. We collected all of the models created during each study session. During the two study tasks, we recorded usage of MarmalAid and chat transcripts through time-stamped activity logs.

To measure participants' performance on the study tasks, we analyzed the 3D models created during each of the study tasks. Each model was scored by a 3D modeling expert along two axes: accuracy of the model to the original reference image, and completion of the model. The expert scored each model on a separate 10-point scale for accuracy and for completion. Accuracy was scored based on use of the same primitive shapes, similar positioning of the shapes, and the characteristics of the shapes used (color, size, and aspect ratio). Completion was scored based on the number of components completed (castle base, pillars, walls).

To understand participants' help-seeking behavior across the study tasks, we collected and analyzed the text of the questions asked by the participants. We performed qualitative coding on each question to determine whether it was related to the 3D modeling task itself (e.g., "how can I combine objects to create a cone?") or to the use of the UI (e.g., "is there a copy and paste feature?"). Finally, to understand participants' subjective responses to using the two prototypes, we analyzed the post-task interviews. We used a bottom-up inductive analysis approach to identify broader themes within the participants' feedback.

D. Results

Task Completion and Accuracy: We designed our modeling tasks so that participants would be able to continue the modeling task for at least 20-minutes, regardless of their talent for creating 3D models. Indeed, no participant completely finished either of the modeling tasks during the 20-minute task period. On average, participants completed about 30% of each of the castle models, resulting in an average completeness score of 3. There was no significant difference of completion rate across two study conditions, according to a paired-sample t-test ($t=-0.55$, $p= 0.59$, two-tailed) and we did not observe any order effects.

Regarding the accuracy of the completed models, the models created in the In-Context Chat condition were marginally more accurate than models created in the External Chat condition, with an average accuracy score of 5.8 vs. 5.5, respectively. However, this difference in accuracy was not statistically significant ($t=1.07$, $p= 0.31$, two-tailed) and there were no order effects.

Questions Asked: Overall, participants asked more questions during the In-Context Chat condition than in the External Chat condition, averaging 6.01 vs. 5.10 questions, respectively. This difference was significant ($t=2.24$, $p\leq 0.05$, two-tailed).

We analyzed participants' to determine whether there were more questions about how to create the 3D model or about how to operate the modelling UI. We found that the majority of questions in all conditions were about how to perform actions within the 3D modeling task, rather than about operating the UI. In fact, on average, participants asked more than 4 times as many questions about 3D modeling than about the UI, averaging 4.55 and 0.97 questions per study session, respectively. This difference was significant ($t=3.92$, $p\leq 0.01$, two-tailed).

Additionally, study condition seemed to affect the types of questions asked. Participants asked more 3D modeling questions during the In-Context Chat condition than in the External Chat condition, averaging 4.90 vs. 3.90 questions, respectively. This difference was significant ($t=2.34$, $p\leq 0.05$, two-tailed). This finding suggests that participants may have been more focused on the 3D modeling task when using the in-context help chat.

We analyzed the topics of questions in each category. The 3D modeling questions included questions about manipulating the 3D workspace (i.e., the workplane), how to compose complex shapes, and how to create holes. With the brown and blue castle (Fig. 3, left), the top question was usually about creating a cone, as MarmalAid did not offer a cone primitive. For the yellow and brown castle (Fig. 3, right), the majority of participants asked about how to create a window through the castle wall, as they were not experienced in using holes.

UI questions included asking for explanations of how certain features worked, such as object grouping, and questions about whether certain features were available, such as copy-and-paste.

Conversational Structure: We analyzed the conversations across each study condition both from the perspectives of novices asking questions and the expert providing answers.

A key observation that we made was that the number of clarification questions asked by the expert varied across the study conditions. Although participants in the In-Context Chat condition asked more questions about 3D modeling, the expert asked fewer follow-up questions in the In-Context Chat condition than in the External Chat condition, averaging 5.18 vs 3.09 questions, respectively. This difference was significant ($t=-2.24$, $p\leq 0.05$, two-tailed), suggesting that participants may have asked fewer ambiguous questions when using In-Context Chat.

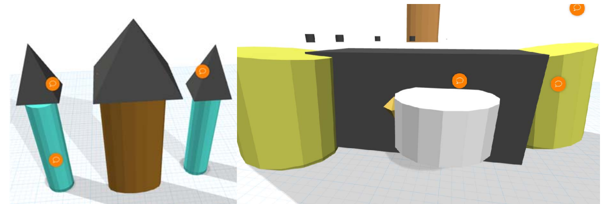

Fig. 3. Castle models created during the Internal Chat condition (left from P01; right from P03). The orange spheres indicate several chat windows at different locations.

In the External Chat condition, participants sometimes used screenshots to provide additional context for their questions (Fig. 4). On average, participants shared 2 screenshots per study task in this condition.

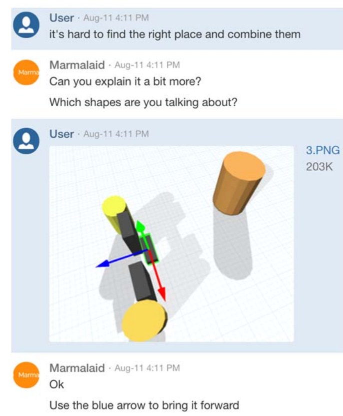

Fig. 4. Example conversation from the External Chat condition. The novice's initial question is considered ambiguous by the expert, and the expert requests a screenshot to provide further context for the question.

The length of conversations also differed between conditions. Participants wrote more words in the In-Context Chat condition than in the External Chat condition, averaging 336.45 words vs. 269.35 words, respectively. This difference was significant ($t=-2.55$, $p\leq 0.05$, two-tailed). This difference in the length of conversations may reflect greater engagement from participants when using in-context help. Longer conversations in the In-Context Chat condition may also be influenced by MarmalAid's ability to create multiple simultaneous chat windows. Nearly all participants (10 out of 12) in the In-Context Chat condition created multiple chat windows, creating three windows on average (examples shown in Fig. 3).

Subjective Feedback: At the end of each study condition, we asked participants to rate their experiences and preferences. Participants' ratings on Likert-style questions indicate that participants significantly preferred In-Context Chat ($w= 37$, $p\leq 0.05$), and found In-Context Chat to be more useful ($w=27$, $p\leq 0.01$). Participant responses are shown in Fig. 5.

We asked participants to rate the difficulty of modeling in each condition. On a scale of *Very Difficult* to *Very Easy*, most participants commonly rated the modeling tasks as *Somewhat Difficult*, which is not surprising given that this was the first time the participants had created 3D models. A Wilcoxon signed-rank test found no difference between the perceptions of difficulty under both conditions (w=6, p=0.33).

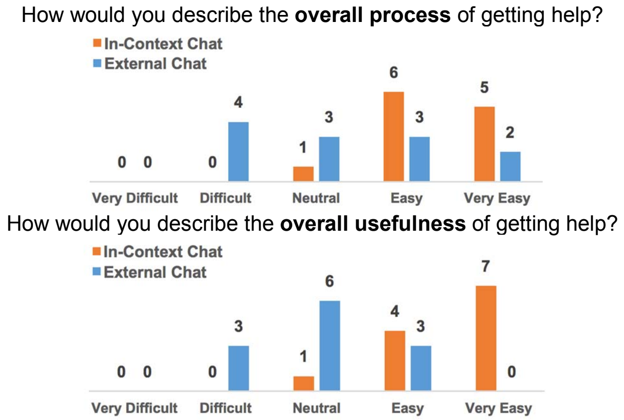

Fig. 5. Participants' ratings for overall preference and usefulness.

User Perspectives of Help Quality: We asked participants about task difficulty, and what they liked and disliked about each user interface. Participants pointed out that they had difficulty in assembling objects, and in navigating around the different views of the 3D model. When seeking help, participants noted that they lacked the correct vocabulary to ask their questions.

Participants noted several advantages to MarmalAid's real-time, in-context help. First, they noted that in-context help created a shared visual context that made it easier to ask questions:

> ***P04:*** *I found the interaction here [in-context chat] to be more intuitive and straight-forward because with the other [external] chat, I had to explain exactly what I meant and … it was hard to get the exact shape across … for example, with the castle ridges on top, I didn't know how to explain that, so I had to be creative and explain it like a chess piece…made it more difficult.*

The ability to anchor questions on a 3D model enabled participants to direct the expert's attention, and allowed the novice and expert to separate out different discussion threads:

> ***P11:*** *When you set up a chat for one particular shape, it implicitly implies that the context is for this shape…so I like that you can sort of create a specific zone [for the questions]*

Sometimes, the lack of shared visual context resulted in miscommunication in the chat between the novice and the expert:

> ***P12:*** *I was trying to [explain] how to create the smokes on the cylinder…ended up with the option of making holes with the square…that wasn't the desired result because it was difficult to explain what I was trying to do [in HipChat]…I ended up with something exactly opposite of what I wanted…just frustrating.*

Participants noted several additional benefits of the in-context chat, such as reducing the need for context switching and receiving notifications as soon as the expert replied:

> ***P08:****[With HipChat] I had to stop my train of thought, go to another tab, maybe take a screenshot, and then attach it…sometimes the expert would be waiting for me to say something and I would just be on the other tab…*

MarmalAid's in-context chat enabled participants to ask questions without switching tasks. Also, because MarmalAid notified users as soon as the expert provided a response, participants did not miss out on responses from the expert. In contrast, we observed that five participants neglected to look at the chat window during the External Chat condition, even after the expert had provided a response.

Suggested Improvements: Participants suggested several additional features for future versions of MarmalAid, including the ability to ask voice-based questions, features for direct annotations on the model, and the ability to selectively share parts of a model, rather than sharing the entire model.

While ten of the twelve participants preferred MarmalAid's in-context chat to the external chat, two preferred the external chat. Both of these participants used external chat for their second modeling task and stated that they asked for more assistance with external chat. One of these participants mentioned that in-context chat window felt out of place in the modeling tool and took up too much screen space. The other participant noted that chat windows took up screen space even after the questions were answered. While the In-Context Chat condition allowed users to minimize or close chat windows, we encouraged participants to retain their chat windows throughout the study task.

Reuse Value and Broader Applicability: During the interviews, most participants (10 out of 12) said that they would be likely to keep using MarmalAid's in-context Q&A feature if it were available. Participants also suggested that MarmalAid's feedback model could be extended to other types of applications such as image editing programs, statistics tools, and programming environments. Participants also suggested that this type of collaborative help could be useful when working collaboratively, such as when soliciting feedback on a design project.

One participant compared the ability to work with an expert in MarmalAid to following along with an expert in online videos, but noted that the novice cannot directly ask for help when watching a video, as they can with MarmalAid:

> ***P05:*** *YouTube is good so you can mimic what they [experts] are doing… but, you can have differences in version and speed and understanding… [experts] can use shortcuts or use functions which you may not know…with real-time [help], the expert can really break it down for you.*

Participants noted that MarmalAid's in-context chat can benefit novices by making it easier to ask questions and access experts, but also that MarmalAid would make it easier to provide help when taking on the expert role.

VII. DISCUSSION

This paper contributes a new design for real-time, in-context help for 3D modeling, and provides empirical insights into how novice 3D modelers use and perceive remote real-time expert help via real-time, in-context chat vs. an external chat application and how it affects their 3D learning tasks. We now reflect on key lessons learned from this research and design opportunities in HCI for improving real-time, in-context help systems.

A. Supporting 3D Modeling with Real-Time, In-Context Q&A

Participants in our study encountered various challenges in learning 3D modeling, similar to those discussed in prior work (e.g., [4,17,21,27,37]), including problems in understanding 3D geometry and developing an appropriate domain-related vocabulary. Although our findings were not conclusive about whether real-time, in-context help provides any immediate performance improvement for novice 3D modelers, our findings suggest that in-context help enables novices to spend less time clarifying questions and more time asking additional questions. There is clear indication that users can ask more learning-related questions when they have to spend less time on explaining their context to the helper. Furthermore, users strongly preferred the in-context chat experience and found it more useful than external c hat, in part because they were able to localize conversations with experts at the relevant location within a 3D model.

Our findings have immediate applicability to improving help in settings such as 3D modeling workshops, classrooms, makerspaces, and online design sharing and troubleshooting forums (e.g., *Thingiverse* [1]), where novice 3D modelers already work with more experienced users to solve problems and learn about new features. Future work can tease out the learning and performance benefits of in-context help through long-term field deployments of MarmalAid in such settings and can explore the differences between novice and more expert users. In the future, we may be able to automatically assign certain helpers based on their skills and their experience working on similar problems. Since MarmalAid's features mostly support primitive 3D modeling tasks, it would be useful to expand the scope of the application to support more polygon-oriented 3D shape modeling, and to assess the usefulness of in-context, real-time help for more advanced 3D modeling tasks.

B. *Design Opportunities for Real-Time, In-Context Help*

Although our main goal in this work was to understand and design remote real-time help to support 3D modeling tasks, our participants expressed enthusiasm for expanding this approach to other application domains. Below we discuss opportunities in to further develop and expand the idea of real-time, in-context help for tasks beyond 3D modeling.

Allowing for domain-related queries in-context: A key finding from our study was that participants asked a larger proportion of questions about 3D modeling when using real-time, in-context help (rather than asking questions about the UI). This differs from many previous in-context help systems (e.g., [9,16,20,29]) that focused on teaching the user interface rather than the domain of the task. Our findings suggest that more work is needed to understand users' preferences in soliciting domain-specific contextual help. In particular, participants in our study liked to point at objects when asking a question to direct attention to that area. This notion of situated questions could apply in other domains, but would likely differ significantly in tasks such as programming where contextual help needs are very different [6]. In the future, we may develop approaches to automatically detect "location" across different task domains.

Expanding the scope of on-screen referents: Another lesson we learned from this work was that users appreciate being able to anchor their questions to a specific problem area, but that some questions may reference multiple objects. How can users describe their referents in these situations? Participants in our study suggested adding annotation tools that would enable them to point at various items on screen, and to explain their problems at different levels of granularity. Future work can explore what on-screen referents are needed for more complex 3D scenes, as well as how these referents can be automatically detected. For example, LemonAid [9] can detect on-screen referents automatically and attaches crowdsourced Q&A discussion if a white list of existing UI labels is initially provided—what would be the input for automatic detection for 3D geometry or other types of applications, such as programming tools?

Scaling up real-time help requests: Our current work only tests real-time help between one novice and one expert. How can this approach scale to larger groups? One approach may be to combine real-time Q&A user interfaces with emerging work in real-time crowdsourcing [25,26]. Another approach may be to cache requests or save previously asked Q&A. It may be worth exploring how real-time help could be combined with asynchronous help, such as by viewing questions from other users, as has been explored in crowdsourced contextual help tools (e.g., [9,29]).

Automating real-time help: Another key research opportunity is to design better automated real-time help systems, such as chatbots. Our findings imply that in-context chatbots would be perceived as useful in learning contexts as users would be able to better describe their help needs (requiring less back-and-forth). Future work can explore how learners can be supported by automated approaches that combine synchronous help technologies [28], intelligent tutoring systems [32], and context-aware help. Furthermore, given the emergence of customer-service chatbots [22,43], future work can explore the challenges and opportunities of localizing these chatbots in context of a given application or informational website.

Supporting design feedback and collaboration: Several participants noted that MarmalAid's real-time, in-context conversation features could also be useful for soliciting feedback or working collaboratively on design problems. Although collaborative CAD systems for professionals has been explored previously [13], there is opportunity to further explore how in-context communication that allows users to localize their conversations within the interface can impact collaboration. In particular, there may be a benefit of this type of communication for virtual teams [2], where establishing a shared context can pose communication challenges. In addition, not all collaborative communication needs to occur in real-time; the in-context Q&A can also be used asynchronously and augment the history of design decisions.

Limitations: While all of our participants were novice modelers, our sample size was small, and we did not control for other individual differences, such as experience with different types of software or learning styles. Although we worked with 3D modeling experts to develop test tasks suitable for novices, it is possible that participants' behaviors could be different if they had more time or if they had worked on different 3D modeling tasks. Finally, our prototype was tested on an experimental application with only basic features; some caution should be used when generalizing results to other 3D modeling applications.

CONCLUSIONS

In this paper, we have investigated how novice 3D modelers can use remote, real-time expert help during their initial learning tasks. We have introduced MarmalAid, a web-based 3D modeling tool that allows users to visually share their context and obtain in-application real-time help from another user. The key innovation of MarmalAid is that users can have real-time conversations with experts that is anchored to the 3D geometry of their models. Our empirical findings lead to several design opportunities in HCI for using real-time, in-context help in applications beyond 3D modeling, and inform the design of automated real-time help systems.

References

[1] Celena Alcock, Nathaniel Hudson, and Parmit K. Chilana. 2016. Barriers to Using, Customizing, and Printing 3D Designs on Thingiverse. In *Proceedings of the 19th International Conference on Supporting Group Work*, 195–199.

[2] Gary Baker. 2002. The effects of synchronous collaborative technologies on decision making: A study of virtual teams. *Information Resources Management Journal* 15, 4: 79.

[3] Silvia Breu, Rahul Premraj, Jonathan Sillito, and Thomas Zimmermann. 2010. Information needs in bug reports: improving cooperation between developers and users. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, 301–310.

[4] Erin Buehler, Shaun K. Kane, and Amy Hurst. 2014. ABC and 3D: opportunities and obstacles to 3D printing in special education environments. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility*, 107–114.

[5] Andrea Bunt, Patrick Dubois, Ben Lafreniere, Michael A. Terry, and David T. Cormack. 2014. TaggedComments: promoting and integrating user comments in online application tutorials. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, 4037–4046.

[6] Yan Chen, Sang Won Lee, Yin Xie, YiWei Yang, Walter S. Lasecki, and Steve Oney. 2017. Codeon: On-Demand Software Development Assistance. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 6220–6231.

[7] Pei-Yu Chi, Sally Ahn, Amanda Ren, Mira Dontcheva, Wilmot Li, and Björn Hartmann. 2012. Mixt: automatic generation of step-by-step mixed media tutorials. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, 93–102.

[8] Parmit K. Chilana, Tovi Grossman, and George Fitzmaurice. 2011. Modern software product support processes and the usage of multimedia formats. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, 3093–3102.

[9] Parmit K. Chilana, Andrew J. Ko, and Jacob O. Wobbrock. 2012. LemonAid: Selection-based Crowdsourced Contextual Help for Web Applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '12), 1549–1558.

[10] A. Crabtree, J. O'Neill, P. Tolmie, S. Castellani, T. Colombino, and A. Grasso. 2006. The practical indispensability of articulation work to immediate and remote help-giving. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, 228.

[11] D. K Farkas. 1993. The role of balloon help. *ACM SIGDOC Asterisk Journal of Computer Documentation* 17, 2: 3–19.

[12] Jennifer Fernquist, Tovi Grossman, and George Fitzmaurice. 2011. Sketch-sketch revolution: an engaging tutorial system for guided sketching and application learning. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 373–382.

[13] Jerry YH Fuh and W. D. Li. 2005. Advances in collaborative CAD: the-state-of-the art. *Computer-Aided Design* 37, 5: 571–581.

[14] Susan R. Fussell, Robert E. Kraut, and Jane Siegel. 2000. Coordination of communication: effects of shared visual context on collaborative work. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, 21–30.

[15] Floraine Grabler, Maneesh Agrawala, Wilmot Li, Mira Dontcheva, and Takeo Igarashi. 2009. Generating photo manipulation tutorials by demonstration. *ACM Transactions on Graphics (TOG)* 28, 3: 66.

[16] Tovi Grossman and G. Fitzmaurice. 2010. Toolclips: An investigation of contextual video assistance for functionality understanding. In *Proceedings of the 28th international conference on Human factors in computing systems*, 1515–1524.

[17] Tovi Grossman, George Fitzmaurice, and Ramtin Attar. 2009. A survey of software learnability: metrics, methodologies and guidelines. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 649–658.

[18] Grossman, Tovi, Matejka, Justin, and Fitzmaurice, George. 2010. Chronicle: Capture, Exploration, and Playback of Document Workflow Histories. In *ACM Symposium on User Interface Software & Technology*.

[19] Susan M. Harrison. 1995. A comparison of still, animated, or nonillustrated on-line help with written or spoken instructions in a graphical user interface. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 82–89.

[20] Björn Hartmann, Daniel MacDougall, Joel Brandt, and Scott R. Klemmer. 2010. What Would Other Programmers Do: Suggesting Solutions to Error Messages. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10), 1019–1028.

[21] Nathaniel Hudson, Celena Alcock, and Parmit K. Chilana. 2016. Understanding Newcomers to 3D Printing: Motivations, Workflows, and Barriers of Casual Makers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.*

[22] Shep Hyken. 2017. AI And Chatbots Are Transforming The Customer Experience. *Forbes*. Retrieved from https://www.forbes.com/sites/shephyken/2017/07/15/ai-and-chatbots-are-transforming-the-customer-experience

[23] Caitlin Kelleher and R. Pausch. 2005. Stencils-based tutorials: design and evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 541–550.

[24] Benjamin Lafreniere, Tovi Grossman, and George Fitzmaurice. 2013. Community Enhanced Tutorials: Improving Tutorials with Multiple Demonstrations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13), 1779–1788.

[25] Walter S. Lasecki, Kyle I. Murray, Samuel White, Robert C. Miller, and Jeffrey P. Bigham. 2011. Real-time crowd control of existing interfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 23–32.

[26] Walter S. Lasecki, Rachel Wesley, Jeffrey Nichols, Anand Kulkarni, James F. Allen, and Jeffrey P. Bigham. 2013. Chorus: a crowd-powered conversational assistant. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, 151–162.

[27] Ghang Lee, Charles M. Eastman, Tarang Taunk, and Chun-Heng Ho. 2010. Usability principles and best practices for the user interface design of complex 3D architectural design and engineering tools. *International Journal of Human-Computer Studies* 68, 1–2: 90–104.

[28] Olivera Marjanovic. 1999. Learning and teaching in a synchronous collaborative environment. *Journal of Computer Assisted Learning* 15, 2: 129–138.

[29] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2011. IP-QAT: In-product Questions, Answers, & Tips. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (UIST '11), 175–184.

[30] Andrew Milne, Bernhard Riecke, and Alissa Antle. Exploring Maker Practice: Common Attitudes, Habits and Skills from Vancouver's Maker Community. *Studies* 19, 21: 23.

[31] S. Palmiter, J. Elkerton, and P. Baggett. 1991. Animated demonstrations vs written instructions for learning procedural tasks: a preliminary investigation. *International Journal of Man-Machine Studies* 34, 5: 687–701.

[32] Martha C. Polson and J. Jeffrey Richardson. 2013. *Foundations of intelligent tutoring systems*. Psychology Press.

[33] Suporn Pongnumkul, Mira Dontcheva, Wilmot Li, Jue Wang, Lubomir Bourdev, Shai Avidan, and Michael F. Cohen. 2011. Pause-and-play: automatically linking screencast video tutorials with applications. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 135–144.

[34] Erika Shehan Poole, Marshini Chetty, Tom Morgan, Rebecca E. Grinter, and W. Keith Edwards. 2009. Computer help at home: methods and motivations for informal technical support. In *Proceedings of the 27th international conference on Human factors in computing systems*, 739–748.

[35] Erika Shehan Poole, W. Keith Edwards, and Lawrence Jarvis. 2009. The Home Network as a Socio-Technical System: Understanding the Challenges of Remote Home Network Problem Diagnosis. *Comput. Supported Coop. Work* 18, 2–3: 277–299.

[36] Kimberly Sheridan, Erica Rosenfeld Halverson, Breanne K Litts, Lisa Brahms, Lynette Jacobs-Piebe, and Trevor Owens. Learning in the Making: A Comparative Case Study of Three Makerspaces - ProQuest. *Harvard Educational Review* 84, 4. R

[37] Rita Shewbridge, Amy Hurst, and Shaun K. Kane. 2014. Everyday Making: Identifying Future Uses for 3D Printing in the Home. In *Proceedings of the 2014 Conference on Designing Interactive Systems* (DIS '14), 815–824.

[38] Piyawadee Sukaviriya, Ellen Isaacs, and Krishna Bharat. 1992. Multimedia help: a prototype and an experiment. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 433–434.

[39] Michael Twidale and Karen Ruhleder. 2004. Where am I and who am I?: issues in collaborative technical help. In *ACM conference on Computer supported cooperative work*, 378–387.

[40] Laton Vermette, Shruti Dembla, April Y. Wang, Joanna McGrenere, and Parmit K. Chilana. 2017. Social CheatSheet: An Interactive Community-Curated Information Overlay for Web Applications. *Proc. ACM Hum.-Comput. Interact.* 1, CSCW: 1–19.

[41] Ron Wakkary, Markus Lorenz Schilling, Matthew A. Dalton, Sabrina Hauser, Audrey Desjardins, Xiao Zhang, and Henry WJ Lin. 2015. Tutorial authorship and hybrid designers: The joy (and frustration) of DIY tutorials. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 609–618.

[42] Tricia Wang and Joseph "Jofish" Kaye. 2011. Inventive Leisure Practices: Understanding Hacking Communities As Sites of Sharing and Innovation. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '11), 263–272.

[43] Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. 2017. A New Chatbot for Customer Service on Social Media. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 3506–3510.