Optimizing Text Entry with Intuitive Keyboard Shortcuts: User-Centered Insights

COOPER HICKMOTT*, YUTONG AI*, MELINA O'DELL*, CARTER GALBUS*, and ALEX MORTON*, University of Michigan, USA

In this paper, we present the evolution of "Shortcut Companion", a low fidelity prototype of an innovative keyboard shortcut text entry system. Our research commenced with a comprehensive 21-question survey, meticulously piloted, refined, and distributed to a diverse audience. This survey delved into the multifaceted realm of text entry, exploring usage patterns, ease of use, user preferences, and accessibility features. Next, we conducted contextual inquiry interviews with five participants, seeking to gain a better understanding of their unique usage contexts. These interviews provided insights into user procedures, satisfaction levels, sources of frustration, and efficiency within their distinct usage scenarios. To comprehensively analyze the findings from our contextual inquiry, we crafted consolidated affinity, sequence, and flow diagrams, unveiling four primary usage scenarios: opening and closing applications, navigation, text formatting, and text transfer. We developed user requirements, which underscored the user's strong preference for simpler and more intuitive shortcuts. Leveraging these insights, the "Shortcut Companion" design was crafted, focusing on simplifying shortcuts, integrating an AI Shortcut Recommendation Assistant, and ensuring cross-platform compatibility. Following the development of a low fidelity prototype, we conducted heuristic evaluation and simplified user testing. Feedback from these evaluations highlighted critical usability issues such as the absence of documentation and challenges in clarity and intuitiveness of the user interface. Participants encountered difficulties in accessing the application, understanding functionalities, and recognizing shortcuts without external guidance. Issues ranged from loading times to clarity in user interface elements, emphasizing the need for improved accessibility and design cues. The findings from heuristic evaluation and simplified user testing underscore the importance of refining design cues, enhancing clarity, and aligning functionalities with user expectations. Insights garnered from these evaluations inform iterative improvements, focusing on reducing ambiguity, enhancing accessibility, and refining visual cues to ensure a more intuitive user experience. In the final phase of our user-centered design process, we advanced to a high-fidelity prototype and executed quantitative user assessments. Utilizing baseline and follow-up Likert scale questionnaires, we gauged how well our prototype aligned with user requirements. Statistical analyses, including Wilcoxon Signed-rank tests, revealed heightened user confidence in translating shortcuts across systems, rectifying errors, and recalling forgotten shortcuts. The "Shortcut Companion" prototype serves as a stepping stone toward an enhanced text entry system, aiming to maximize user efficiency and satisfaction when using keyboard shortcuts.

CCS Concepts: • **Human-centered computing → Interaction techniques**.

1 Introduction

In an ever-evolving technological landscape, text entry has transitioned from traditional methods to modern alternatives, partially thanks to the proliferation of accessibility features. Despite the convenience of these modern text entry methods, the challenge lies in making them intuitive for users, especially those less tech-savvy. Our research began by recognizing the challenges discussed in the "Five Challenges for Intelligent Text Entry Methods" [1], and embarking on a mission to understand the diverse world of text entry methods. We focused on various stakeholder groups, each with unique requirements, and aimed to improve user performance in text editors. This research aims to gain a comprehensive understanding of the text entry practices employed by these stakeholders, which may include multilingual individuals and other distinct user demographics, and to use this knowledge to optimize user performance in text editor programs.

One of the key challenges we faced was a lack of comprehensive knowledge regarding the specific focus area we are addressing. To address this obstacle, we design and conduct a comprehensive survey to investigate user performance. This survey will also serve as a foundational step towards gaining a deeper understanding of the context in which our project operates. In the study, we designed a 21-question survey consisting of questions about usage patterns, ease of use, preferences, and accessibility features and distributed to the potential participants via Google form. No responses were collected from any participants under the age of 18. In the end, we collected 25 responses which may have limitations regarding generalizability. The results of our survey yielded several noteworthy findings and insights. Firstly, ease of use is one of the key factors that determine the user's favorite text entry method. Laptop keyboards and smartphone touchscreens were commonly used methods for text entry, but external keyboards and laptop keyboards emerged as the favored option. The voice entry and handwritten translation display various user responses. Another significant observation was the widespread preference for auto correct capabilities. Although some users report frustrations and encounter issues, the majority of the users considered it important, suggesting that while there is room for improvement, auto correct remains a widely appreciated feature. Lastly, we noted that visual modifications, such as text and color adjustments, appeared to enhance the overall user experience.

The feedback collected from the survey also reflected the importance of keyboard shortcuts when performing text entry. It helped us refine our focus to optimizing user performance with intuitive keyboard shortcuts in a text editor program. After the survey, to further understand the context of use for text entry methods and keyboard shortcuts specifically, we conducted several interviews using contextual inquiry. Context inquiry allows us to step into user scenarios and gain knowledge of users' subjective view of text entry shortcuts, helping us to better understand the user's procedures, satisfaction, frustration, and efficiency in the context of use. During the interview, we asked the participants to describe the last time they used a keyboard shortcut and requested that they walk us through the process. The interviewers took notes and create interpretations during the inquiry and afterwards analyzed the notes as a whole to build sequence and flow diagrams. As a result, we identify four different primary usage scenarios: opening and closing applications, navigation, text formatting, and text transfer. Users tend to perform simpler and more intuitive shortcuts, such as those matching the first letter of the intended functionality, whereas complex and lengthy shortcuts are less easily memorized, recalled, and sometimes frustrating for users, highlighting the importance of shortcut design for user performance and usability. The results points to a future of optimizing user performance through encouraging the use of shorter, simpler, and more intuitive shortcuts.

The culmination of our contextual understanding endeavors led us to the development of the "Shortcut Companion." This innovative design attempts to unify key features and design elements from the five individual solutions into one cohesive system, addressing the user requirements we developed. The "Shortcut Companion" is a multifaceted tool designed to enhance text entry and application interaction through keyboard shortcuts. It introduces an AI Shortcut Recommendation Assistant, context-aware pop-ups, direct user feedback, and cross-platform compatibility, providing a seamless and efficient keyboard shortcut experience for users across various applications and platforms.

After the development of a low fidelity prototype for "Shortcut Companion", we evaluated our system using heuristic evaluation and simplified user testing. This was done to gather expert opinions and critiques of our design of a keyboard shortcut assistant tool, and to test against a subset of our defined user requirements and collect the target audience's feedback and opinions on the usability of our design of a keyboard shortcut assistant tool. The heuristic evaluation pinpointed a pressing need for user documentation and guidance, prompting plans for an "on-boarding period" to enhance user familiarity. Simultaneously, user testing highlighted challenges in discoverability, especially within the translation mode, alongside frustrations stemming from the absence of a search feature for regenerated shortcut suggestions and discrepancies in user understanding of the "Undo" button's functionality.

The culmination of our user-centered design process was crafting a high-fidelity prototype for the 'Shortcut Companion' application and subjecting it to quantitative user evaluation. We developed baseline and post-task followup Likert scale questionnaires to gauge the prototype's alignment with user requirements, conducting null hypothesis testing using the Wilcoxon Signed-rank test. The evaluation revealed statistically significant enhancements in user confidence, particularly in translating shortcuts across operating systems, rectifying erroneous shortcuts, and rediscovering forgotten ones.

2 Related Work

Writing systems are cultural artifacts deeply entwined with human civilization, evolving through various media such as cave walls, rune stones, parchment, and now networked computers. The transformation of writing systems over millennia underscores their pivotal role in human history, reflecting societal needs and technological advancements. In recent years, the text entry landscape has witnessed a metamorphosis, especially with the advancement of handheld devices and emerging display technologies (VR/AR). The traditional desktop keyboard, though efficient, does not cater to all user needs, particularly individuals with physical limitations or those seeking a more intuitive alternative [4] [5].

The biggest challenges when designing text entry methods: localization, error correction, editor support, feedback, and context of use [1]. Localization, influenced by keyboard layouts and language-specific models, poses a significant hurdle in achieving efficient text entry across diverse linguistic contexts. Desktop keyboards and mobile text entry methods necessitate tailored support for each language, demanding significant efforts in lexicon collection and language adaptation. Error correction and editor support mechanisms are essential in text entry for mitigating both cognitive and motor errors. Cognitive errors arise from users' improper modeling of intended words, while motor errors stem from physical challenges during text input. Error correction in text entry depends on effective editor support. While immediate feedback impacts user perception and performance, effectively utilizing confidence scores to annotate words/sentences requires further accuracy and integration of external knowledge. Balancing user trust in automated systems, particularly in handling spelling errors, presents an ongoing research area. Designing text entry methods that adapt to various user contexts is a critical challenge. Understanding user preferences and context-aware switching of input methods are essential aspects of creating a seamless and flexible text entry experience. This paper serves as a road map for advancing our research on text entry methods. Building upon these challenges, we aim to explore innovative strategies and methodologies to enhance the text entry experience in the context of efficiency, user experience, and adaptability.

When evaluating text entry methods, the "important" factors to consider are text creation versus text copy tasks, novice versus expert performance, quantitative versus qualitative measures of performance, and the speed-accuracy trade-off [3]. Advantages of text copy tasks include mitigating the presence of behaviors not required of the interaction ("What do I type next?"), easier identification of errors (what's supposed to be typed is known), and having control over the distribution of letters pressed. The main advantage of creation tasks is they mimic typical usage as well as requiring a lower degree of focus of attention (FOA). A well designed experiment can result in reaping the advantages from both these techniques however. First, present participants with short, easy-to-memorize phrases of text. Participants are directed to read and memorize each phrase before entering it. Both text copy and text creation task advantages are captured this way (reduced FOA, control over letter frequencies, easy error identification, and not forcing the user to think about what to write). Text input method evaluations often prioritize expert entry rates, however the success of these methods relies heavily on the experience of novices as well. Immediate usability is crucial, as novices might be deterred if they have to invest significant effort to achieve expert speed initially. We consider both novice and experienced users when designing our method evaluation criteria. To accurately evaluate a new text input technique, a controlled experiment using both quantitative metrics and established qualitative test instruments is essential. While quantitative tests are fundamental, qualitative aspects are equally vital, particularly in human-computer interfaces. Users' comfort and perceived payoff in task accomplishment matter, and their opinions should be systematically gathered and considered. To assess a new text input method, speed and accuracy are key metrics. Speed can be measured by characters per second (cps) or words per minute (wpm), considering five characters as a word. Accuracy is more complex, often evaluated by error rates and types like substitution or omission. Automation is challenging due to the compounding nature of errors. Achieving a balance between speed and accuracy is crucial, as participants may trade one for the other. Effective evaluation should consider both aspects for a comprehensive understanding of the text input method's performance.

Users are often frustrated when their thoughts occur faster than they are able to input the text [2]. One common resolution to this is voice entry, but the authors note that due to technical limitations of speech recognition users are often left dissatisfied. Additionally, humans' short term memory is fickle and we function much better when reading printed text. On the other hand, typing in the English language typically results in a 40-50% reduction in speed compared to speech. The authors found that handwriting, especially when the ability to write shorthands and abbreviations, was faster than typing but more reliable that speech text entry. Handwritten text also allows the user to draw diagrams or annotations. Handwritten text of course comes with the problem that every user has a different style of handwriting, so discerning what letters and words a user is attempting to input presents a challenge. One specific shorthand that was discussed was the Pitman Shorthand, which is a way to encode one or many phonetic sounds into pen strokes. Advanced writers with the Pitman Shorthand can convey approximately 100 words per minute, whereas handwritten text typically falls around 35 words per minute. The authors used this shorthand to create guidelines for a viable "machinography," or a script for machine communication. Some of these guidelines include a viable structure and grammar, a match between the script and the language's alphabet, and that each symbol's difficulty of writing corresponds to its frequency in the language.

The papers discussed above have paved the way for our current research by shedding light on the challenges and nuances of text entry methods. Notably, the papers delved into the complexities of text entry, emphasizing issues such as localization, error correction, and user context [1] [3]. These works have served as a road map for advancing our research and encouraged us to explore innovative strategies to enhance text entry experiences. However, the survey we designed and our contextual inquiry investigation extends the scope by directly investigating users' preferences and experiences with various text entry methods. While these papers provide valuable insights into the challenges, they do not comprehensively cover the user perspective or the specific needs of distinct user demographics. Our survey and contextual inquiry aims to bridge this gap by gathering data on usage patterns, ease of use, preferences, and accessibility features, allowing us to create a more user-centric framework for text entry method design and optimization.

3 Establishing Focus: Initial Survey

Broadly speaking, the purpose of the survey is to understand how various stakeholder groups currently enter text when interacting with existing user interfaces. The survey includes questions meant to quantitatively determine the most common and least common methods of text entry, the methods of text entry that exhibit the highest and lowest ease of use, and the most and least valuable text entry and accessibility features. All of these questions are asked for the purpose of identifying areas of text entry interaction that have opportunity for improvement, and for the purpose of narrowing down the focus of the research to a specific aspect of text entry.

3.1 Method

3.1.1 Pilot: A pilot survey (see appendix B.1) was first performed to identify any flaws in the initial survey design with the help of user feedback. 5 participants were given the pilot survey while the survey designers took notes.

The pilot participants were timed with the purpose of determining whether or not the survey had the proper number of questions. All of the surveyors found that the time to take the survey fell within 5-10 minutes, the ideal range, and it was decided that no major change to the length of the survey was necessary.

Several trends were noticed when the surveyors compiled their notes for the pilot. First, participants often did not understand the terminology that was used. To remedy this, terminology that was found to be confusing was given a definition, more context, or additional details in order to better convey the intended meaning. Second, additional options for multiple choice questions were given to several questions in order to avoid forcing an answer. For example, several of the pilot participants did not know their words per minute (WPM). While the pilot question asking about WPM was careful to not include overlapping ranges, it fell into the common pitfall of forcing an answer, even for those who did not know their WPM. To remedy this, the final survey included the option "I don't know." Finally, there were several errors that resulted from transferring the survey to a Google form. For example, a question that asked users to "select all that apply" was set up as a multiple choice question only allowing a single choice in the Google form. All of the errors of this kind were identified with the help of the pilot users and fixed for the final survey.

3.1.2 Final survey: The final survey (see appendix B.2) is divided into 4 sections: background, usage patterns, ease of use, and preferences and accessibility.

The purpose of the first section, background, is to collect demographic information about survey participants. The age, education level, and work industry could all affect the responses to the subsequent sections, so this information is valuable for analyzing results.

The purpose of the second section, usage patterns, is to collect data about the the 6 text entry methods chosen for the survey: laptop keyboard, external keyboard, smartphone text entry, VR/AR keyboard, voice entry, and handwritten text transferred to virtual (including smart watches). First, participants are asked to respond which of the text entry methods they have used before. Next, they are asked to pick the one they use the most, and then which one is their favorite. They are then asked why the chosen text entry method is their favorite, and are given the options speed, accuracy, ease of use, or other (the option to fill in their own reasoning). Participants are then asked how often they interact with text entry (with ranges from never to constantly), what language they usually type in, a self-rating of their own typing skill (on a Likert scale from "very unskilled" to "very skilled"), and finally their estimated WPM. Like the information collected in the background section, the results from these last 4 questions could be valuable for further stratifying the data.

The purpose of the third section, ease of use, is to determine each participant's opinion of how easy it is to use each of the 6 chosen text entry methods. Responses are collected using a Likert scale with options from "very difficult" to "very easy."

The final section, preferences and usability, lists common text entry features and accessibility features and asks participants which they feel are important to them.

3.2 Tasks and Procedures

Informed consent was gathered when distributing the survey to potential participants. Participants were told that their submission of the Google form would serve as consent to participate in a single semester duration research study, the purpose of which is to improve text entry interactions within the realm of Human-Computer Interaction. Participants were told that their responses would be anonymized, and they were also informed that should they at any time wish to be removed from the research study, they need only contact one of the surveyors to have their data removed.

Responses to the survey were collected over a 24 hour period. No responses were collected from any participants under the age of 18. Participants who answered "Younger than 18" to the first question would have their survey automatically discarded.

3.3 Participants

We received 28 voluntary responses to the survey we sent out (see appendix B.2).

- 87.5% of respondents were between the ages 18-24, 8.3% were between ages 25-34, and 4.2% were between ages 55-64.
- 53.6% of respondents classified themselves as students, 32.1% of respondents as Employed (full time), 28.6% as Employed (part-time), 7.1% as unemployed and 3.6% as retired.
- 17.9% of respondents classified themselves as a "Very Skilled" typer, 60.7% as "Skilled", and 21.4% as "Neither Skilled nor Unskilled".
- 64.3% of respondents classified themselves as interacting with text entry methods "multiple times per hour" while 35.7% answered "constantly".
- 96.4% of respondents often type in English, while 10.7% of respondents often type in Chinese (some type in both English and Chinese.)

Each member of our research group sent the survey to 10 people, aiming for a response from around half of them which, resulting in a total of 25 complete responses. While 25 responses should give us a solid foundational understanding of the current context of use, it's important to note that a sample size of 25 may have limitations in terms of generalizability and statistical precision.

3.4 Results

3.4.1 Usage Patterns: We asked which text entry methods the respondents had used previously. Most respondents had used a laptop keyboard (100%), external keyboard (96.4%) and smartphone touch screen (96.4%) for text entry. We followed this question up by asking which method is the "Favorite to use." The majority of respondents picked external keyboard (53.6%), while 25% chose laptop keyboard and 21.4% chose a smartphone touch screen. When asked why they chose their previous answer as their favorite, 50% of respondents claimed “ease of use,” defined as not feeling strained while using the entry method. 39.3% claimed “speed” for their reason, defined as both in your entry and responsiveness of the device. Other responses included accuracy, having the most experience with that device, or enjoying the aesthetics.

3.4.2 Ease of Use: Respondents were asked about the ease of use – definition up to the interpretation of the individual – for each of the text entry methods seen in Fig. 1 below.

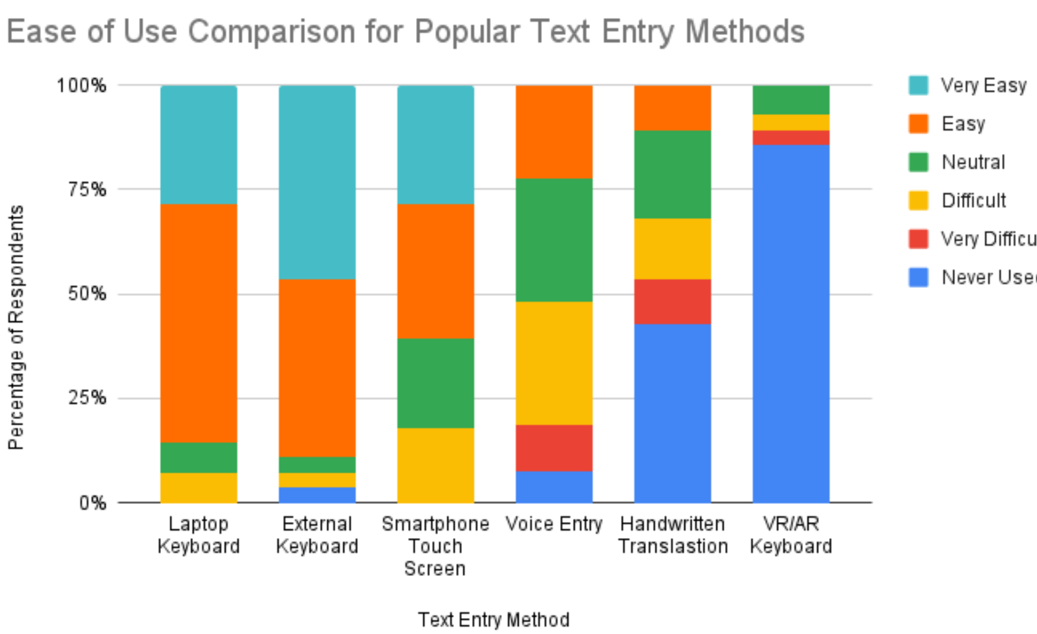

Fig. 1. Laptop and external keyboard are considered easy to use amongst the general population, while voice entry and handwritten translation methods are split relatively evenly amongst all answer choices.

- Over 85% of respondents classified laptop and external keyboards as either “Very Easy” or “Easy” to use
- The majority of respondents classified smartphone touch screens as either “Very Easy” or “Easy” to use (60.7%), however the other 39.3% classified this screen as being either “Neutral” or “Difficult” to use
- Voice entry text translation methods were split pretty evenly among the respondents, with 0% claiming “Very Easy, 22.2% claiming “Easy”, 29.6% claiming “Neutral”, 29.6% claiming “Difficult”, and 11.1% claiming “Very Difficult.” The remaining 7.4% of respondents had never used this type of screen.
- Neither handwritten translation nor VR/AR keyboard text entry methods were popular amongst the respondents. 10.7% of respondents classified handwritten yranslations as “Easy”, 46.4% of respondents classified it as either “Neutral”, “Difficult” or “Very Difficult”, and the other 42.9% claim they have never tried this method before. Similarly, 14.3% of respondents

classified AR/VR keyboards as either "Neutral", "Difficult" or "Very Difficult", and the other 85.7% claim they have never tried this method before.

3.4.3 Preferences and Accessibility: We first asked respondents what text entry features are important to them. We provided some options, but also allowed them to respond with whatever they thought of. 74.1% of respondents report an autocorrect feature is important. Additionally, 44.4% of respondents claim predictive text is important; and 40.7% of respondents say keyboard sounds (or absence of) is important to them. For our next question, we asked the respondents what accessibility features are important to them. 60.7% of respondents claimed this question was not applicable to them, so we do not consider these. Of the remaining surveys, 54.5% of respondents state that both large text size and high contrast is important to them. Additionally, the ability for customizable programmable keys is important to 36.4% of respondents. Finally, we gave room for respondents to report any gripes or annoyances related to their experiences with text entry. Most individuals did not report any such gripe, however 2 individuals reported frustration with autocorrecting tools.

3.4.4 Themes: From our results, we gathered a few key takeaways. First, we noticed that while laptop keyboards and smartphone touchscreens were among the highest used methods, they were both dominated by external keyboard as being the favorite method. Combining this with the 50% of responses that "ease of use" was the primary factor in choosing their favorite method, we hypothesized that laptop keyboards and smartphone touchscreens are more difficult to use than an external keyboard. This was supported by the 39.3% of users reporting that the smartphone touch screen was not easy to use. Another theme that we discussed was the widespread preference of autocorrect capabilities. While two users reported issues with autocorrect failing at times, the 74.1% of users that selected autocorrect as being important leads us to hypothesize that while autocorrect can be improved upon, it is still a widely utilized feature. Finally, while our sample size for accessibility features was about half of our total respondents, we noticed that visual modifications of text and color tend to improve user experience.

4 Understanding Context of Use: Contextual Inquiry

Contextual Inquiry is a principled method of gaining qualitative information in the field. Its structure makes use of artifacts and reenactments to jog the memory of participants, all in order to remind them of the actions that they took, the reasons why they took them, and the feelings they had at the time.

The interviewer's role is to guide the conversation toward the focus decided beforehand, yet must remain flexible enough to allow the focus to shift or broaden as the participant relates their experiences. If all goes well, the participant will identify the tasks performed and the breakdowns that occurred in a natural way, giving the interviewer deeper insight into the context of use.

4.1 Method

To investigate, each member of our group conducted an interview. The reason we use contextual inquiry as our research method is that by stepping into the user scenario, we can better understand the user's procedures, satisfaction, frustration, and efficiency in the context of use. Since the research focus of our study is to optimize the text entry method, we need to first investigate the client's subjective view of the current text entry method. Therefore, understanding the user's frustrations, efficiency, etc. via contextual inquiry is a suitable method for us to investigate potential improvements.

4.2 Tasks and Procedures

The interviewers first contacted the participants and obtained each participant's consent to conduct an interview. The interviewers later scheduled a 1-hour long interview with their participant. Before the interview started, the interviewers first asked for the participant's oral consent again to continue the interview. After the agreement, the interviewers introduced themselves and told the participant that they could always stop the interview if they felt uncomfortable answering any question. During the interview, the interviewers first asked the participants about the last time they used a keyboard shortcut and requested them to walk through the process, taking notes whenever possible. After the participants had recalled the last time they used a shortcut, the interviewers started to ask some follow-up questions. The interviewers took notes during and after the inquiry in the form of post-it notes with interpretations written on them (C.1). Interviewers focused on the accomplishments of the participants, any connections the participant had with others, the participant's sense of self in relation to the focus, and finally any strong emotions felt by the participant. The interviewers planned to perform as many interviews as necessary to ensure the quality of the information gathered, but each interview decided after the first that it was sufficient.

Each interviewer then analyzed their interpretations and modeled the sequence of events in one or more sequence diagrams (C.2). A sequence diagram shows the sequence of steps that participants took to perform their tasks, the participants' intentions behind the different steps they took, and the breakdowns that prevented or made it difficult for participants to complete their tasks. The interviewers then modeled the flow of artifacts and information in a flow diagram (C.3). The flow diagram shows the different stakeholders that were mentioned in the interview and shows the different technologies that the stakeholders used to complete their tasks. Once all of the interviews were completed, all of the interviewers met together to consolidate the results into an affinity diagram (C.6). First, all of the interpretations were laid out together on one Miro board. The interviewers grouped together interpretations based on general themes and then broke them down further into specific categories. Finally, the interviewers reviewed their sequence and flow diagrams and consolidated those together as well (Fig. 2) (Fig. 3).

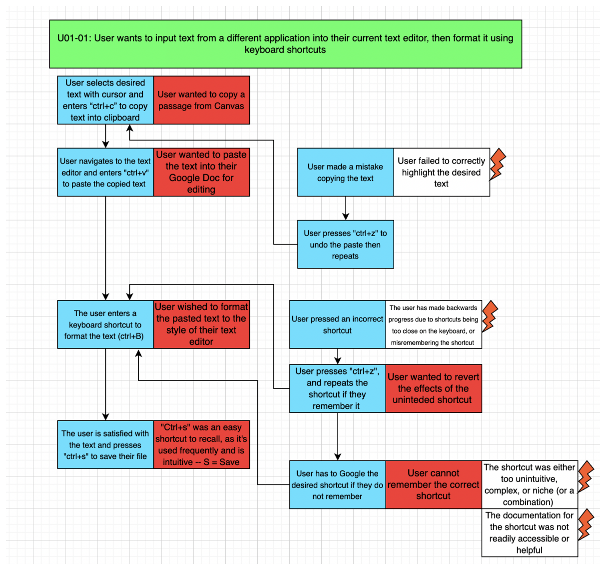

Fig. 2. Consolidated sequence diagram for a user transferring text and formatting it in a text editor

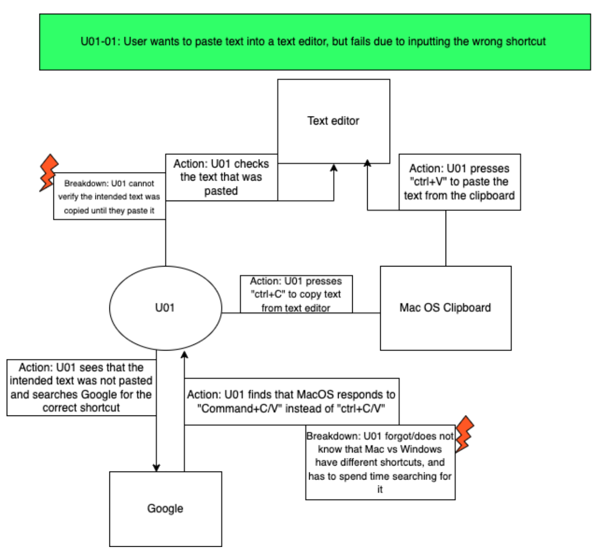

Fig. 3. Consolidated flow diagram for a user failing to paste text using a keyboard shortcut

4.3 Participants

We have interviewed five different participants. The criteria for participation was that they had taken the previous survey. To ensure each participant's privacy, we name them User 01-05 and use these labels throughout the paper. Only one interview took place virtually via Zoom (user 02), while the rest happened in person, in front of a computer with keyboard. No incentives were promised to participants.

User 01 is a 21-year-old female college student who is majoring in Computer Science. They use laptop keyboard shortcuts regularly and define themselves as proficient with technology.
User 02 is a 23-year-old male who works in the tech industry. They identified themselves as very proficient with technology and use keyboard shortcuts regularly.
User 03 is a 50-year-old female who works in the education industry. They classified themselves as not proficient with technology and stated that they sometimes use keyboard shortcuts.
User 04 is a 23-year-old male who is currently a graduate student. They use keyboard shortcuts on a daily basis and define themselves as proficient with technology.
User 05 is a 21-year-old male college student who is somewhat tech savvy and occasionally uses basic keyboard shortcuts.

4.4 Results

The affinity diagram (Link to Miro Board, C.6) elucidated much about the users' context of use, and allowed us to draw some interesting conclusions regarding what makes a keyboard shortcut intuitive or difficult to use.

Regarding the context of use, we found that users primarily used keyboard shortcuts in 4 situations, with corresponding themes on the Miro board: opening and closing applications, navigation around their computer, formatting of text, and transfer of text. Both users 03 and 04 reported using the keyboard shortcut ctrl+alt+delete to close an application or close out of the desktop itself (U03-18, U04-16). In regards to navigation, user 02 told of their use of cmd+f to find text (U02-09), and user 04 recounted their use of cmd+tab to switch tabs (U04-02). On the other hand, users 01, 02, and 03 all had instances where they described using the mouse over keyboard shortcuts to navigate (U01-07, U02-03, U03-03). 2 users described using keyboard shortcuts to format text, using ctrl+b to bold text (U05-07), or ctrl+a to select all text in order to change the font (U03-16), while 1 more user described using a shortcut to open up the Apple emoji menu (U02-07). 3 users also described using the shortcut cmd+z or ctrl+z to undo. Finally, all 5 users mentioned the use of some combination of copying (ctrl+c or cmd+c), cutting (ctrl+x or cmd+x), and pasting (ctrl+v or cmd+v) to transfer text from one place to another (U01-03, U02-06, U03-10, U04-13, U05-03).

Of the 4 themes described, it is clear that transfer of text is the situation in which our users most frequently use keyboard shortcuts, as all 5 mentioned it during their interviews. Formatting and reverting of text comes in second place, with 4 users describing the use of keyboard shortcuts to accomplish those tasks. Opening and closing of applications were described by 2 users, and navigation was described by 2 users as well. It's important to note, however, that 3 users explicitly called out using the mouse over using keyboard shortcuts when navigating the screen, making navigation possibly an area of improvement when it comes to using keyboard shortcuts.

Regarding the usability of keyboard shortcuts, most users state that they tend to recall those shortcuts that are more intuitive. According to the affinity diagram (C.6), users 01, 04, and 05 can better retrieve the shortcut for save, copy-and-paste, and so on because these shortcuts' keys match the first letter of their intended functionality or these shortcuts are shorter and therefore easier to remember(U01-05, U04-11, U05-09, U05-14). On the other hand, users tend to not memorize those shortcuts that are complex and longer. Users 01, 04, and 05 reflected that complicated shortcuts such as printing part of the screen that are combinations of several keys cause frustration and sometimes need to look up the correct combination using the internet as a resource (U01-18, U04-18, U04-20, U05-10, U05-11). From users' feedback, we can clearly see that users tend to remember and perform simple, intuitive keyboard shortcuts. The length of the shortcuts as well as the intuitiveness of the shortcuts are two main factors impacting the user performance.

When consolidating the sequence diagrams, we decided to take the action described by all users – copy and pasting or transferring text – and add all of the possible breakdowns that each individual user ran into to the sequence (Fig. 2). These breakdowns, which include failure to copy the correct text, using an incorrect keyboard shortcut, and consulting the internet, are all potential areas where the usability of keyboard shortcuts can be improved.

The consolidation of the flow diagram narrowed the task once again to only encompass copy and pasting or transferring of text (Fig. 3). There was debate about whether or not to include multiple stakeholders, but it was decided that since most of the users described only their interactions with computers directly, only a single user would be present on the final diagram, with artifacts surrounding them. These artifacts, which include Google, the computer's clipboard, and the text editor application, all have links modelling the flow between them and the user. Additionally, the clipboard and the text editor have a link to model the transfer of data from one to the other. The breakdowns modeled reflect the same issues present in the sequence diagram.

5 User Requirements and Functional Constraints

In this section, we will discuss the user requirements for optimizing keyboard shortcut usage with a text entry device. All cited interpretations correspond to the interpretations in the yellow notes of the affinity diagram [C.6].

1. Users must be able to discover new shortcuts without consulting external resource (like Google): U04-19, U01-06, U02-16, U01-12, U02-13, U02-15, U02-14.
In the cited interpretations and first category of the affinity diagram [C.6], it is clear to see users often forgot or simply are unaware of common keyboard shortcuts, and resort to googling the functionality they are looking for. This is currently the fastest way to discover keyboard shortcuts, as combing through OS system settings either does not produce an answer or searching takes too long. Users should be able to discover shortcuts to improve their efficiency quickly, without having to disrupt the flow of their work or having to go too far out of their way to search for them.

2. Users must be able to complete all 'select' actions a mouse can complete via their keyboard: U03-04, U03-05, U03-08, U04-01, U04-02.
As seen in the cited interpretations, some users claim using their keyboard to 'select' what is currently highlighted on their screen is easier and faster than looking for icons on the home screen and selecting them with their mouse. Thus, it is necessary that all screen navigation 'select' actions that can be completed by a mouse should be able to be completed only using they keyboard.

3. Users must be able to undo previous text entry actions and resume their intended task with a single keyboard shortcut completed without lifting their hands from the

keyboard: U03-14, U03-15, U04-15.
A major theme discovered through the affinity diagram [C.6] is the user of keyboard shortcuts to edit, revert, and format text in many contexts. As seen in the cited interpretations, it is common for users to make a mistake while conducting a text entry task (such as misspelling a word). Thus, it is necessary to have an "undo" mechanism that can be used quick and efficiently without delaying or significantly distracting the user from their current task.

4. Users must be able to style text using keyboard shortcuts, allowing for formatting options such as bold, italic, and underline: U05-07, U03-16.
As seen from the cited interpretations and continuing the theme of text editing with shortcuts, a common use case for keyboard shortcuts while writing a document is editing the formatting of text (such as bolding, italicizing, and underlining). By incorporating keyboard shortcuts for formatting, users can apply styles swiftly, saving time and enhancing productivity, which is essential for a seamless and productive document creation or text entry process.

5. Users familiar with a given operating system's keyboard shortcuts must be able to recognize these shortcuts on other operating systems: U04-06, U04-10.
In the cited interpretations, users voiced frustrations with transferring knowledge of keyboard shortcuts across operating systems. Keyboard shortcuts between MacOS and Windows keyboards are different enough that users who are switching platforms will likely have to look up the shortcuts to familiarize themselves. This happens frequently enough to be a significant inconvenience for users (discovered by grouping multiple occurrences of cross-platform shortcut confusion into an affinity diagram category [C.6]) and may discourage the use of keyboard shortcuts. To make the switch between platforms less confusing, it is requirement that users can transfer their knowledge from one operating system to another.

6. Users must be able to copy, cut and paste text each using a single keyboard shortcut without lifting hands from the keyboard (both within an app and across different apps): U01-03, U02-06, U02-11, U02-04, U05-03, U03-10, U03-11.
The actions of copying, cutting, and pasting text were by far the most used and most highly reviewed keyboard shortcuts, as shown in all of the cited interpretations and the affinity diagram [C.6] theme related to text transfer with keyboard shortcuts. Overall, these text editing actions are critical for text entry tasks in all contexts. Therefore, it is important that users can perform this task most efficiently with only keyboard shortcuts, without having to use their mouse or click any menus, which would disrupt their workflow.

7. Users must be able to execute their intended shortcut and resume intended task after a slip (i.e. using the wrong shortcut): U05-22, U05-23, U04-07, U04-08.
Users utilizing keyboard shortcuts sometimes make slips while performing a text entry task, where they click the wrong combination of keys. As seen in the cited interpretations, this may be a simple misclick by the user, or more interestingly, an attempt to guess a keyboard shortcut that they knew in the past. If undoing these mistakes is not easy, users will be discouraged from trying to use shortcuts at all, limiting their overall productivity. This situation is included in the theme related to factors that determine if users choose to use keyboard shortcuts in the affinity diagram [C.6]. The user should be able to recover quickly from an incorrect keyboard shortcut, continuing their task by using the correct one.
Note that this is different from using a keyboard shortcut to undo a typing mistake (see requirement 3), and instead focusing on the use of an incorrect keyboard shortcut to perform some text entry or

navigation action.

8. In the event of a user failing to recall a keyboard shortcut, users must be able to recognize potential shortcuts without consulting an external resource: U05-10, U05-11, U04-20, U02-15.
As seen in the cited interpretations, it is common for users to forget a keyboard shortcut that they had previously learned. This is a different pitfall than not knowing a keyboard shortcut in the first place, which requires it to be discovered from scratch. Failing to remember a keyboard shortcut makes users more inefficient and annoyed when typing. Similar to the first user requirement, users should be able to rediscover keyboard shortcuts that they have forgotten without derailing them from the task at hand.

6 Initial Design and Low Fidelity Prototypes

Based on their understanding of the context of use, the 5 authors created designs that attempt to fulfill the user requirements defined previously.

6.1 Individual Sketches and Storyboards Design Critique

In order to consolidate their ideas into a single design, the authors met to critique the flaws in each design as well as identify the best ideas to carry forward to the final design.

6.1.1 Design 1 The first design provides a comprehensive approach to enhancing text entry through a software application called "Shorty," which assists users in discovering and using keyboard shortcuts effectively. It offers features like a Shortcut Recommendation Engine, Cross-Platform Shortcut Translator, Contextual Shortcut Pop-ups, and a Shortcut Dictionary. The corresponding sketch (Fig. 53) shows Lisa (see her persona in Fig. 48) getting intelligent keyboard shortcut recommendations, using a shortcut dictionary, and translating shortcuts between operating systems with an application. The storyboard (Fig. 58) shows Lisa getting custom suggestions for keyboard shortcuts based on common actions she performs.

The main critique for this design was that the frequent popups could become overwhelming for users. Either the frequency or the circumstances that the popups appear would need to be changed for the final design. Another critique was that translating some keyboard shortcuts between operating systems might not be possible because the keyboards themselves are different.

6.1.2 Design 2 The second design focuses on providing direct user feedback and the ability to undo incorrect shortcuts, as well as displaying a list of relevant shortcuts to help users find the correct one. The corresponding sketch (Fig. 54) shows Anny (see her persona in Fig. 49) undoing an incorrect shortcut and receiving suggestions for the correct shortcut. The storyboard (Fig. 59) shows Anny guessing a keyboard shortcut incorrectly and having a tool suggest recommendations based on the mistake.

Similar to the first design's critique, it was decided that popups displaying shortcuts every time a popup was performed would be too often.

6.1.3 Design 3 The third design introduces the concept of a context-aware popup that appears when the user holds down a designated key for 5 seconds. The popup displays common keyboard shortcuts based on the active application. It aims to help users remember shortcuts and reinforces them within their memory. The corresponding sketch (Fig. 55) shows Damian (see his persona in Fig. 50) looking at a popup with suggested keyboard shortcuts for the application he is using (a code editor). The storyboard (Fig. 60) shows Damian forgetting the rest of a keyboard shortcut and having the popup suggest recommendations based on the first key in the combination.

During the design critique it was suggested that instead of holding down a key for 5 seconds, the user should be able to configure how long the key should be pressed down. It was reasoned that a user looking for high efficiency would not have the patience to wait 5 seconds for a popup to appear.

6.1.4 Design 4 The fourth design suggests an AI assistant that monitors user text entry, detects repetitive actions, and provides relevant keyboard shortcut suggestions. It also offers auto-correction for incorrectly entered shortcuts. The corresponding sketch (Fig. 56) shows Maya (see her persona in Fig. 51) getting customized keyboard shortcut recommendations (based on patterns and/or mistakes) in a popup, in addition to a shortcut search functionality. The storyboard (Fig. 61) shows Maya getting keyboard shortcut suggestions from an AI tool in a popup while she types.

The design critique noted that an AI assistant with a face and personality was tried by Microsoft before (Clippy), and it received an overwhelmingly negative response by users. Clippy was helpful to new users, but annoying to anyone with even a small amount of experience. The final design should be careful to avoid the mistakes that Microsoft made.

6.1.5 Design 5 The fifth design proposes a browser extension that interacts with Google Docs, providing suggestions of which keyboard shortcuts to use based on the user's current context. The corresponding sketch (Fig. 57) shows Sam (see their persona in Fig. 52) getting suggested keyboard shortcuts to edit text. The storyboard (Fig. 62) shows Sam using a keyboard shortcut suggester to format text more efficiently.

During the design critique it was decided that focusing on a single application was too narrow, and that the final design should be usable in as many contexts as possible.

6.2 Personas

The final persona is a combination of the 5 individual personas we created in the design stage. Each of us agreed that a typical user of our design will be someone relatively tech-savvy and someone who values their efficiency and performance while working with their computer, but hasn't quite figured out the best way to learn and utilize keyboard shortcuts to get there yet.

Our final persona [4] describes Sarah, a project manager for a software company who is enthusiastic about tech. She values efficiency and precision in her work because her job requires her to be on top of things all the time.

Sarah is a quick learner and has some experience with keyboard shortcuts, but only the very common ones (like Ctrl+C and Ctrl+V). She knows that there are more keyboard shortcuts out there that could help her be more productive, but she struggles to find the time to learn them. She wishes that there was a quicker and easier way to discover new shortcuts without disrupting her normal workflow.

Sarah's goals revolve around top performance in the workplace. It is her personal goal to keep the company's technical specifications and documentation well-organized, and to do that as efficiently as possible. She also values precision, as mistakes in her work can cost the company lots of time and money. She presently uses some keyboard shortcuts in her endeavors to keep up with these goals, but she knows she hasn't reached her full potential yet.

Sarah struggles to learn and remember keyboard shortcuts on the job. She frequently switches between MacOS and Windows machines, where the keyboard shortcuts don't match. Transitioning between the two is very difficult because of her muscle memory. She also has difficulties remembering new keyboard shortcuts right after she learns them, requiring her to stop a task to open a browser and search for what she wanted to use. Sometimes, she types so fast that she makes slips during her typical text entry tasks and uses the wrong keyboard shortcut. This requires her to manually undo the consequences of the incorrect shortcut, which is very frustrating for her.

This persona is featured in the final design sketches and storyboards.

Name: Sarah
Background:

- Age: 32
- Gender: Female
- Occupation: Project Manager

Goals and Motivations:

- Keep her technical specifications and meeting notes well organized to support successful project development cycles
- Complete her work as efficiently as possible, as her job demands efficient management of tasks and documentation. She knows this is supported by learning keyboard shortcuts.
- Values precision in her work to avoid any confusion amongst the team members and leadership she works with every day

Challenges:

- Uses both Windows and macOS, and the transition between the two can sometimes be difficult because of the differences in keyboard shortcuts
- Struggles with remembering different shortcuts shortly after learning them and frequently has to Google them when she wants to use them
- Sometimes types so fast that she slips and uses the wrong keyboard shortcut and has to manually undo the consequences, which is frustrating

"I know what I'm missing out on, but I don't have the time to learn…"

Lisa has a strong interest in technology. She is the PM at a software development company, where she organizes and coordinates the engineers that build the product. Her job mostly involves running meetings and maintaining or creating documentation, which she believes she's very good at – and she's very passionate about it. She is a quick learner, and has picked up on some keyboard shortcuts over the last 5 years, all of which she uses efficiently. However, there are some more complex tasks that she still relies on the mouse to complete, just because she isn't aware of existing keyboard shortcuts to accomplish the same tasks. She knows that there's definitely more useful shortcuts out there for her to learn and boost her efficiency, but she struggles to find time during the busy work day to explore them. She wishes that there was an easier way to find them than doing a thorough Google search after catching herself doing something tedious. When she learns a new shortcut, she tends to forget them and wish she could relearn them without disrupting her workflow. She's optimistic about improving her productivity in the future, but hasn't found the time or the best method to go about it yet.

Fig. 4. Final persona: Sarah

6.3 Final Design Sketches and Storyboards

Our final design is called the "Shortcut Companion." It is a software tool designed to assist in enhancing the efficiency of text entry through the use of keyboard shortcuts. It is meant to make keyboard shortcuts more discoverable and to reduce frustrating mistakes the user can make while learning new keyboard shortcuts.

The Shortcut Companion draws from the best features of our individual designs to cover the user requirements and breakdowns identified previously. There are 4 key features to our design, all with initial interface designs shown in the final sketch [5].

Application-based Shortcut Recommendations: Inspired by Design 1, Design 4 and Design 5, the Shortcut Companion incorporates an AI Shortcut Recommendation Assistant powered by machine learning. It provides real-time suggestions for relevant shortcuts, reducing the need for users to memorize them and helping them discover new shortcuts based on their actions. By making shortcuts more discoverable, users will be able to boost their productivity without going out of their way and disrupting their workflow to learn new ones. This feature is shown in the first final storyboard [6].

Context-aware Recommendations: Building on Design 3, the assistant displays context-aware pop-ups when the user holds down a designated key for a certain user-configured amount of seconds. The pop-ups show the most common keyboard shortcuts for the active application, reinforcing users' memory and promoting efficient text entry. This feature makes it easier to learn keyboard shortcuts similar to ones that the user is already familiar with, without having to use a search engine to find them. The user can learn shortcuts directly within their current application.

Shortcut Undo: Inspired by Design 2, the assistant allows users to undo incorrect shortcuts with an undo button, displayed on the interface any time a keyboard shortcut is triggered. This making it easy to recover from slips or guesses, which users cited as frustrations when using keyboard shortcuts during our contextual interviews. This feature is demonstrated in the second final storyboard [7].

Cross-platform Shortcut Translation: Another strong feature of Design 1, the assistant offers an operating system translation mode. Users can activate this mode, select the operating system they want to translate to and from, input a keyboard shortcut they want to translate, and the assistant will display the equivalent shortcut for the desired operating system.

This design provides a wide range of features to assist users in discovering, learning, and using keyboard shortcuts more effectively and avoids the pitfalls from the design critiques. The final design avoids an overwhelming number of popups and allows the user to minimize the interface, creates an application that supports the user in multiple text entry contexts, and allows a configurable and customizable tool to support user preferences.

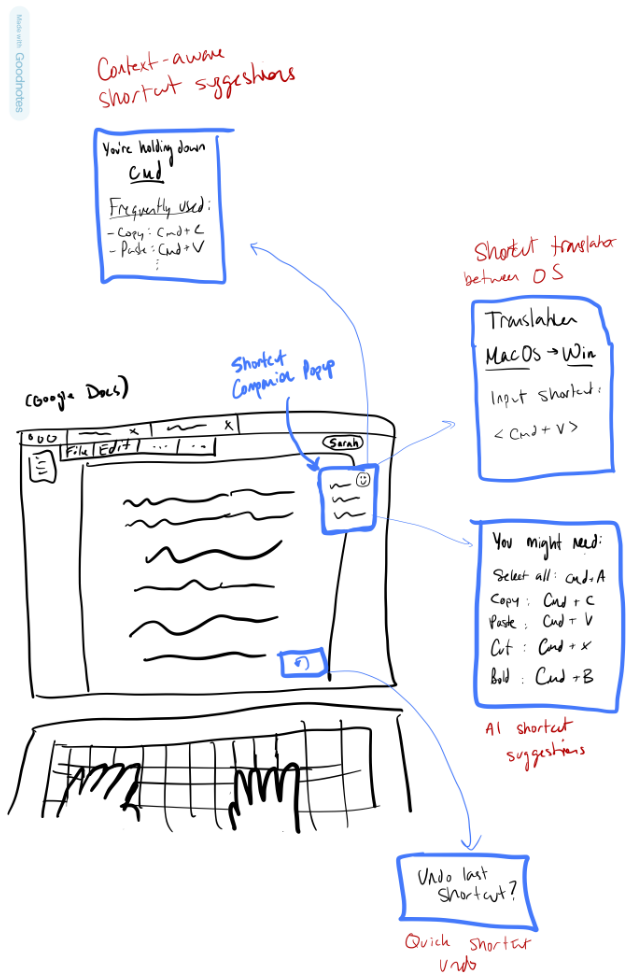

Fig. 5. Final sketch, depicting a user interacting with the shortcut assistant to perform a variety of tasks: Receiving suggestions based on actions or previous keystrokes, undoing previous shortcuts, and translating shortcuts between operating systems

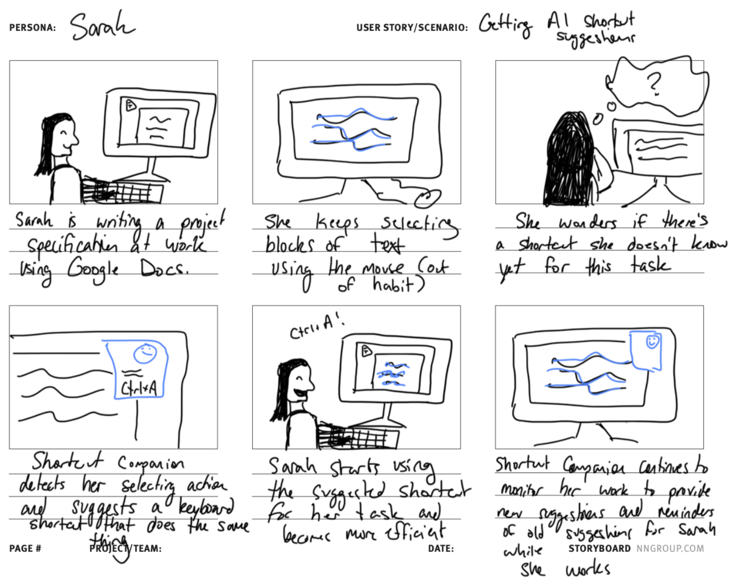

Fig. 6. Final storyboard 1, depicting a user performing a repeated text entry task (selection) and receiving a suggestion from the shortcut assistant

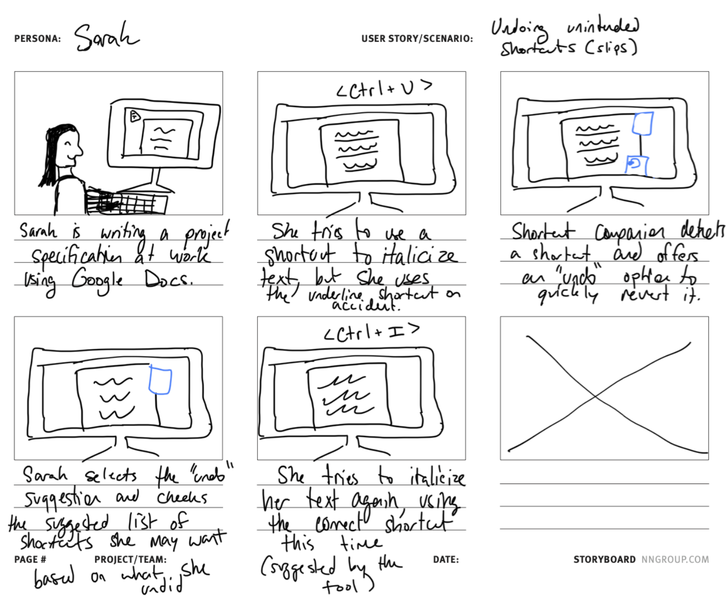

Fig. 7. Final storyboard 2, depicting a user making a slip with a keyboard shortcut, then using the assistant to undo the previous shortcut and receive a recommendation for the correct one

6.4 Paper Prototype

Our final paper prototype is a representation of a MacOS laptop displaying a Google Document website on its screen. This prototype starts off without any additional window opening. After the user holds Command + Control for 3 seconds, a window with an AI assistant pops up on the right side of the screen, displaying suggested shortcuts based on user's previous actions (**??** picture 6). Design 1, 4, and 5 all mention the idea of AI assistance powered by machine learning. We decided to keep this as part of our final prototype. However, we do modify this and add the feature so that users can regenerate shortcut suggestions by clicking on the button (**??** picture 7). We believe this new feature can make our design more discoverable. After the users perform a shortcut, they can undo their shortcut action by clicking the undo button that appears in the bottom right corner (**??** picture 12). We kept this idea from design 3 because we think this is a good solution to our user requirements. We ended up discarding the design to display the shortcut on the screen every time the users perform a shortcut because we believed it was too overwhelming. We also keep the idea of translating the keyboard shortcut of one OS to another. Inspired by design 1, in the final design the users can also translate shortcuts from an old OS to another OS by clicking on the 'T' icon that is in the top menu bar. If the users all of a sudden forget the shortcut they've used frequently, they can hold the "command" key for a user-configured number of seconds and see a window displaying their most frequently used shortcuts in the middle of the screen.

Last but not least, the investigators can "Wizard of Oz" the prototype by altering or adding the paper layers to the paper prototype's screen. They achieve this by removing unnecessary layers and inserting different paper layers into the screen, allowing investigators to exhibit and modify the prototype to reflect the changes made by the participants. For example, when a user clicks the 'T' icon in the menu bar, the investigator can insert the paper layer containing the translation window into the screen to simulate the device's response.

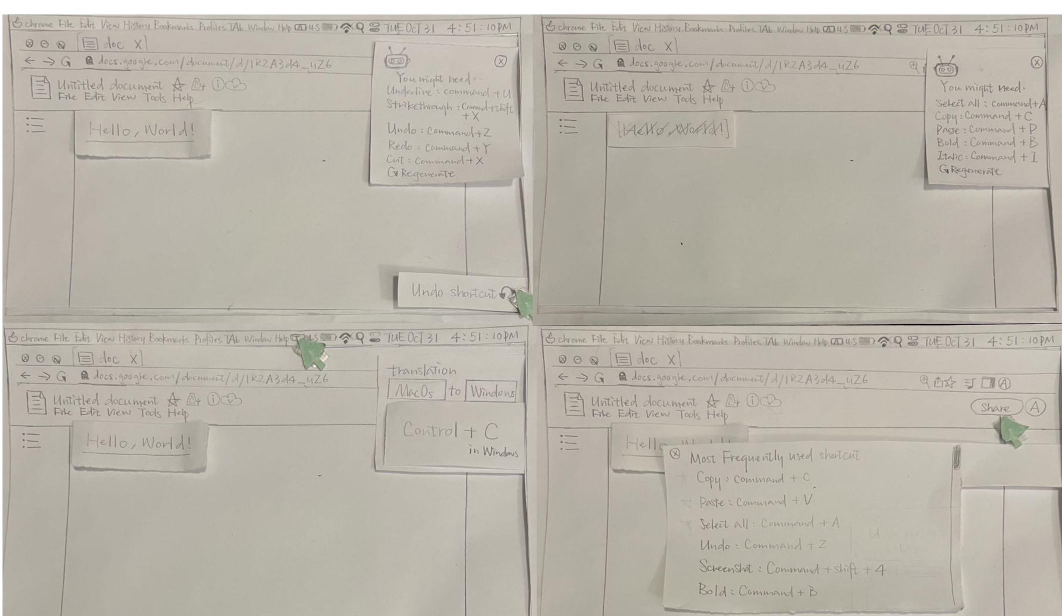

Fig. 8. Paper prototype for Shortcut Companion

7 Usability Evaluation

7.1 Heuristic Evaluation

7.1.1 Purpose We conducted heuristic evaluation to gather expert opinions and critiques of our design of a keyboard shortcut assistant tool. This is the next step in the design process as we begin to evaluate our design against user requirements. Using heuristic evaluation allows us to collect qualitative data related to the usability of our design, which will be valuable to us as we make improvements and prepare for the next round of prototyping.

7.1.2 Method Performing heuristic evaluation with usability experts will help us assess how usable our current design iteration is. Using a low-fidelity paper prototype, each member of the design team demoed the intended use case for the keyboard shortcut application we designed to a "usability expert" and asked them to rate the design using the 10 usability heuristics. This allows us to detect usability problems early in the design stage and improve upon them before the next round of evaluation.

7.1.3 Tasks and Procedures Each designer used a copy of the team's paper prototype to demonstrate several expected user goals and uses for the user interface of the application by using the "Wizard-of-Oz" technique to mock functionality. The usability experts were instructed to evaluate aspects of the demonstrated design against the 10 usability heuristics.

Each designer demonstrated the following tasks:

- Discovering New Shortcuts: Finding a keyboard shortcut for copying text in a document.
- Regenerating Shortcut Suggestions: Forgetting or simply wanting to see different shortcut suggestions for your current context.
- Undoing an Action: Accidentally applying an incorrect keyboard shortcut in your text editor.
- Context-Aware Shortcut Discovery: While working in a specific application, wanting to learn keyboard shortcuts relevant to that application.
- Translating Shortcuts between Operating Systems: If switching from one operating system to another and finding the equivalent keyboard shortcuts for the new OS.

7.1.4 Participants All of the usability experts are CS graduate students, in the age range of 22-25. 3 of the 5 were male, 2 of the 5 were female. They were paired up with each of the 5 designers by the HCI course staff. The heuristic evaluation meetings took place in person, where the designers demonstrated user tasks with a copy of the paper prototype.

Uniqnames: gaoya, jackholl, inair, anarthur, arjav.

7.1.5 Results The usability experts reported 19 usability problems with various severity ratings for 9 of the 10 usability heuristics (see Figure 9 for a chart describing all of the documented issues and their severity ratings).

We aggregate descriptions of the issues with the prototype design into groups for each (relevant) usability heuristic below:

- **1: Visibility of system status**
  - There is no confirmation that shortcuts were successful.
  - There is no confirmation that undoing a shortcut was successful.
- **3: User control and freedom**

  - There are no back buttons on the pop-ups that navigate to a different application feature page.
  - There is no way to see previous shortcut recommendations after regenerating them with the "regenerate" button.
- **4: Consistency and standards**
  - It is unclear which application the "undo" button belongs to (the text entry application, or the Shortcut Companion application itself).
  - The term "translation," used in the context of translating keyboard shortcuts from one operating system to another, is confusing and may be mistaken for translating between human languages.
- **5: Error prevention**
  - If a user makes a typo while searching for a shortcut, the interface should behave reasonably by helping the user recover and showing a message indicating why the application is not performing as they expected.
- **6: Recognition rather than recall**
  - Users have to remember the shortcuts and key combinations to activate the application or some of its features (such as contextual suggestions).
  - "Most frequently used" shortcut pop-up lacks descriptions of the functionality for each of the recommended shortcuts.
- **7: Flexibility and efficiency of use**
  - There are multiple ways to activate the application's features (either with keyboard shortcut combinations or using the mouse).
  - Some of the application's features cannot be activated using only the keyboard, which defeats the purpose of having a keyboard shortcut application.
- **8: Aesthetic and minimalist design**
  - Pop-ups for similar features (ex. context-aware recommendations vs. frequently used) have inconsistent UIs.
  - "OS Translation" feature is hard to find in the application.
  - The meaning of the "regenerate" button is unclear.
  - The "undo" button is not always on screen for user actions.
  - The "undo" button is unnecessary for text entry applications, as "Ctrl+Z" performs an identical action.
- **9: Help users recognize, diagnose, and recover from errors**
  - There is no "redo" button to accompany the "undo" button.
- **10: Help and documentation**
  - There is no documentation or walkthrough for how to use the tool for new users.
  - It is unclear what the input format to the "translate" feature is, because the UI doesn't specify it.

Out of these issues, the most common and severe were the complete lack of instructions or documentation for using the Shortcut Companion, and that users have to remember the shortcut to start up the application (or specific features within the application) (See Figure 9). These two aspects of the design were initially thought to be features, as using keyboard shortcuts is all about memorizing key combinations for certain actions that don't require a visual aid, but this was an oversight by the design team. Users starting to use the application will still need "onboarding" instructions to learn how to use the application for their desired purpose. Additionally, focusing on the "regenerate" button for keyboard shortcut suggestions, there is currently no way to revisit past suggestions after regenerating fresh ones. This was unexpected to the usability experts, as similar applications (like ChatGPT) allow you to revisit past cycles of generated content to provide the most information to users at all times.

Usability Problem	E1	E2	E3	E4	E5
No confirmation of successful shortcut (#1)	1				
No confirmation of undo action (#1)				2	
Missing back buttons on most popup screens (#3)					2
No way to see previous shortcuts after regenerating (#3, #5, #9)	1			4	3
Unclear what application undo button belongs to (#4)			1		
Terminology of "Translation" for keyboard shortcuts is unexpected, expected to be in the context of languages (#4)			3		1
Users have to remember to hold keys to activate the application or its contextual suggestions (#4, #6)	3	2	2		3
Needs to reasonably handle typos and display their effects in user input (#1, #5)					2
"Most frequently used" shortcuts lack descriptions of functionality (#6)				3	
Different tool features are activated in different ways (shortcuts and buttons) (#7)	2				
Some features cannot be activated using only the keyboard (#7)		1			2
Different types of pop-ups for shortcut suggestions vs frequently used shortcuts (#8)	2				
The translation button/feature is hard to find (#8, #10)	2				3
Regeneration button is unclear (#8)		2			
Undo button is not always available (#8)		1			
Undo button is unnecessary for text entry shortcuts (#8)					2
No redo button (#9)	1				
No documentation about how to use the tool (#10)		3	3	4	3
Unclear what the input type for the "Translation" feature is (#10)					3

Fig. 9. Usability problems and their severity ratings, measured against the 10 usability heuristics by usability experts.

7.2 Simplified User Testing

7.2.1 Purpose We conducted simplified user testing to test against user requirements and collect the target audience's feedback and opinions on the usability of our design of a keyboard shortcut assistant tool. By collecting results from the simplified user testings, we can gather valuable feedback and data against user requirements and therefore modify our design according to the result before investing in the high-fidelity. Our user tests are against the following user requirements: 1) Users must be able to discover new shortcuts without consulting external resources. 2) In the event of a user failing to recall a keyboard shortcut, users must be able to recognize potential shortcuts without consulting an external resource. 3) Users must be able to execute their intended shortcut and resume the intended task after a slip (i.e. using the wrong shortcut). 4) Users familiar with a given operating system's keyboard shortcuts must be able to recognize these shortcuts on other operating systems.

7.2.2 Method In this step, we conducted simplified user testing with a target audience via the method of think-aloud. Each group member of our team recruited a participant by using our low-fidelity paper prototype to represent the user interface. The participant was asked to use the paper prototype to perform different tasks and provided feedback. These user tests allow us to explore the real user experience and confusion when they use the prototype.

7.2.3 Tasks and Procedures The simplified user testing took place in person. Each group member, i.e. the designer, used a copy of the team's paper prototype during the interview and utilized the "Wizard-of-Oz" technique to mock functionality while the participants interacted with the user interface. The designer first introduced the design and intention of the design and then explained the procedure of the testing and the think-aloud method in detail to the participant. The participant was asked for consent. Then the designer asked the participants to present four different tasks in sequence without any prior instructions on how to use the application. However, the participants were given the task information on what they should do. They were asked to accomplish the tasks via think-aloud to describe their reactions and thought processes. We designed the tasks to test the main features of our design, each relating to a different feature. These tasks would help us to explore the usability of our design in practice and would provide feedback on whether the design fulfills the user requirement we defined earlier. Therefore, it points us in a direction to improve the design before investing in the high-fidelity. The designer took notes about the difficulties and issues the participants met when the participants were performing the task using think-aloud. Each task took about five min on average to finish. To ensure the quality control of the tasks, we didn't provide any prior instruction on how to use the application to perform the task to the participants to make sure all the participants started from the same point. We also made sure all the designers' copies of the paper prototypes were the same to avoid any potential variable that could lead to different results.

Each participant was asked to present the following tasks in sequence:

- Discovering New Shortcuts: You want to use keyboard shortcuts for copying and pasting text in a document without using the mouse, after doing it with the mouse a few times. Please use the "Shortcut Companion" to discover this shortcut.
- Regenerating Shortcut Suggestions: You've forgotten or you simply want to see different shortcut suggestions for your current context. Use the "Shortcut Companion" to regenerate shortcut suggestions.
- Undoing an Action: You accidentally applied an incorrect keyboard shortcut in your text editor. Use the "Shortcut Companion" to undo this action.

- Translating Shortcuts between Operating Systems: You have recently switched from one operating system to another and need to find the equivalent keyboard shortcuts. Utilize the "Shortcut Companion" to translate the "copy" and "paste" shortcuts from your old OS to the new one.

7.2.4 Participants We recruited five participants for this simplified user testing. Three of them are male, two of them are female. Three of them are 21-year-old, and two of them are 55-year-old or older. While all of them consider themselves as fairly competent with technology, only three of them report that they are familiar with keyboard shortcuts. One is unfamiliar with many shortcuts and one has some experience with using shortcuts. The criteria for participation was that they had taken the survey before and were targeted stakeholders of this design. To ensure the quality of the simplified user testing, we decided to keep the number of participants small. The user testing took place in person. The participants were not given any incentives. Before the test started, they were asked if they consented to take part in a 1-semester-long research project and their participation will be used for a keyboard shortcut research analysis. They could stop the testing whenever they want to and the result would be discarded. After the oral consent, we began to conduct the tests.
To ensure each participant's privacy, we name them Participants 1-5 and use these labels throughout the paper:
Participant 1 is a 55-year-old male who works in the supply chain field within the automotive industry. They consider themselves fairly competent with technology. They have some experience with using keyboard shortcuts.
Participant 2 is a 21-year-old female senior college student who is majoring in CS. They consider themselves fairly competent with technology. They use keyboard shortcuts daily and are familiar with shortcuts.
Participant 3 is a 60-year-old male retiree who worked in IT. They consider themselves fairly competent with technology and keyboard shortcuts. They use shortcuts on a daily basis and are familiar with shortcuts.
Participant 4 is a 21-year-old female senior college student who is majoring in CS. They self-identified as proficient with technology, but unfamiliar with many keyboard shortcuts.
Participant 5 is a 21-year-old male first-year graduate student studying Civil Engineering. They consider themselves fairly competent with technology and keyboard shortcuts and use shortcuts on a daily basis.

7.2.5 Results We have collected the following simplified user testing results:
Discovering New Shortcuts:

- Participant 1's initial approach to launching the "Shortcut Companion" application was marked by confusion. They appeared unsure about how to access this tool and began to guess various key combinations in an attempt to open it. After a few unsuccessful attempts, the designer intervened and suggested that they use "ctrl +command to open the application. This highlights a usability issue related to the clarity and accessibility of the feature for launching the "Shortcut Companion." Users should be able to readily identify how to access the tool without guesswork or external guidance. Once inside the "Shortcut Companion," participant 1's task was to discover a keyboard shortcut for copying and pasting text into a document without using the mouse. Once on the popup screen, participant 01 was able to recognize the shortcut they were looking for and successfully closed the screen. The design initially fails to meet the user requirement that users must be able to discover new shortcuts without consulting external resources (like Google). Participant 1's confusion in accessing the "Shortcut Companion" and their need for external guidance to use "ctrl +command" to open the application demonstrates a lack of clarity and accessibility in the design.

- Participant 2 was trying to find the copying shortcut but they didn't know how to get access to the application in the first place. They began to guess key combinations and attempt to open it by exploring the settings. This highlights an essential issue reflecting the lack of accessibility and discoverability of the application. After being told how to open up the application, they were able to open up the window and find the corresponding shortcut easily. This is against user requirement "In the event of a user failing to recall a keyboard shortcut, users must be able to recognize potential shortcuts without consulting an external resource" because participant 2 was unable to open up the application without consulting an external resource.
- Participant 3 did not know how to open up the keyboard shortcut assistant. When they were told, they noted that it was a strange choice to use "ctrl +command" to open up the shortcut assistant. This is a failure of all requirements. Once they opened up the AI assistant window, they were making fun of how long the loading of new shortcuts in the popup took. This is an indication of dissatisfaction with the long waiting. Once the loading was done, they were able to find the corresponding shortcut. While participant 3 was able to finish the task, they were wondering why there was a "TV" in the popup. It was supposed to be a representation of the AI assistant, a character. Again, this highlights the issue of lack of clarity in the design and the failure of the user requirement.
- Participant 4 was initially confused about how to open up the application, indicating a failure of the user requirement. After being instructed to use the application, they held the keys to try to open the interface. When a "Loading" screen was shown, they were not surprised or upset. The interface showed the "Most frequently used shortcuts for the current application" as the default view. They thought that it was convenient to have a list of recommended shortcuts available as soon as the application opened (since that was their goal). The suggestions were appropriate and made it easy to find the right shortcut and use it without exiting the current text entry application.
- Participant 5 was very confused about where to start since they didn't know how to open up the application. Again, this highlights the issue related to the clarity and accessibility of launching the application and fails the user requirement that the users must recognize potential shortcuts without external resources. Once they were instructed, they could open up the application and see the suggested shortcut after the "loading". Participant 5 reflected that they were familiar with most keyboard shortcuts that the pop-up suggested. After a short browsing, participant 5 was able to find the corresponding shortcut and complete the task.

Regenerating Shortcut Suggestions:

- Participant 1 instinctively looked for a "Regenerate" or "Refresh" button on the paper prototype instead of holding "ctrl+command" again. This led to vocal expressions of confusion on how to proceed. This action and utterance once again demonstrate a gap in the design where essential features of the application might be lacking or unclear. A potential solution is providing clear documentation on how to navigate the application, especially for new users. The design does not meet the requirement that users must be able to recognize potential shortcuts without consulting external resources, as the participant's confusion suggests that users might struggle to navigate and utilize the tool effectively.
- Participant 2 was able to open up the AI assistant because of the previous task and saw the suggested shortcuts that were generated by the AI. However, when the participants

tried to regenerate the shortcut suggestion, they didn't realize the "regenerate" button was clickable. They intuitively closed the AI assistant window and opened it up again, hoping it would generate a new list of shortcuts. The AI assistant did generate new shortcuts and they finally realized the button was clickable. After they clicked the "regenerated", they successfully finished the task. This highlights the issue within our design that the design of the button was not clear enough for the users to use.

- Participant 3 was able to open up the AI assistant this time and see the suggested shortcut. However, they thought that at a glance it was tough to distinguish between the different suggested shortcuts because they were all similar-looking text. They suggested showing picture representations of the keys instead of text. This highlights the flaw in terms of the design choice. When they were in the AI assistant, it was tough for them to find the regenerate button. They suggested that it should stick out in some way. This is once again another design flaw. They've also noted it was strange that closing the pop-up required using a mouse and clicking. They expected to be able to close it with the keyboard as well.
- Knowing how to open the application from the previous task, participant 4 opened the interface and saw a list of suggestions. This time, the expected shortcut was not present. They noticed a "regenerate" button, but they didn't know what it meant at first because the term was confusing to them. This demonstrated the issue that the functionality is unclear to the users and jeopardizes the usability of the application. To see what it would do, they clicked the regenerate button, hoping that it would do what they needed it to. They were not surprised when the regenerate button appeared to shuffle/regenerate the recommended shortcuts. After thinking for a moment and realizing that the application was meant to be like an AI assistant, they realized that the term "regenerating" made more sense and tracked with their present knowledge of AI tools.
- Knowing how to launch the application from the previous task, participant 5 was able to open up the AI assistant and have the pop-up window open. Participant 5 could easily find the "regenerate" button and regenerate a new list of suggested shortcuts and accomplish the task. However, participant 5 wished the pop-up with shortcut suggestions was a scrollable target, instead of just a limited set with the option to regenerate so they could scroll through them afterward. This highlights the issue of the usability of the design in practice. Even though participant 5 was able to finish the task, they showed confusion when closing the pop-up. They intuitively wanted to press escape on the keyboard instead of using the cursor to press the X on the pop-up. This indicates that our design doesn't follow users' intuitive workflow.

Undoing an Action:

- Participant 1 initiated the task by successfully selecting the "Undo" button within the paper prototype using their mouse cursor. While they successfully completed the task as instructed, they expressed surprise that they couldn't use a keyboard shortcut to perform the undo action. This feedback is crucial as it indicates a deviation from the user's expectation. The design fails to meet the user requirement that users must be able to execute their intended shortcut and resume intended tasks after a slip (using the wrong shortcut). The absence of a keyboard shortcut for undoing an action, which is a common user expectation, may negatively impact the tool's usability and the user's ability to recover quickly from errors.
- Participant 2 first performed a shortcut action and then they saw an "undo shortcut" button displayed on the screen. They successfully clicked the button and undone the shortcut without any error. After they undo the shortcut, they expect a confirmation of undoing. This user testing shows that our application can fulfill the user's requirement "Users must

be able to execute their intended shortcut and resume intended task after a slip(i.e. using the wrong shortcut)" since the participant can undo the action and recover from a slip.

- Participant 3 first performed an incorrect shortcut and then saw the undo button appear on the screen. They were able to undo the task by clicking the undo button. They felt the process was intuitive. They were able to finish the task without any instruction.
- Participant 4 first performed an incorrect keyboard shortcut, triggering the application's "Undo shortcut" button displayed on the screen. Note that having the undo button appear in the interface made the user want to click it just out of curiosity. After clicking it, they realized that they could have achieved the same thing with the traditional OS "undo" shortcut (Cmd/Ctrl+Z), but were tempted to click the button because it appeared and felt like a prompt.
- Participant 5 first performed a shortcut and saw the undo button appear on the screen. Participant 5 successfully undo the shortcut by clicking the button but they were confused on "command-Z" showing up as a shortcut suggestion while there was also a pop-up to "Undo Shortcut" so they didn't know if these do the same thing. The pop-up seemed to be more available and intuitive so participant 5 pressed that.

Translating Shortcuts between Operating Systems:

- Participant 1 initiated the task by attempting to find the button that activates the translation mode for OS shortcut conversion. They encountered initial difficulty in locating this button within the paper prototype. While they ultimately identified the button that triggered the translation mode popup, their initial struggle may indicate a need for improved visibility or labeling of this crucial functionality in the actual interface. Once in the translation mode, the participant expressed significant interest in this feature, underscoring its perceived value. However, they seemed to be uncertain about how the translation of shortcuts occurs. They were unsure whether the translated shortcuts are automatically executed upon typing in an old OS shortcut, or if they merely display as suggestions for the user to manually input. This indicates a potential ambiguity in how the feature functions within the design, which could lead to confusion for users. The design does not fully meet the user requirement that users familiar with a given operating system's keyboard shortcuts must be able to recognize these shortcuts on other operating systems. The participant's uncertainty about how the translation of shortcuts occurs and whether the translated shortcuts are automatically executed or merely displayed as suggestions suggest potential ambiguity in the design. This ambiguity could lead to user confusion when translating shortcuts between operating systems.
- Participant 2 tried to convert a MacOs shortcut to a Windows shortcut. Again, they didn't know how to turn up this translation mode in the first place. Once they knew how to open it up, they saw the translation pop-up showing up on the screen. They saw two boxes. It took them a while to realize these two boxes were the drop-down menu because there was no sign indicating they were drop-down. This once again indicates the lack of clarity. Once they've selected the OS, they intuitively inputted the shortcut and saw the conversion. This user testing indicated that our design can fulfill the user requirement of "Users familiar with a given operating system's keyboard shortcuts must be able to recognize these shortcuts on other operating systems" since the participant can convert the shortcut successfully. But it suggests flaws in terms of the design.
- Participant 3 was confused about how to turn on the translation mode. When they were told it was a button on the top bar, they still couldn't find it because it was a very small button in the menu bar. This is a clear failure of the requirement that users must recognize

potential shortcuts without external resources because they couldn't find the translation mode without using external resources. Once they were in the translation mode, they were able to finish the translation. However, they expected the translation mode to always be on, instead of a separate popup screen. They brought up that it shouldn't prevent the user from being suggested shortcuts based on context.

- Participant 4 once again had trouble locating the translation mode. This again points out the issue of lacking discoverability and failing the user requirement. After being instructed, they clicked on the "Translate" option in the tool's interface to activate the OS translation mode. They understood that the interface asked the user to select a new and old OS to translate to and from. They did not have trouble understanding how to select and switch these fields. However, they had trouble understanding the expected input method when the interface asked the user to input a keyboard shortcut. The intention was to have them input a keyboard shortcut using the keyboard that would get translated, but that would not make sense if the keyboard layout was not the "old OS" layout (since the buttons would not exist). They were very confused and did not understand how to input a different OS key. They instead expected that there would be some type of search bar that allowed them to type the action that they wanted to perform (ex."copy") and see a list of shortcuts matching that query and allowing them to select one. The existing interface did not allow the user to successfully translate a keyboard shortcut across systems because of a lack of interface detail. It was very unclear how to go about reaching this goal.
- Participant 5 was unable to turn on the translation mode without any instruction in the first place. Again, this indicates the design flaw of our design. Once they were instructed, they were able to turn it on and see the pop-up. Once in the translation pop-up, the participant was pleased with the functionality and was satisfied with the idea of translating shortcuts between OS. They believed this would be a useful functionality in practice. They had no trouble translating the shortcut but they expected a search capability to be present in the translation pop-up so that if they forget the exact shortcut they wanted to translate, they could still translate the shortcut without recalling anything. This again fails the user requirement that in the event of a user failing to recall a keyboard shortcut, users must be able to recognize potential shortcuts without consulting an external resource.

8 Final Design and Functional High-Fidelity Prototype

In the final prototype, the first change we made from the initial prototype was the way users opened the application. From previous evaluations, the participants reflected that they didn't even know how to open the application in the first place. Based on this feedback, we changed the way so that the user can open our application by clicking the keyboard icon in the menu bar and launching the "Shortcut Companion"(10.Fig.1). This will improve our accessibility and discoverability. After clicking the "Open Shortcut Companion", the application is launched automatically and a confirmation pops up for a few seconds, and then an arrow displays the menu bar of the keyboard companion, helping to improve the visibility of the system status(10.Fig.2). After the application is launched, the user may click the icon "S" that's on the menu bar to see four options: AI assistant, shortcut translation, frequently used shortcut and help. We used to let the user use a keyboard shortcut to open these functionalities up but we changed it and made them more accessible by adding a drop-down menu(10.Fig.3). Based on the previous evaluation, We also added a help document(10.Fig.4) in order to solve the issue. The users may click the "Help" and see a pop-up window showing "How to get started?" documentation that can help users get started with the application. They may close it by clicking the 'x' on the top left corner. To translate a shortcut from one OS to another OS, the user may click the "Open Shortcut Translation" button under the dropdown menu. After clicking the button, a window will open with “Shortcut Translator” being the title. The users may choose an OS to translate from and an OS to translate to by typing or clicking the desired OS(10.Fig.5). After choosing the OS, the users may input the selected shortcut using the keyboard. The window will automatically show the translation. Different from the initial design, we added one more functionality to this shortcut translation so that If the user forgets what shortcut they were looking for, they might click “search here” to search. Users may type the name of the desired shortcut and the translation will display automatically(10.Fig.6). To generate suggested shortcuts based on the current task, the users may click the “Open AI Assistant” under the dropdown menu. Once they click, a window will display showing suggested shortcuts based on the current task/page. The users may click “regenerate” to generate new suggestions(10.Fig.7). The user may gain different suggestions by clicking the “Click here to regenerate” button. The only two changes we made to the "AI Assistant" were the wording and making the color of the word change while hovering over the button. To see the users’ most frequently used shortcut, they may click the “Most Frequently Used Shortcut” under the dropdown menu bar. Once the window shows up, they may see their most frequently used shortcuts(10.Fig.8). The only change we made based on the evaluation was that the users may search for any shortcut by typing the name of the shortcut into the search bar. Last but not least, the users may undo the undesired shortcut by clicking on the “undo” button. The button appears on the bottom right corner of the screen once they perform a shortcut(10.Fig.9) and it will automatically disappear after a few seconds.

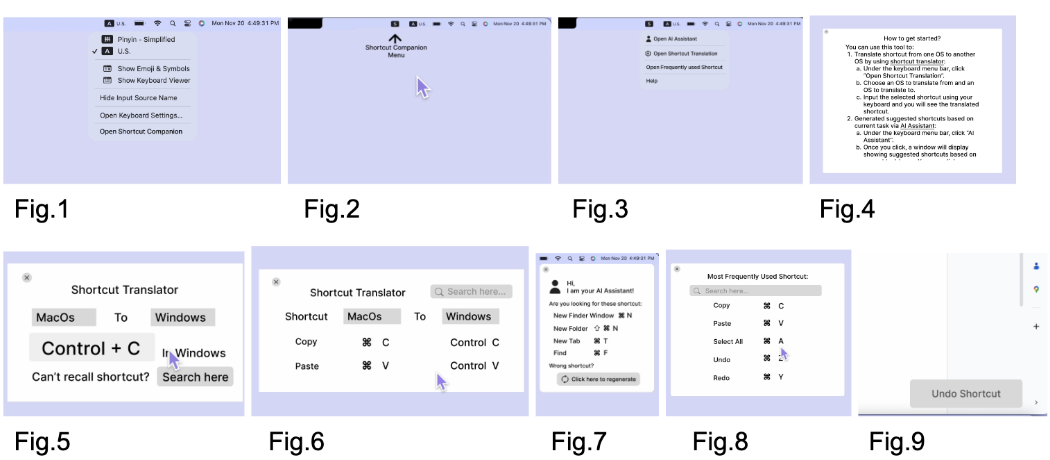

Fig. 10. Final functional prototype (Figma) for Shortcut Companion

9 User Evaluation

The quantitative evaluation of our final prototype in this study serves as a systematic means to objectively measure and quantify user perceptions, confidence levels, and usability metrics related to keyboard shortcuts both with and without the shortcut companion application. By employing structured Likert scale questionnaires and task-based assessments, this evaluation aims to numerically assess participants' motivation, confidence in cross-platform shortcut usage, recovery from errors, and rediscovery of shortcuts.

9.1 Method

For our methodology, we utilized a two-part Likert scale questionnaire to gauge participants' motivation and confidence levels in handling keyboard shortcuts. This involved baseline questions (pre shortcut companion usage) regarding motivation and confidence in translating shortcuts across operating systems, recovering from mistakes, and (re)discovering shortcuts. Once the user completed the baseline questions, they performed a series of tasks related to the subset of user requirements selected earlier. After performing each of the tasks, participants were prompted to answer a question related to the user requirement with Likert scale answer choices. These answers will be compared to the baseline in our analysis of the data. The subset of user requirements we have chosen to evaluate against are:

- 1 - Users must be able to discover new shortcuts without consulting external resource (like Google)
- 5 - Users familiar with a given operating system's keyboard shortcuts must be able to recognize these shortcuts on other operating systems
- 7 - Users must be able to execute their intended shortcut and resume intended task after a slip (i.e. using the wrong shortcut)
- 8 - In the event of a user failing to recall a keyboard shortcut, users must be able to recognize potential shortcuts without consulting an external resource

9.2 Apparatus

Below are the Likert scale questionnaires used in our quantitative user evaluation.

Baseline Likert Scale Questionnaire:

- "How motivated do you feel to explore and use new keyboard shortcuts?"
  Not motivated at all | Not very motivated | Neutral | Somewhat motivated | Very motivated
- "Do you feel comfortable translating familiar shortcuts across different operating systems?"
  No confidence | Somewhat confident | Neutral | Slightly Confident | Very confident
- "When making a mistake with a keyboard shortcut, how confident are you that you'd be able to quickly recover from the mistake and use the correct shortcut?"
  No confidence | Somewhat confident | Neutral | Slightly confident | Very confident
- "When forgetting a keyboard shortcut, how confident are you in your ability to quickly rediscover it without external assistance?"
  No confidence | Somewhat confident | Neutral | Slightly confident | Very confident

Post Task Likert Scale Questionnaire:

- "How motivated do you feel to explore and use new keyboard shortcuts with the system?"
  Not motivated at all | Not very motivated | Neutral | Somewhat motivated | Very motivated
- "After using the system's cross-platform translation feature, how confident are you in recognizing and using familiar shortcuts across different operating systems?"
  Unconfident | Less confident | No change | More confident | Very confident

- "When making a mistake with a keyboard shortcut, how confident are you using the system's "undo" shortcut to quickly recover and use the correct shortcut?"
  Unconfident | Less confident | No change | More confident | Very confident
- "When forgetting a keyboard shortcut, how confident are you in quickly rediscovering it with assistance from the system?"
  Unconfident | Less confident | No change | More confident | Very confident

9.3 Tasks and Procedures

Our structured procedure began with participant introductions and consent, followed by demographic and familiarity assessments. The data collection process involved 12 participants, with meticulous recording of Likert scale responses for each requirement. This methodical approach enabled the systematic compilation and organization of data, facilitating subsequent in-depth analysis of user confidence and motivation pertaining to keyboard shortcut usage across different tasks and system functionalities. The tasks the user was asked to complete were:

- Discovering New Shortcuts: You want to use keyboard shortcuts for copying and pasting text in a document without using the mouse, after doing it with the mouse a few times. Please use the "Shortcut Companion" to discover this shortcut.
- Translating Shortcuts between Operating Systems: You have recently switched from one operating system to another and need to find equivalent keyboard shortcuts. Utilize "Shortcut Companion" to translate the "copy" and "paste" shortcuts from your old OS to the new one.
- Undoing an Action: You accidentally applied an incorrect keyboard shortcut in your text editor. Use the "Shortcut Companion" to undo this action.
- Regenerating Shortcut Suggestions: You've forgotten, or you simply want to see different shortcut suggestions for your current context. Use the "Shortcut Companion" to regenerate shortcut suggestions.

9.4 Participants

A total of 12 participants completed the evaluation and questionnaire.
Participant 1: A 21-year-old college female student majoring in computer science who identifies as moderately proficient with technology and keyboard shortcuts.
Participant 2: A 21-year-old college female student majoring in computer science who identifies as moderately proficient with technology and keyboard shortcuts.
Participant 3: A 21-year-old college female student majoring in computer science who identifies as highly proficient with technology and keyboard shortcuts.
Participant 4: A 21-year-old college female student majoring in computer science who identifies as moderately proficient with technology and keyboard shortcuts.
Participant 5: A 21-year-old college female student majoring in computer science who identifies as moderately proficient with technology and keyboard shortcuts.
Participant 6: A 23-year-old male software engineer who identifies as highly proficient with technology and keyboard shortcuts.
Participant 7: A 60-year-old male who identifies as highly proficient with technology but not proficient with shortcuts.
Participant 8: A 24-year-old male who identifies as highly proficient with technology but moderately proficient with shortcuts.
Participant 9: A 55-year-old female works in the education industry who identifies as not proficient with technology but uses a computer daily.
Participant 10: An 18-year-old male high school student who identifies as highly proficient with

technology and keyboard shortcuts.
Participant 11: A 21-year-old male college student majoring in economics who identifies as moderately proficient with technology and keyboard shortcuts.
Participant 12: A 22-year-old male college student majoring in mechanical engineering who identifies as highly proficient with technology and keyboard shortcuts.
Among all 12 participants, 6 participants are female. 6 participants identify as moderately proficient with keyboard shortcuts, whereas 4 participants classify themselves as highly proficient with shortcuts and 2 participants as not proficient.

9.5 Results

The evaluation employed a Wilcoxon Signed-rank test, a non-parametric statistical test suitable for comparisons between users' baseline perceptions and their post-interaction assessments with the shortcut companion application. Four distinct null hypotheses were formulated to explore whether the tool had no significant impact on users' motivation to learn keyboard shortcuts, their confidence in translating shortcuts across operating systems, their ability to recover from incorrect shortcuts, and their proficiency in rediscovering forgotten shortcuts. Conversely, corresponding alternative hypotheses were proposed, anticipating a positive impact of the tool on these aspects of user confidence and behavior. This statistical approach aimed to rigorously analyze the data obtained from individual structured Likert scale questionnaires and task-based assessments, providing substantive evidence regarding the tool's impact on users' perceptions and usability metrics related to keyboard shortcut usage.
Null Hypotheses:

- The users were not more motivated to learn keyboard shortcuts.
- The users were not more confident in translating keyboard shortcuts across operating systems.
- The users were not more confident in undoing and recovering from an incorrect keyboard shortcut.
- The users were not more confident in quickly learning, rediscovering, or remembering a forgotten keyboard shortcut.

Alternative Hypotheses:

- The users were more motivated to learn keyboard shortcuts.
- The users were more confident in translating keyboard shortcuts across operating systems.
- The users were more confident in undoing and recovering from an incorrect keyboard shortcut.
- The users were more confident in quickly learning, rediscovering, or remembering a forgotten keyboard shortcut.

The data used for statistical analysis can be seen in Appendix F.2 [67]. The results of the statistical analysis are as follows:

Motivation to Learn Keyboard Shortcuts: A Wilcoxon Signed-rank test did not show a significant effect ($Z = -1.0874$, $p = 0.3691 > 0.05$). Therefore, we fail to reject the null hypothesis. There is not sufficient evidence to conclude that users were more motivated to learn keyboard shortcuts with the tool.

Confidence in Translating Shortcuts Across Operating Systems: The analysis revealed a significant effect ($Z = -2.8175$, $p = 0.003906 < 0.05$). Hence, we reject the null hypothesis. Users were more confident in translating keyboard shortcuts across operating systems after using the tool.

Confidence in Undoing Incorrect Keyboard Shortcuts: There was a significant effect observed (Z = -2.0867, p = 0.03906 < 0.05). Consequently, we reject the null hypothesis. Users exhibited increased confidence in undoing incorrect keyboard shortcuts with the assistance of the tool.

Confidence in Rediscovering Forgotten Shortcuts: The analysis demonstrated a significant effect (Z = -3.052, p = 0.0009766 < 0.05). Thus, we reject the null hypothesis. Users were more confident in quickly rediscovering forgotten keyboard shortcuts with the aid of the tool.

10 Discussion

10.1 Initial Survey

10.1.1 Usage Patterns: As expected, an overwhelming majority of respondents reported having experience with using laptop keyboards, external keyboards, and smartphone touchscreens. A more interesting follow up question we asked was "which of the (...) text entry methods is your favorite to use?" Over 75% of the respondents chose either an external keyboard or laptop keyboard, while only 21% chose a smartphone touch screen. External keyboards emerged as the top favorite, likely due to ease of use and availability. Smartphone touchscreens, though popular, were not favored primarily due to usability concerns. This implies there is value in improving this text entry method as it is widely used, yet not the favorite for many users. Multiple users stated annoyances related to key size, which could explain the low number of respondents choosing a smartphone touch screen as their favorite text entry method.

10.1.2 Ease of Use: From the survey results, laptop and external keyboards were perceived as "very easy" or "easy" to use by the majority. This is not surprising, as these are the text entry methods most respondents encounter on a daily basis and they are most familiar with. Smartphone touchscreens received mixed opinions, indicating usability challenges. We hypothesize two reasons for this. The first is familiarity and adaptability: individuals that grew up with touchscreen devices may find this technology intuitive, while older generations who are not as familiar may find themselves struggling to adapt. The second is accessibility challenges: for users with fine motor challenges or visual impairments, touchscreens can prove difficult to use as they are small relative to other types of keyboard. Moving forward, we discovered that voice entry and handwritten translations presented a varied user experience. No respondents consider these "very easy" methods of text entry, and few consider them "easy". Additionally, 85.7% of respondents reported that they had never tried AR/VR text entry technology before. This implies that there is an extreme drop in the quality of text entry technology if we don't consider the top three most popular (external, laptop, touchscreen).

10.1.3 Preferences and Accessibility: A large portion of respondents stated that autocorrect is a crucial feature for their text entry experience. However, it should be noted that multiple participants reported frustrations with the autocorrect functionality that they currently use. This is not an extremely surprising result, as the concept of an autocorrect tool sounds great, but in practice it can be annoying to use and prone to making mistakes. In addition to autocorrect, users responded that predictive text, keyboard sounds, and swipe to text are useful features. Our final question of the survey accessed about what accessibility features our respondents considered important. Notably, a significant portion (60.7%) found the inquiry about accessibility features not applicable to them, potentially signifying a gap in awareness or individual needs for such features. However, for those who expressed their preferences, a majority (54.5%) emphasized the importance of large text size and high contrast, showcasing a critical need for clear and easily readable text. Customizable programmable keys were also highlighted as significant by 36.4% of respondents, indicating a desire for personalized and efficient text input methods. Additionally, the feedback from two individuals regarding frustration with autocorrecting tools highlights an area for improvement.

10.1.4 Themes: Our investigation into text entry methods illuminates considerations that hold significant potential for the design and enhancement of user experience. Primarily, focusing on ease of use emerges as a paramount objective. This involves carefully incorporating user feedback to refine existing methods and aligning them with users' preferences and expectations. Additionally, attention must be directed towards the improvement of less favored methods, specifically voice entry and handwritten translation, striving to make them more user-friendly and effective.

10.2 Contextual Inquiry

The contextual inquiry conducted has provided invaluable insights into users' context of use and their interactions with keyboard shortcuts. The structured approach allowed us to gain a deeper understanding of users' procedures, satisfaction, frustration, and efficiency in the context of keyboard shortcut usage, which is fundamental to optimizing text entry methods.

The affinity diagram constructed from the contextual inquiry illuminated critical aspects of users' context of use. Four main scenarios emerged where users frequently utilized keyboard shortcuts: opening and closing applications, navigation around the computer, formatting of text, and transferring text. These scenarios shed light on the practical situations where keyboard shortcuts are extensively employed. Additionally, users' preferences for certain keyboard shortcuts, such as using shortcuts matching the first letter of the intended functionality, emphasized the importance of intuitiveness in shortcut design. Among the identified scenarios, the transfer of text emerged as the most prevalent context of use for keyboard shortcuts. All participants mentioned the use of copying, cutting, and pasting shortcuts, emphasizing the significance of these actions in their workflow. This finding accentuates the necessity of prioritizing and enhancing keyboard shortcuts associated with text transfer functionalities to streamline users' tasks effectively. While navigation emerged as a context of use, it is noteworthy that three participants explicitly favored using the mouse over keyboard shortcuts for navigation. This indicates a potential area for improvement in the design and intuitiveness of keyboard shortcuts related to navigation. Enhancing the intuitive nature and ease of use of navigation shortcuts may encourage users to adopt keyboard-based navigation more readily, thereby enhancing overall efficiency.

The user requirements identified from the contextual inquiry underscored key areas for improvement in keyboard shortcut usage. Users expressed a need for discoverability, efficiency, ease of recall, and seamless cross-platform usability. Integrating these requirements into the design of keyboard shortcuts will not only enhance user satisfaction but also boost productivity. Participants frequently resorted to external resources like search engines to discover new keyboard shortcuts, highlighting the need for better discoverability within the system. Enhancing discoverability would reduce disruptions in the workflow, enabling users to quickly and efficiently integrate new shortcuts into their repertoire. Users voiced frustrations with different keyboard shortcuts across operating systems, emphasizing the importance of uniformity in shortcut designs. Ensuring consistency in keyboard shortcuts across platforms will facilitate a seamless transition for users moving between different operating systems, minimizing the learning curve.

10.3 System Design and Prototyping

The initial survey and contextual inquiry provided us with an in-depth understanding of the context of use. This allowed us to develop primary personas that represent real end users of the product. By having personas, we were able to create our individual prototypes with the end user in mind. This meant that their abilities, goals, and frustrations were taken into account when designing. Essentially, the persona enables the developers to design a solution while focusing on the stakeholders, rather than their personal beliefs or implicit biases. Additionally, by having each group member create a persona and low fidelity design, we could critique each idea and converge on a final persona and design that best fit that user. Each persona and design was unique, which allowed for a productive discussion into various perspectives that a user might have. For example, we decided that one design that showcased a pop-up for every user action could be overwhelming, so we decided to limit the number of pop-ups to only relevant and repeated shortcuts in our paper prototype. The final persona could potentially resonate or conflict with many of the functionalities of each individual design, so we critiqued and combined the individual designs into a final low fidelity prototype.

The group's low fidelity prototype was then checked to ensure it fulfilled all of the primary persona's goals and did not bring up any frustrations. However, just because the group converged on a prototype design does not mean the users will actually enjoy the product. All of our personas and critiques were solely a discussion and educated guess within the context of use. The way to ensure the viability of our prototype is to evaluate it with various end users and stakeholders. Therefore, the design was developed into a paper prototype that showcases the various features and functionalities of the product. This paper prototype will be tested with users who will interact with it, and a "Wizard-of-Oz" technique will be used to mimic real functionality for a more realistic user experience. The reasoning behind using a paper design is to make the tester believe not much time was spent creating the prototype, which leads them to criticize it without worry.

Our paper prototype, as described in [6.4], mimics a computer screen, and we layer it with new pieces of paper to implement the "Wizard-of-Oz" technique for showing new screens. One example of this functionality is when a pop-up appears for translating between MacOS and Windows shortcuts. This can be mimicked using a new piece of paper with the same background and a pop-up in focus on the page.

10.4 Interactive System Evaluations

10.4.1 Heuristic Evaluation: The group tested the paper prototype with five usability experts that commented on the design's obedience to Jakob Nielson's 10 usability principles. Each expert was walked through five tasks that the prototype was able to Wizard-Of-Oz, and they took notes on what usability principles were broken. They ranked the usability errors on a scale of 0 to 4, 0 being not an error and 4 being a severe error. The premise behind having five experts is that each might notice errors that the others do not. When all the results are combined, it was assumed most of the prominent errors will have been discovered by at least one of the experts. After discussing the individual results and consolidating them into a table, we were able to see which experts found errors with each of the ten principles, and how severely they rated each error. We used this table to prioritize errors through a combination of the most severe and most common errors. If a failed principle was ranked as a high severity by multiple experts, it became very obvious from the table layout. We chose to prioritize these errors, followed by errors that were either severe or very common but not as severe.. For example, four out of the five experts ranked the lack of documentation (heuristic #10) as a severity 3 or above. This means that our next prototype should absolutely focus on including a help page for onboarding new users. The next feedback we chose to prioritize was when a few experts found severe errors. While not every expert might have discovered this, we discussed how this is actually the goal of the heuristic evaluation process. By having five separate experts, at least one will typically find severe errors that the others do not. An example of this is the feedback we got on our regeneration button. Only three out of the five experts found the error, and only two rated it as a severity of 3 or above. The errors were a combination of a lack of: user control and freedom (heuristic 3), error prevention (heuristic 5), and recovering from errors (heuristic 9). This was due to the fact that when users click the button to regenerate shortcuts they cannot undo this action or see previous shortcuts. From this feedback, we decided that we need to offer a way to show the history of generated shortcuts. Finally, we prioritized all feedback that was commonly discovered by most of the experts. While they might not have been as severe, they stood out to all of the experts, and we assumed this would be the case for our users as well. An example of this type of feedback was when four out of the five experts rated an error a severity 2 or 3. The error had to do with the fact that users have to remember to hold down the keys to activate Shortcut Companion. The experts found that this broke the usability heuristics of consistency and standards (4) and recognition rather than recall (6). The intended fix to this is to make the activation to Shortcut Companion more intuitive, so that users are able to recall it with ease.

10.4.2 Simplified User Testing: The conducted simplified user testing aimed to validate the design of a keyboard shortcut assistant tool against predefined user requirements while gathering valuable feedback for further refinement. The method involved employing think-aloud sessions with recruited participants to perform tasks using a low-fidelity paper prototype representing the user interface. These tasks were carefully designed to test various features and aspects of the tool, simulating real user experiences and gauging their usability. The results highlighted several usability issues, such as confusion in accessing the application, lack of clarity in launching various features of the application, and difficulties in recognizing potential shortcuts without external resources. The user testing phase brought to light significant challenges related to the "Shortcut Companion" tool's accessibility and usability. Participants encountered obstacles ranging from difficulty in accessing the application to dissatisfaction with loading times and representation issues within the interface. For instance, the struggles faced by Participant 1 and Participant 2 in finding access key combinations and needing external guidance exposed critical clarity and accessibility issues. Participant 3's dissatisfaction with loading times underscored a lack of seamlessness in the tool's performance, while Participant 4's positive response to the recommended shortcuts highlighted the potential usability of the application. These varied experiences emphasized the prevalence of challenges related to access and clarity, within the "Shortcut Companion" tool. The findings from user testing shed light on crucial usability issues regarding the "Regenerate Shortcuts" functionality. Across multiple participants, a few common threads emerged: struggles in discovering buttons, uncertainty surrounding feature functionalities, and complexities in distinguishing and engaging with various UI elements. These collective experiences highlight the need for enhanced design cues and more intuitive navigation, specifically emphasizing the visibility of buttons and alignment with users' natural workflow. The user testing outcomes regarding the "Undo" functionality also reveal crucial insights into the application's usability. Participant 1's surprise at the absence of a keyboard shortcut highlights a significant deviation from expected user behavior, emphasizing the design's failure to fulfill user expectations and meet the requirement for seamless recovery from errors. Conversely, Participant 2's successful use of the provided "undo" button showcases the application's ability to meet user needs by facilitating the execution and recovery from unintended actions. Participant 3's intuitive experience in utilizing the undo button underlines the usability of the feature without requiring explicit instructions. However, Participant 4's interaction with the undo button as a prompted action indicates a potential user inclination to respond to visual cues, even when a traditional OS shortcut exists. Participant 5's confusion between the provided "Undo Shortcut" and the OS-based "command-Z" suggests a need for clarity regarding their respective functions, highlighting a possible discrepancy in user understanding. These diverse experiences underscore the importance of aligning user expectations, ensuring clarity in functionality, and facilitating intuitive recovery mechanisms to optimize the usability of the "Undo" feature within the application. The insights gained regarding shortcut translation between operating systems revealed several design challenges. One recurrent issue is the difficulty in activating the translation mode, specifically not being able to find the translation mode button without external guidance; this highlights a significant failure in meeting the requirement for users to recognize potential shortcuts autonomously. Participants also encountered ambiguity about how the translated shortcuts function within the design, leading to uncertainty and potential confusion during usage. Additionally, participants expressed expectations for a more intuitive input method and search capability within the translation pop-up, both of which weren't met, indicating a gap in usability when recalling specific shortcuts. The simplified user testing outcomes strongly advocate for iterative improvements in the design, focusing on enhancing accessibility, reducing ambiguity, and refining visual cues within the interface. These findings offer critical guidance for refining the tool, aligning its functionalities more closely with user expectations, and ensuring a more intuitive user experience in subsequent iterations.

10.4.3 Final Prototype User Evaluation: The second iteration of user evaluation occurred with the high-fidelity prototype. A subset of user requirements were chosen based on the evaluations we wished to perform. Next, we developed two sets of Likert scale questions to ask participants that served to test our fulfillment of the user requirements. The first set served as a baseline, and consisted of asking the participant about their current background and comfort with keyboard shortcuts. Following the baseline survey, the user interacted with the high-fidelity prototype in four tasks that pertained to our requirements. After completing the tasks, the user responded to four more Likert scale questions to gauge their change in understanding of keyboard shortcuts. To test our performance, we first chose a statistical test and created four null hypotheses. Each null hypothesis was formatted to assume there was no difference between the users' baseline comfort with keyboard shortcuts and their comfort following the demo with the prototype. Then, following the results from our statistical analysis, we could verify if these hypotheses could be rejected, in which case a difference was statistically likely, or inconclusive. To perform our statistical analysis, the results from the Likert scale questions were input into a Wilcoxon Signed-rank test. We compared the results to an alpha value of .05, so if our p-value was less than .05 we were able to reject the null hypothesis. In our case, this meant that the prototype was effective for the given user requirement. For the first user requirement, which surrounded the motivation to learn keyboard shortcuts, we failed to accept the null hypothesis. There wasn't sufficient evidence to conclude that users were more motivated to learn keyboard shortcuts with the tool. This result prompts further exploration into factors influencing user motivation and potential adjustments to the tool to enhance its impact. Next, our hypothesis about confidence in translating shortcuts across operating systems was able to be rejected, which suggests that Shortcut Companion effectively enhances users' ability to use multiple different operating systems with ease. The null hypothesis about increasing confidence in undoing incorrect keyboard shortcuts was also rejected. Users exhibited heightened confidence in undoing incorrect keyboard shortcuts with the assistance of the tool. This finding implies that the tool effectively supports users in recovering from errors, contributing to a more resilient user experience. Finally, increasing confidence in rediscovering forgotten shortcuts was shown to be statistically likely from our analysis. Users were more confident in quickly rediscovering forgotten keyboard shortcuts with the aid of the tool. This result showcases the tool's strength in facilitating users' memory recall and highlights its potential as a valuable resource in enhancing proficiency. By performing quantitative user analysis, we were able to show with some evidence that our prototype effectively increases users' experiences with learning, translating, undoing, and discovering keyboard shortcuts. While not all aspects showed a significant effect, the positive outcomes suggest that the tool holds promise in certain domains and may benefit from targeted improvements in others. Further research and user feedback will inform refinements to optimize the tool's overall effectiveness, specifically in the sector of increasing the motivation to learn shortcuts.

11 Conclusion and Future Work

In conclusion, our project has navigated through the entire user-centered design process, which included survey design, contextual inquiry, the creation and iteration of multiple prototypes, and conducting system evaluations on these prototypes. Our research started by scrutinizing user experiences and preferences, highlighting VR/AR, voice entry, and handwritten translation as areas facing notable challenges in ease of use. Despite the potential of VR/AR, its scope was beyond our project's timeline, leading us to focus on exploring voice entry and handwritten translation for further research. The contextual inquiry interviews yielded crucial insights into users' struggles with complex shortcuts, particularly across various operating systems. Discovering these pain points emphasized the need for intuitive design, cross-platform consistency, and improved discoverability to elevate user efficiency and satisfaction in utilizing keyboard shortcuts. Transitioning to the design phase, we carefully crafted user personas based on contextual insights and converged on a low-fidelity paper prototype named "Shortcut Companion." This tool, driven by machine learning, suggests shortcuts based on user actions and regenerates previously used shortcuts, aiming to enhance recall and discovery. The subsequent simplified user testing and heuristic evaluation unveiled essential refinements needed for the keyboard shortcut assistant tool, highlighting issues around discoverability, recall, and usability, particularly within the translation mode and shortcut regeneration functionalities. Our last stage of the user-centered design process was developing a high-fidelity prototype and conducting quantitative user evaluations on it. This was performed by developing baseline and follow-up Likert scale questionnaires used to gauge how our prototype faired against our user requirements. We performed null hypothesis testing using a Wilcoxon Signed-rank test to see which user requirements were statistically likely to be fulfilled. From this evaluation, we found that users were more confident in: translating shortcuts across operating systems, undoing incorrect shortcuts, and rediscovering forgotten shortcuts. Our only user requirement that we could not prove was fulfilled was an increased motivation to learn keyboard shortcuts. This suggests that future work is needed in enhancing our user experience and showing users the possibilities and benefits to using Shortcut Companion. By iterating on our prototypes, revisiting our understanding of the context of use, and honing in on user requirements, we can continue to use the user-centered design process to improve our product. Specifically, we would like to enhance our interface by increasing findability and discoverability, so that users can seamlessly use Shortcut Companion.

Acknowledgments

This paper was continuously and thoroughly reviewed by EECS 593 staff, and we value their constructive feedback to our research.

References

[1] Ola Kristensson. 2009. Five Challenges for Intelligent Text Entry Methods. *AI Magazine Useable AI Vol. 30 No. 4: Winter 2009* (2009). https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2269

[2] C.G. Leedham and Y. Qiao. 1992. High Speed Text Input to Computer using Handwriting. *Instructional Science – Computers and Writing* (1992). https://link.springer.com/chapter/10.1007/978-94-011-2674-8_16

[3] I. Scott MacKenzie and R. William Soukoreff. 2002. Text Entry for Mobile Computing: Models and Methods, Theory and Practice. *Human–Computer Interaction* (2002). https://www.yorku.ca/mack/hci3-2002.pdf

[4] Shari Trewin. 2003. Automating Accessibility: The Dynamic Keyboard. *Association for Computing Machinery* (2003). https://dl.acm.org/doi/pdf/10.1145/1029014.1028644

[5] Shari Trewin and Helen Pain. 1998. A Study of Two KeyboardAids to Accessibility. *People and Computers XIII* (1998). https://link.springer.com/book/10.1007/978-1-4471-3605-7