Improving One-Handed Textual Entry on Mobile Devices

ZIRUI ZHAO*, FRANKLIN KONG*, YICHEN ZHONG*, SANG HYUN SHIM*, and ISAAC TONG*, University of Michigan, USA

With highly developed information technology, people have become accustomed to using virtual keyboards to type and text in various situations. In many scenarios, people need to type with only one hand and use the other hand to do other things. Nonetheless, mobile device keyboards still lack comprehensive support for one-handed typing, potentially leading to accidental touch and missed touch, which hinders typing efficiency. Our goal is to better understand the user experience of one-handed typing and to explore ways to improve the effectiveness and efficiency of typing on mobile devices with one hand. In this article, we provide a detailed description of our study along with our initial design for one-handed typing. We employed a detailed survey and conducted contextual inquiries to study the potential difficulties and motivation of one-handed typing on mobile devices and developed 6 user requirements. We then established a paper prototype that contains key features such as adjusted key width, and revised punctuation keyboard. Afterward, we evaluated our paper prototype via simplified user testing and heuristic evaluation and evaluated the results against our user requirements. The results show that our design fails to reduce recognition load. Based on the feedback, we implemented our design into a high-fidelity prototype and evaluated it with 15 participants. The statistical analysis shows that our design decreases the typing time when using our design keyboard, in comparison to the baseline design. The conclusion from this work paves the way for developing one-handed typing support features to increase typing efficiency.

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

Additional Key Words and Phrases: text entry, one-handed, typing, mobile phones, textual entry

1 INTRODUCTION

The history of one-handed typing is closely tied to the design of keyboard layouts. The QWERTY keyboard we are familiar with and the oldest was designed with one-handed typing in mind. In the QWERTY layout, more English words can be spelled using only the left hand than using only the right hand. Specifically, more than 3,000 English words can be typed with only the left hand, while only more than 300 words can be typed with only the right hand. The three most common letters in English, 'E', 'T', and 'A', can all be typed with the left hand. [2] While this is detrimental to right-handed people, it also lends credence to the effectiveness of one-handed typing. In 1936, Dvorak improved the QWERTY keyboard and created the Dvorak Keyboard (also called American Simplified Keyboard, or simply Simplified Keyboard), and Dvorak was specially designed for the efficiency of one-handed typing. [3] In 1960, Dvorak designed separate left- and right-handed Dvorak layouts for one-handed typing. The two keyboards are generally mirror images of each other, except for some differences in some uncommon keys, which are suitable for left-handed people and right-handed people. In the design, he tried to minimize the need to move the hand from side to side (lateral movement), as well as reduce the movement of the fingers. In the past twenty years, there has been new development in the keyboard for one-handed typing. Edgar Matias et al [6]invented the Mirrored Keyboard. The idea is to only use one hand (preferably the left one) and type the right-hand letters by holding a key which acts as a modifier key. The layout is mirrored, so the use of the muscle memory of the other hand is possible , which greatly reduces the amount of time needed to learn the layout, if the person previously used both hands to type. This represents people's continuous pursuit of the combination of ergonomics and typing efficiency.

After entering the 21st century, the rapid development of informatization and the increasingly popular electronic products have enabled people to use virtual keyboards on mobile devices in

*All authors contributed equally to this research.

more scenarios than traditional physical keyboards.[10] As mobile devices are more integrated into people's life scenes, people can not only sit at a computer desk and type on a heavy keyboard, but can also conveniently type in coffee shops, classrooms, or on the bus. In different situations, people often free up the other hand to do other things and use only one hand to conveniently complete the typing action. This makes the demand for one-handed typing greatly increased in today's era. [7]

People's pursuit of one-handed typing efficiency piqued our interest and we set our promise as uncovering the challenges that arise with one-handed typing in order to enhance the user experience when typing with a single hand on mobile devices. After preliminary research, we discovered that besides the keyboard layout, other keyboard features like autocomplete/autocorrect, and key sizes, as well as the device's operating system, can also affect user's experience with one-handed text entry. To gain deeper insights into users' experiences with one-handed operation, we meticulously devised a survey and distributed to the public. Our survey of the area indicated that many participants do have difficulties using one-handed typing and are not satisfied with current keyboard design (see Section 3). From the survey results, we were also able to narrow down a preliminary focus to be, "mobile text entry with the right hand preferably on an iPhone, specifically regarding fixing typos, switching between numbers/letters, and switching languages".

In the second assignment, we further explored this focus and the context of use in order to ultimately identify the user requirements. To achieve this, we conducted contextual inquiries, and used the field data collected to identify different sequences of tasks users perform to achieve their goal via a sequence diagram, the flow of information via different artifacts connecting different stakeholders via a flow diagram, and most importantly the potential breakdowns in those diagrams and the affinity diagram. Taken all together, we gained a better understanding of the context of use and identified the user requirements based on the findings. In short, we established that the existing keyboard layout presents difficulties for one-handed text entry, leading us to create user requirements for more intuitive and efficient one-handed operations.

At the current stage, we wanted to address the gaps and breakdowns (user requirements) in the context of use we identified previously. To do so, we had to design potential solutions based on the 6 specific user requirements identified previously. Firstly we designed our primary persona Jack with detailed demographic and behaviorals with respect to the user requirements. Jack has been the central focus throughout this assignment, guiding our design decisions to ensure they align with real-world user scenarios. Throughout the design process, we used a rigorous design critique process and carefully combined various team members' ideas to satisfy Jack's needs. Notably, our approach to improve autocorrect, shift keyboard size to accommodate one-handed use, novel design of convenient punctuation keys have been pivotal in resolving Jack's challenges in one-handed typing. Then we employed sketches to show these new features and storyboards to detail describe the context Jack will use our features. Finally, we combined the sketches and storyboards to the low-fidelity paper prototype. The transition to prototype required us to discuss design critique and make critical adjustments to improve user experience. We also used the Wizard of Oz testing approach to iteratively make informed design and prioritize user needs. In short, based on the specific user requirements, we demonstrate our initial design through low-fidelity paper prototypes to show possible solutions to improve one-handed typing. The details are in Section 6.

After building our paper prototype, we moved to the interactive systems evaluation section. We firstly revised the wording of user requirements to improve its testability and objectivity. Then we separately conducted heuristic evaluation with usability experts to identify usability issues with respect to the 10 usability heuristics. Our results show that we still have usability problems mainly in visibility of system status, user control and freedom, and recognition load. We also conducted simplified user testing with stakeholders to test the features and usability of paper prototype against our user requirements. In this task we encouraged the participants to 'think aloud' while doing the tasks and we acted as observers to watch for errors and confusions. The difficulties are to elaborately design our subtasks to reflect user requirements, as well as letting the participants to perform 'think aloud' during the test. Our results show that the participants still make errors and get confused when completing several subtasks, especially with the discoverability issues. We are going to fix those main issues in the next iteration. In short, we evaluated our paper prototype using simplified user testing and heuristics evaluation to get valuable feedback to improve the existing prototype. The details are specified in Section 7.

Based on the evaluation part we conducted in the previous section, we decided to improve our prototype and make a high-fidelity prototype. We first revised the original design functionality with user requirement 5, then we utilized Figma to build our high-fidelity prototype. One of the obstacles was to implement the interaction in Figma so we had to pay attention to every detail. After building our high-fidelity prototype, we evaluated it with 15 participants by conducting several tasks and collecting the completion time for each task. The difficulty was setting proper measurement metrics for those quantitative data. We incorporated hypothesis testing and the Wilcoxon test to do the data analysis, and the result showed that our keyboard design decreased the typing time compared to the original keyboard. In short, we created the high-fidelity prototype and did statistical analysis to evaluate our design in a quantitative way. The details are in Sections 8 and 9.

2 RELATED WORK

Lifestyle and Health Impact:

One-handed textual entry on mobile phones has been studied extensively by HCI researchers. This area of study is driven by a wide variety of use cases, ranging from improving typing accessibility for those with disabilities to the increasing prevalence of texting while walking. We carry mobile phones around with us all the time, and as a result, interacting with it while on-the-go has become a daily routine [1]. According to a recent study in 2022, 26% of people always use a mobile phone while walking on the street, 43% sometimes do, 28% occasionally do, and only 3% do not [4]. While texting on-the-go, individuals must multi-task. They must not only maintain awareness of their surroundings but also be focused on interacting with the mobile device that is in motion. As a result, the combination of multitasking and interacting with a often shaken device, impairs the user's typing performance and accuracy significantly [1].

One common breakdown discussed in the contextual inquiries was the physical discomfort of texting with one hand. This motivated the team to look into the effects of mobile phone use on an individual's health. We found a decade-long study that monitored the effects of mobile phone use on people's physical health. From 2011 to 2020, the study reported that there have been phone-related injuries ranging from 3,389 to 7320 annually. These injuries included wrists and hand pain, ruptured tendons, and enlarged median nerves. The age group most affected were between the ages of 11 to 30, making up 48.6% of the injuries [9]. One common reason for these injuries were due to overuse of the person's thumb. People who frequently text with one hand do not have the flexibility or the reach to access keys on the other side of the hand. Thus, the users will have to over-strain their thumbs and wrists which puts them at risk for the injuries mentioned above.

Proposed Solutions:

Past research has been done studying how different types of layouts can be used to optimize typing accuracy and speed for one-handed textual entry for mobile devices. Some of the existing one-handed keyboard layouts include curved keyboards as well as a user-adaptable standard QWERTY keyboard [11]. The CAK (Circularly Arced Keyboard), as shown in Figure 1, equally divides each key by 90 degrees into a quarter-circle layout [5]. It is intended to be operated with a single thumb and training time to become familiar with the CAK layout is minimal. The idea behind it is to allow the user's thumb to be able to reach every single key without having to reposition

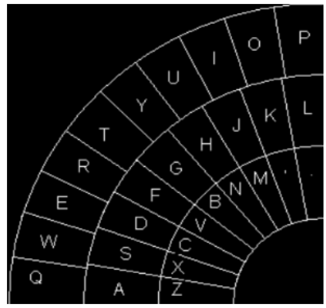

Fig. 1. CAK Keyboard Layout [11]

the phone. However, studies have shown that the improvement to speed and accuracy is not that significant on top of the added learning curve to being familiar with the layout [5].

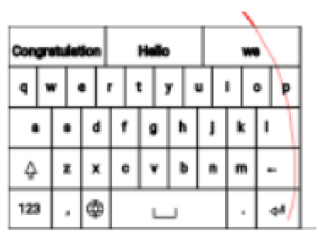

Fig. 2. User-adaptable keyboard [11]

The user-adaptable keyboard layout allows the user to resize the standard QWERTY layout to their desired size as well as shift it to the left or right side of the screen, depending on their typing hand. This layout has a much lower learning curve and is significantly better received by users [11].

There have also been existing studies and concepts that focus more on one-handed typing on physical keyboards instead of virtual keyboards on mobile devices. One of these existing technologies is the Half-QWERTY keyboard. The HALF-QWERTY keyboard has half the keys of a standard keyboard. To type the characters that are not present on the half keyboard, the user must press the spacebar and the mirror half of the keyboard will be mapped onto the keys. Studies show that there is an increased error rate as well as decreased speed [6].

HCI researchers have also completely redesigned the keyboard from the standard QWERTY to specifically support one handed text entry. One example is the 5-key chording device. As the name suggests, this new keyboard only consists of 5 buttons. The 31 different combinations produced from the 5 buttons allow mappings for each 26 letters, backspace, space, comma, period and enter to a combination. The researchers conducted experiments to test the learnability and usability of this new "keyboard." While the participants made a significant amount of mistakes in their first attempts, the average number of errors decreased exponentially with each subsequent round. Furthermore, similar trends occurred with respect to speed. The wpm (words per minute) increased at a constant rate from 4.2 wpm to 15.2 wpm [8].

Although there are already existing technologies and keyboard layouts to address the need of one-handed typing on mobile phones, most of these existing technologies have a steep learning curve and reduce the accuracy and speed of typing. In addition, most HCI research is focused on

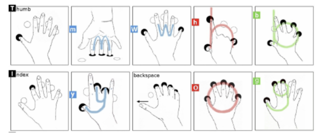

Fig. 3. 5-key keyboard [8]

the English alphabet and does not look closely into other writing systems, such as Chinese or the Cyrillic alphabet. As a result, the effects of one-handed textual entry for the majority of the human population is severely understudied. Furthermore, there has been no research that we are aware of that studies the effects of switching keyboard languages on speed and accuracy. This is important as bilingualism among the general population is on the increase. Lastly, most HCI research is also focused on letter entry and does not focus on entry of symbols such as % or !, and emojis. Entries of these symbols and emojis typically require the user to switch to a different keyboard, which may not be standardized over all keyboards interfaces, as opposed to the standard QWERTY layout.

3 ESTABLISHING FOCUS: INITIAL SURVEY

The purpose of the survey was to determine whether one-handed text entry on a mobile device could be a valid focus and if so, further narrow it down by examining its potential difficulties and issues. Otherwise, we hoped to perhaps broaden or even shift the focus by including questions about other areas within text entry on a mobile device beyond just one-handed typing.

3.1 Method

After deciding the initial focus of the survey to be one-handed text entry on mobile devices, we defined the survey protocol and designed the questionnaire.

3.1.1 Define inclusion/exclusion criteria. The inclusion and exclusion criteria for participants was intentionally lenient, since this was only the initial stage of user-centered design and we aimed to understand the context of use for as many stakeholders as possible. Thus, we defined the protocol to include all participants over the age of 18 and have previously used a smartphone, and exclude those otherwise.

3.1.2 Design questions and question paths. The survey was created using Google Forms and initially consisted of the following six sections: (1) consent, (2) demographics, (3) whether the participant had previously typed on their phone with one hand, (4) one-handed mobile text entry questions, (5) general text entry questions, (6) end of survey. Section 1 first described the purpose of the study, guaranteed the participant's anonymity, and then asked one question to ensure the participant was over 18 and they have previously used a smartphone (meeting the inclusion/exclusion criteria). If the participant answered "no", the survey would immediately end; otherwise, the survey would proceed to the next section. Section 2 consisted of questions regarding participants' demographics information. This information was important as it would inform us about to which group of people we could generalize our results to, and about the certain human factors the participants may exhibit. Other than the general demographics such as age, gender, etc., we also asked questions about their smartphone usage, such as their brands. Questions like these allowed us to gain a better understanding of the features the OS may provide to the participants. Section 3 consisted of one question asking whether the participant had prior experience with typing on their phone with one hand. If so, they would proceed to section 4 (i.e., the survey proceeds as normal) which asked more questions specifically focusing on the context of use for one-handed text entry on smartphones and its potential difficulties; otherwise, they would skip section 4 and be redirected to section 5 which asked questions regarding the broader context of use for text entry on mobile devices beyond just typing with one hand. Section 6 contained a message thanking the participants for completing the survey. When creating the questions, we tried to avoid common pitfalls of survey design as much as possible. For example, to avoid forcing answers, every question either provided a "neutral" or "prefer not to answer" option, or did not require an answer (i.e. participant could choose not to answer the question and still proceed in the survey). The complete initial survey is located at Appendix B.1.

To ensure the quality of the survey before distributing to a large sample, a pilot test was conducted on the initial survey with a total of 9 participants (see Section 3.2.1 for specific tasks/procedures of the pilot study). We then addressed the issues discovered from the pilot study by collectively examining the entire survey question-by-question and after thorough discussion, making necessary changes to either the question, the options, and/or the order of questions to resolve each issue. For example, one common issue that was found in many questions and answer choices was using terminology that was unfamiliar to the participants. Thus, we rephrased the questions and answer options with layman's terms or use examples to show their meanings. Another issue was that some participants found the placement of demographics question at the beginning of the survey to be "weird", since after reading the purpose of the survey in section 1, they expected the questions to be about smartphone text entries, instead of their gender, race, etc. Thus, we moved the demographics section to be the last section of the survey to avoid any confusions. Furthermore, we addressed some newly-discovered pitfalls that we failed to take into account when designing the initial survey, such as leading questions and unbalanced answer option. We also removed some questions that were found to be repetitive or unrelated to the topic of text entry.

After addressing all the issues discovered from the pilot study, the final survey consisted of the following sections: (1) consent, (2) general smartphone usage questions, (3) one-handed mobile text entry questions, (4) general text entry questions, (5) demographics, (6) end of survey. All sections except section 2 corresponded to an existing section from the initial survey, with appropriate changes applied to them. The current section 2 consisted of the question for determining whether the participant had prior experience with one-handed typing on a smartphone (with the same section skip/redirection logic as before), and two more questions regarding smartphone usage patterns imported from other sections. Additionally, the current section 1 contained a more detailed description of the purpose of the survey, while section 6 contained an extra request to distribute the survey to more people (see Section 3.2.2 and 3.3 for more information). The complete final survey can be found in Appendix B.2.

3.1.3 Specify analysis methods. We planned to use exploratory data analysis (EDA) on the data we collect, since we mainly wanted to observe general patterns.

3.2 Tasks and Procedures

3.2.1 Pilot study. The pilot study was used to ensure the quality control of the survey. Each team member individually conducted the pilot study with 1 to 2 acquaintances (totalling 9 participants) either online or in-person. During the study, each participant was first informed that the purpose of this study was to gather data that could improve the quality of the survey, and no personal identifiable information would be recorded. We obtained the participant's consent via both verbal confirmation and selecting "yes" to the question in section 1 (consent) of the survey. They were then instructed to fill out the survey on any electronic device, while constantly narrating their thoughts and confusions out loud. Each participant's issues regarding the initial survey were recorded during the pilot session. On average, the study took approximately 10 minutes to complete. We thanked the participants after completion, and no monetary compensation was provided.

3.2.2 Final survey. After refinement of our questionnaire, we used a URL link generated by Google Forms to distribute the survey online (see Section 3.3 for sampling method). Upon clicking on the link, each person would be informed of the purpose of the survey and assured with the anonymity of their responses. Then, they would choose to either give their consent to participate and assert that they meet the inclusion/exclusion criteria by selecting "yes" to the very first question, or select "no" otherwise; the former would continue to fill out the rest of the survey (on any electronic device), whereas the latter would be exited out of the survey. We expected the final survey to take about 7 minutes to complete, taking the average time of the pilot studies and the total number of questions we had into consideration. After completion, each participant received a thank you message and a request to forward the survey to more individuals, a choice left to the discretion of the participant. After data collection period ended, we examined the responses to ensure that the data was completely anonymous and no personal identifiable information was recorded.

3.3 Participants

Age group	Number of participants
18-25	26
26-40	1
41-60	0
61-80	0
80+	0

Table 1. Age group of participants.
96.3% between ages 18-25, 3.7% between ages 26-40

Race and ethnicity	Number of participants
White	0
Latinx or Latin American	0
Black or African American	0
American Indian or Alaska Native	0
Asian	21
South Asian	1
Middle Eastern or North African	0
Native Hawaiian or Pacific Islander	0
Multiracial (specifically, White and Asian)	3
Prefer not to answer	2
Other	0

Table 2. Race and ethnicity of participants
77.8% Asian, 11.1% White and Asian, 3.7% South Asian, 7.4% prefer not to say

A convenient sample was selected using a snowball method. Each team member sent out the final survey online as an URL link to all of their acquaintances who met the participants' inclusion criteria defined in the survey protocol (see Section 3.1.1). Those who received the survey were

Gender identity	Number of participants
Woman	7
Man	18
Non-binary / Gender fluid	1
Indigenous or other cultural minority gender (e.g., two-spirit)	0
Prefer not to answer	1
Other	0

Table 3. Gender identity of participants
== 66.7% man, 25.9% woman, 3.7% non-binary/gender fluid, 3.7% prefer not to answer

Employment status	Number of participants
Employed full-time	9
Employed part-time	3
Self-employed	0
Unemployed (including student)	15
Retired	0
Other	0

Table 4. Employment status of participants
55.6% unemployed (including student), 33.3% employed full-time, 11.1% employed part-time

Number of languages proficient	Number of participants
1	6
2	15
3	6
4+	0

Table 5. Language proficiency of participants
55.6% proficient in 2 languages, 16.2% proficient in 1 language, 16.2% proficient in 3 languages

Disability status	Number of participants
Yes	1
No	24
I don't know	2
Prefer not to say	0

Table 6. Disability status of participants
88.9% do not have a disability, 7.4% do not know, 3.7% have a disability

informed to complete it virtually at their convenience and then forward it to 2 or 3 acquaintances who may be potential stakeholders. No monetary incentives were given for the completion of the survey. Using the snowball method, we hoped to gather as many responses as possible. After three days of data collection, a total of 27 participants completed the survey (96.3% between ages 18-25; 77.8% Asian, 11.1% White and Asian, 11.1% other; 66.7% Man; 55.6% unemployed; 71.8% multilingual; 3.7% have a disability; see Table 1- 6 for complete demographics information).

3.4 Results

The survey data were exported from Google Form and the corresponding plots were made for each question (other than the demographics questions) using the Chart feature in Google Sheets. The questions and their results are listed in the order which they appeared in the survey. The results of the questions are analyzed either individually, written immediately after a figure, or in a group, written after a contiguous block of figures. See Appendix B.3 for the complete responses (anonymized and de-identified) from all participants.

3.4.1 General smartphone usage questions. (corresponding to survey's section 2)

Question 1: On average, how long do you use your smartphone per day?

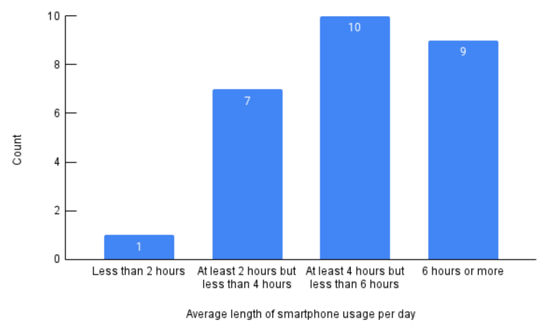

Fig. 4. Count of participants' average lengths of smartphone usage per day

Participants are roughly evenly distributed among using their smartphone for at least 2 hours but less than 4 hours (7 out of 27, 25.9%), at least 4 hours but less than 6 hours (10 out of 27, 37.0%), or 6 hours or more (9 out of 27, 33.3%) per day. Only 1 participant (3.7%) uses their phone for less than 2 hours per day.

Question 2: What brand is your smartphone?

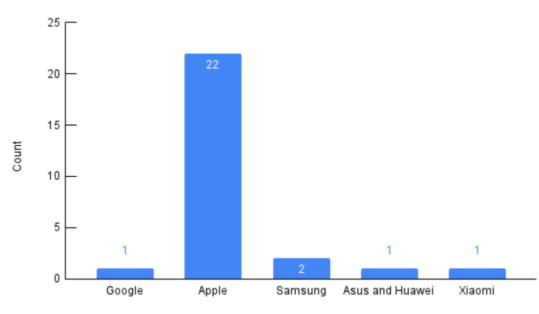

Fig. 5. Count of participants' smartphone brands

An overwhelming majority of participants (22 out of 27, 81.5%) have Apple as their smartphone. The remaining participants were roughly evenly distributed among using Google, Samsung, Xiaomi, and Asus/Huawei as their smartphone. This implies that the majority of the participants would be using Apple's IOS operating system as well as its associated features and are subjected to IOS's user experience designs.

Question 3: Have you ever entered text (typed on the virtual keyboard) on your smartphone with one hand?

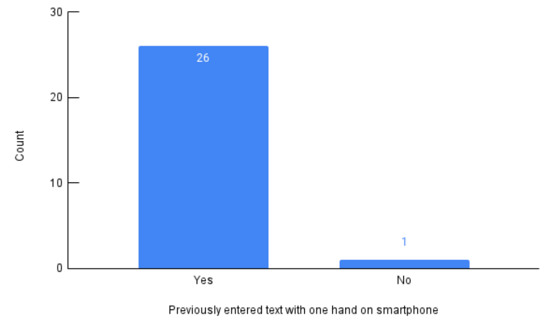

Fig. 6. Count of participants' average lengths of smartphone usage per day

Almost all participants (26 out of 27, 96.3%) have previously entered text with one hand on a smartphone. The one person who has never entered text with one hand skipped the next section of the survey, therefore questions 4 to 17 in the following section only has 26 responses.

3.4.2 One-handed mobile text entry questions. (corresponding to survey's section 3)

Question 4: How often do you enter text on your phone with one hand?

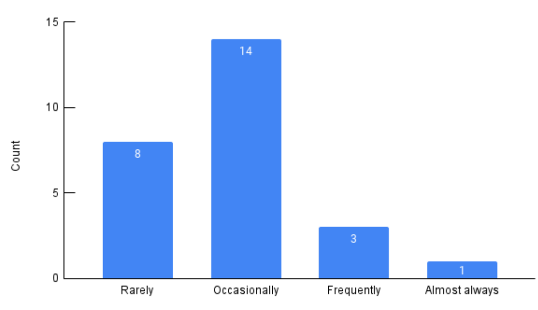

Fig. 7. Count of different frequencies of one-handed text entry on a smartphone performed by participants

Out of the participants who have previously entered text on their smartphone with one hand, a majority of them (14 out of 26, 53.8%) occasionally do so, whereas some of them (8 out of 26, 30.8%) rarely do so. Though these adjectives are not too impactful, the results of this and the previous question still demonstrate the existence of one-handed text entry on smartphones as a valid interaction between the user and the technology.

Question 5: Do you usually enter text with your phone in a vertical or horizontal position?

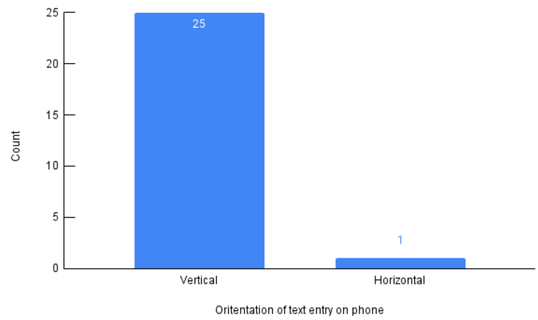

Fig. 8. Count of different orientations of text entry on phone used by participants

Out of the participants who have previously entered text on their smartphone with one hand, almost all of them (25 out of 26, 96.2%) tend to enter text vertically.

Question 6: How comfortable are you with typing on your phone with one hand?

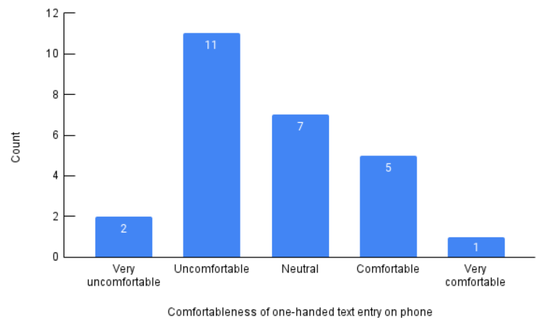

Fig. 9. Count of different levels of comfortableness experienced by participants when performing one-handed text entry

Question 7: What is the level of difficulty you find when entering text with one hand compared to entering text with both hands?

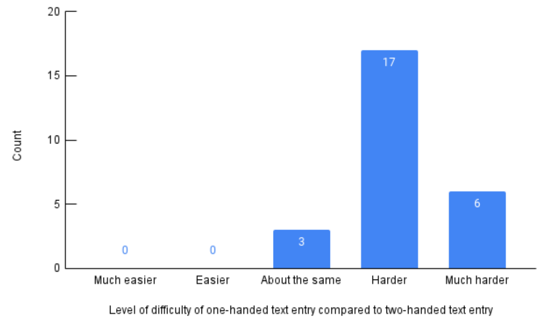

Fig. 10. Count of different levels of difficulty experienced by participants when entering text with one hand vs. two hands

Question 8: What is your typing speed when entering text with one hand compared to entering text with both hands?

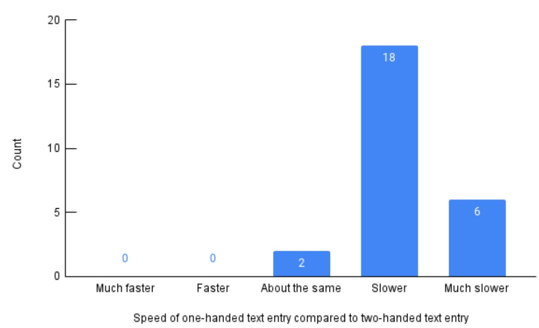

Fig. 11. Count of different typing speeds exhibited by participants when entering text with one hand vs. two hands

Questions 6, 7, and 8 attempt to measure the usability of existing smartphones' keyboards when performing one-handed text entry across three facets: comfortableness, difficulty, and efficiency. As shown in Figure 9, exactly half of participants (13 out of 26) find typing on their phone with one hand to be uncomfortable to very uncomfortable, while the other half of participants either have neutral opinions (7 out of 26, 26.9%) or find it to be comfortable to very comfortable (6 out of 26, 23.1%). Moreover, an overwhelming majority of participants find it harder (23 out of 26, 88.5%) and slower (24 out of 26, 92.3%) to enter text with one hand compared to entering text with both hands, as shown in Figure 10 and 11, respectively. None of the participants find it easier or faster to enter text with one hand compared to two hands. This indicates that there are flaws in the current smartphones' keyboards for one-handed text entries.

Question 9: How often do you make mistakes/typos when entering text with one hand compared to entering text with both hands?

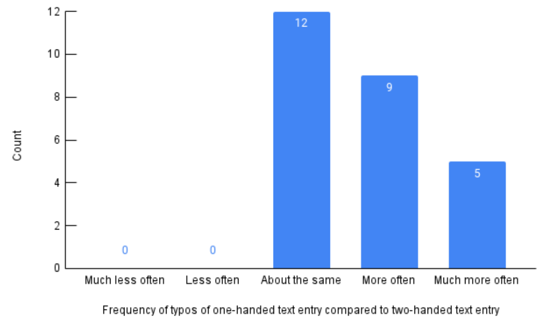

Fig. 12. Count of different frequencies of typos experienced by participants when entering text with one hand vs. two hands

Question 10: What is the level of difficulty you find when fixing mistakes/typos with one hand compared to fixing mistakes/typos with both hands?

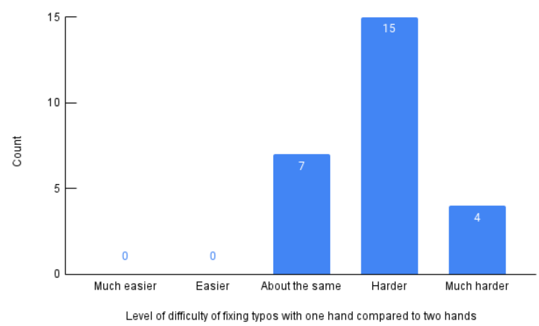

Fig. 13. Count of different levels of difficulty experienced by participants when fixing typos with one hand vs. two hands

Question 11: What is your opinion on the autocorrect feature of your phone's keyboard?

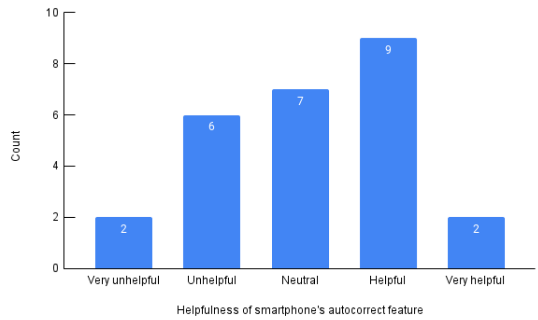

Fig. 14. Count of different opinions expressed by participants on the helpfulness of smartphone's autocorrect feature

Questions 9, 10 and 11 attempt to further explore the usability of the existing smartphone's keyboard for one-handed text entry, specifically regarding the ability to avoid and correct mistakes (typos). As shown in Figure 12, the majority of participants (14 out of 26, 53.8%) tend to make typos more often or much more often when entering text with one hand compared to with both hands, and the remaining participants (12 out of 26, 46.2%) make typos with one hand as often as with both hands. Similarly, the vast majority of participants (19 out 26, 73.1%) believe that it is more difficult to correct typos on their phone with one hand than with both hands, while the remaining (7 out of 26, 26.9%) believe that it is about the same level of difficulty, as shown in Figure 13. This indicates that the existing keyboard on a smartphone may be more error-prone and harder to correct mistakes when typing with one hand compared to with both hands.

Contrary to the prior two questions' data with uneven distributions that skew towards the right, regarding the helpfulness of the autocorrect feature, the opinions are roughly evenly split on both sides. As shown in Figure **??**, 11 out of 26 participants (42.3%) think that autocorrect is either helpful or very helpful, 8 participants (30.8%) think it is unhelpful or very unhelpful, whereas the remaining 7 participants (26.9%) have neutral opinions about autocorrect. Though there are more participants with positive opinions on the autocorrect feature, it is difficult to conclude anything as this difference is marginal and the sample is quite small. Thus, it is safer to say that there appears to be mixed opinions on the autocorrect feature of a smartphone; additional questions and corresponding statistical analysis may be needed, for instance, to test which aspects of the autocorrect feature people think are helpful and which aspects are not.

Question 12: On a scale of 1 (easy to use) to 5 (hard to use), what is your opinion on your phone's keyboard layout when typing single-handed?

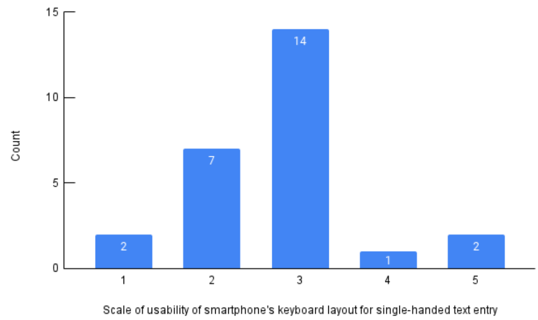

Fig. 15. Count of different levels of usability of keyboard layout experienced by participants when performing one-handed text entry

Question 13: What do you think of the size of the keys on your smartphone's (default) virtual keyboard?

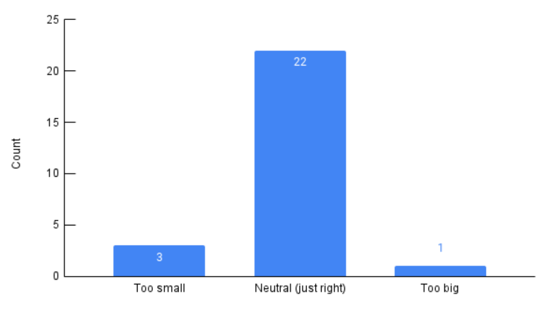

Fig. 16. Count of different opinions on the size of the keyboard expressed by participants

Questions 12 and 13 aim to test whether layout and size play a role in the usability of a smartphone's keyboard for one-handed text entry. Figure 15 shows that a majority of participants (14 out of 26, 53.8%) have neutral opinions about their current text entry keyboard layout, while the general view is that the data is somewhat skewed towards the 'easier' side, with 7 out of 26 (34.6%) participants believing the layout is easy or very easy to use. This indicates that most people are already satisfied with the existing layout of a smartphone's keyboard. Similarly, almost all participants (22 out of 26, 84.6%) think that the size of keys on their smartphone's keyboard is just about right as shown in Figure 16. Together, the data of these two questions imply that key size and general layout are likely not major factors that contribute to the difficulty of single-hand text entry on mobile devices. These results are in face of the finding that the participants make mistakes typing text more often one-handed, if not just as often as with both hands.

Question 14: What is the level of difficulty you find when switching between numbers and letters using one hand compared to that of two hands?

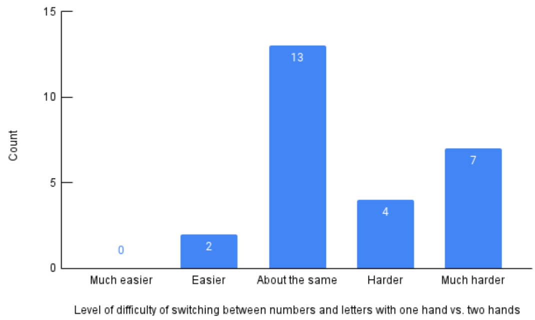

Fig. 17. Count of different levels of difficulty experienced by participants when switching between numbers and letters using one hand vs. two hands

Question 15: How often do you switch languages (including emojis) on your keyboard?

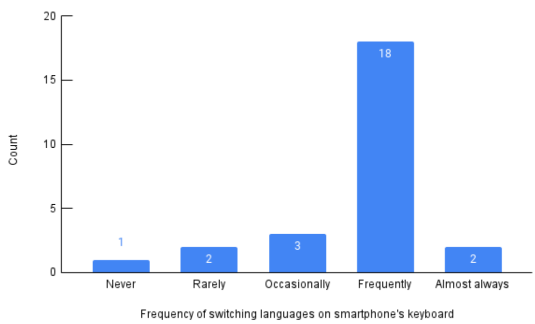

Fig. 18. Count of different frequencies of switching keyboard languages experienced by participants

Question 16: What is the level of difficulty you find when switching languages using one hand compared to that of two hands?

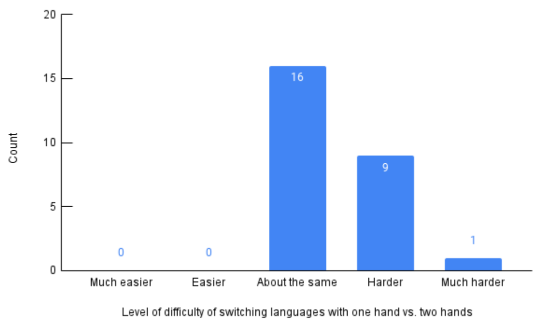

Fig. 19. Count of different levels of difficulty experienced participants when switching keyboard languages with one hand vs. two hands

Questions 14, 15, and 16 aim to test whether keyboard switching plays a role in the usability of a smartphone's keyboard for one-handed text entry. In Figure 17, 11 out of 26 participants (42.3%) think it is hard or much harder to input text when they need to switch between alphabetic and numeric keyboard with one hand. Although 13 participants (0.5%) have neutral opinions on this topic, compared to keyboard layout and key size, keyboard switching does invoke a relatively more negative feedback pertaining to text entry with one hand. A related finding is about switching between language keyboards. It is a similar operation to switching to the numeric keyboard. As is shown in Figure 18 and 19, a majority of participants switch between different languages frequently and think the difficulty to do so with one hand is about the same as with both hands. Yet from Figure 19, we can also observe that no participant thinks switching with one hand is easy; 10 out of 26 (38.5%) believe it is hard or very hard to perform such a task with just one hand. Together, the results of these 3 questions indicate that switching between numbers and letters, and switching languages could be contributing factors towards difficulty of single-hand text entry.

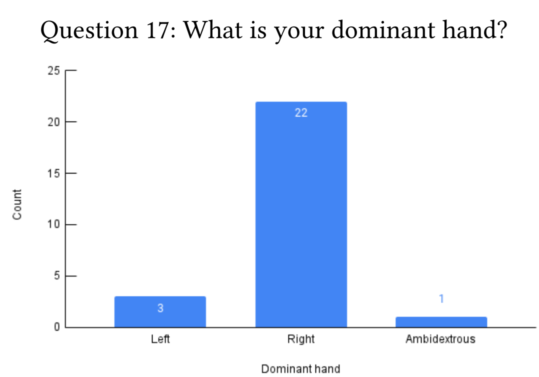

Fig. 20. Count of different dominant hands of participants

Of all 26 participants, 22 (84.6%) have their right hand as the dominant hand, only 3 (11.5%) participants' dominant hand is their left hand, and 1 (3.8%) participant is ambidextrous. An effective keyboard design for single-handed text entry should take the user's dominant hand into consideration and should take the benefit of the doubt that the design for right-hand may not mirror exactly to the design for left-hand. Thus, it may be better to narrow the focus to text-entry with the right hand, and first design specifically for users whose dominant hand is the right hand. In future iterations on the design, designers may take accessibility into consideration and design for those whose dominant hand is not their right hand.

3.4.3 *General text entry questions.* (Corresponding to survey's section 4)

Question 18: How comfortable are you typing on your phone without looking at the keyboard?

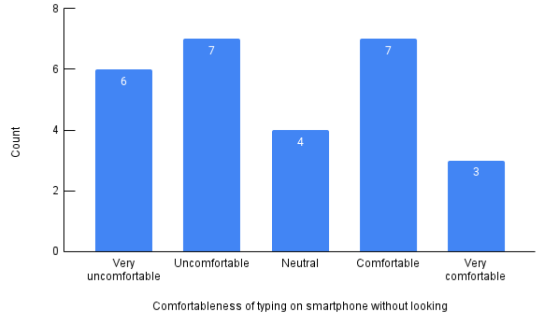

Fig. 21. Count of different level of comfortableness experienced by participants when typing on their phone's keyboard without looking

Figure 21 shows that participants varied a lot in their ability of touch typing (typing without looking at the keyboard), with 10 participants (37.0%) being comfortable or very comfortable, 13 (48.1%) being uncomfortable or very uncomfortable, and 4 (14.8%) having neutral opinions.

Question 19: What is your opinion on the autocomplete (predictive text) feature of your phone's keyboard?

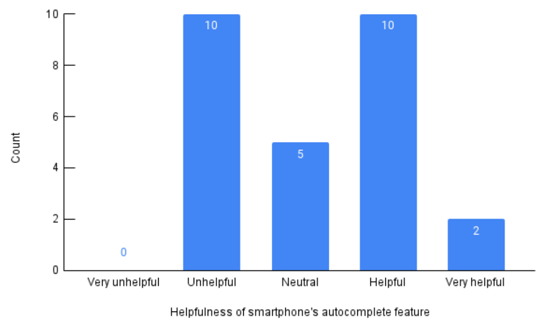

Fig. 22. Count of different opinions expressed by participants on the helpfulness of smartphone's autocomplete feature

In Figure 22, almost half of participants find the autocomplete feature to be at least helpful during textual input. However, the majority have neutral opinions at best, and find it unhelpful at worst.

Question 20: What is your opinion on customized shortcuts for typing? (e.g., type 'omw', turns into 'on my way')

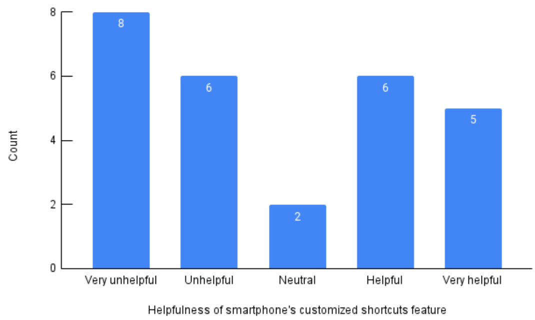

Fig. 23. Count of different opinions expressed by participants on the helpfulness of smartphone's customized keyboard shortcuts

Figure 23 shows that participants are opinionated and divisive about customized shortcuts.

Question 21: Have you ever tried other 3rd party keyboard apps on your phone?

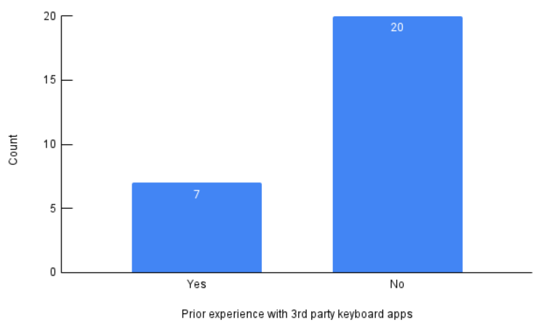

Fig. 24. Count of whether participants have used third-party keyboards

Figure 24 shows that a majority of participants have never used a third-party keyboard.

Question 22: What is the level of difficulty of entering text on a smartphone compared to that of a computer?

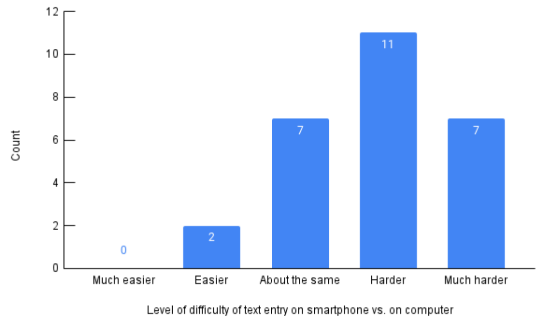

Fig. 25. Count of different levels of difficulty of text entry experienced by participants on a smartphone vs. on a computer

Figure 25 shows that the vast majority of participants (25) think that text entry on the phone is as hard or harder than that on computer, and only 2 participants think it is somewhat easier.

Question 23: What are some other text entry methods that you have tried/used on your smartphone? Select all that apply

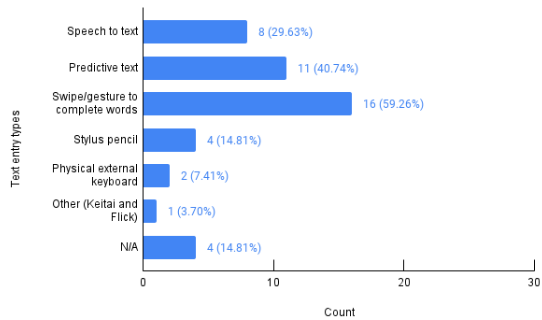

Fig. 26. Count of different text entry types used by participants

Figure 26 shows that alternative forms of text entries on mobile devices are still quite novel and unfamiliar to the participants, with swipe-input (gesture) having the most users (16 out of 27, 59.26%) in our study.

4 UNDERSTANDING CONTEXT OF USE: CONTEXTUAL INQUIRY

The purpose of this stage in the user-centered design was to come up with user requirements based on a better understanding of the context of use, specifically in identifying the tasks performed by the stakeholders and the potential breakdowns within them.

4.1 Method

To achieve the purpose of the study, a contextual inquiry (contextual interview), an excellent method for collecting qualitative field data, was first conducted by each team member (more detail in 4.2). The main benefit of this method was that the interviews took place in the context of use. This allowed not only the interviewee (i.e., participant or user) to better recall the memories of their interaction with the technology but also the interviewer to better understand the environment in which the interaction took place, thus successfully collecting field data grounded in real user experiences, instead of laboratory data that may not apply to real-life settings.

After the contextual inquiry was completed, each team member individually interpreted and analyzed their results by extracting different information from the interpretation notes taken during their interview to create sequence diagrams and a flow diagram. A total of seven sequence diagrams and five flow diagrams were created. Each sequence diagram documented the sequence of steps which a participant took to accomplish a goal, allowing us to inspect the intentions behind what each participant did and the potential breakdowns in specific steps. On the other hand, each flow diagram revealed the flow of information and artifacts between different stakeholders mentioned by a participant, and different technologies used by a participant to complete their tasks.

The seven individual sequence diagrams were then consolidated into one consolidated sequence diagram by considering different participants as just one single user identity and merging (taking the union of) the diagrams in which the user did similar sequences of tasks or had similar goals. The five individual flow diagrams were also consolidated into one consolidated flow diagram using a similar strategy. Lastly, all of the interpretation notes from the five contextual inquiries were consolidated into an affinity diagram, revealing the categories and themes that emerged from the interpretations.

All together, the consolidated sequence diagrams, consolidated flow diagram, and affinity diagram illustrated the big picture, giving us a deeper, more complete understanding of the context of use of one-handed text entry on mobile devices. Consequently, the user requirements were created based on the findings and potential breakdowns we identified from these diagrams and interpretation notes.

4.2 Tasks and Procedures

To ensure the quality and consistency of the contextual inquiry conducted by different team members, we first created a protocol together as a group, specifying the interview's focus, duration, and initial questions, the inclusion/exclusion criteria for the participants, the information and tasks given to the participants, and the process of obtaining informed consent. As such, the core aspects of the contextual inquiry remained the same between different team members. We then split up to conduct the contextual inquiry individually, each with one participant. The following paragraphs describe the tasks and procedures we used for all participants, in general.

The participant first received a message from the investigator online, inviting them to participate in an interview; the message contained the interview's focus, one-handed mobile text entry, estimated duration, one hour, and that it had to be audio-recorded and take place at the context of use. After obtaining the participant's initial confirmation that they had prior experience with one-handed mobile text entry and were willing to do the interview, the participant and the investigator then met in person. To save transportation time, we met directly at the location in which the participant frequently typed on their smartphone with one hand (i.e., the context of use).

Before starting the interview, the participant was once again informed about the purpose, to better understand the context of use of one-handed text entry on mobile devices, the tasks they had to perform, to respond to questions asked by the interviewer to the best of their ability, and the fact that they were "the expert" in this interview. They were assured that their participation was fully voluntary as they may quit the interview at any time. They were also told that the interview would be recorded for further analysis if needed and the data collected from the interview would be completely anonymous and confidential. The participant then gave their verbal consent to participate in the interview had they agreed to all of the conditions mentioned, which they did.

The contextual inquiry then began. Since our initial focus, being "one-handed text entry on a mobile device, preferably on an iPhone and with the right hand, specifically regarding fixing typos, switching between numbers/letters, and switching languages", was very narrow, we decided to begin by only asking about the participant's experience with one-handed mobile text entry in general. If we were out of questions to ask or if the participant actually mentioned these specific aspects of one-handed text entry (fixing typos, etc.), we would then proceed to ask about them. Thus, the first question presented to the participant was, "Please try to think about one specific instance in which you entered text on your phone with one hand. Would you please take us to the location where it [the interaction] happened?" Since we were already at the desired location, we did not have to move and I proceeded to ask the next question, "Would you please try to describe what happened? For example, what prompted you to enter text with one hand? What were you trying to do with the text entry? How did you do it? etc." All of the remaining questions were based on the participant's response and factors in the context of use.

The interview lasted approximately an hour. Once the interview concluded, we thanked the participant for their time.

4.3 Participants

A total of five participants were conveniently selected, one for each contextual inquiry we conducted individually. Each team member sent out invitations to participate in the interview online to close friends who met the inclusion/exclusion criteria, which was 1) they must be at least 18 years old, and 2) they have prior experience with one-handed text entry on mobile devices. The person who agreed first became the participant. No monetary incentives were provided for the completion of the interview. The demographic information about the participants are listed in Table 7. For the analysis, each participant was assigned with an encoding from U01 to U05 respectively, and they were collectively referred to as "the user" in the remainder of the report.

User encoding	Age	Gender	Race	Employment status	Dominant hand	Phone brand	Multi-lingual	Disability
U01	21	Male	Asian	Student	Right	iPhone	Yes	No
U02	22	Male	Asian	Student	Left	Google Pixel	Yes	No
U03	27	Male	Asian	Student	Right	Samsung	Yes	No
U04	23	Female	Asian	Employed full-time	Right	iPhone	Yes	No
U05	22	Male	Asian	Employed part-time	Right	iPhone	Yes	No

Table 7. Participant demographic information.

4.4 Results

In this section, we discussed the results of our contextual inquiry interviews. We have consolidated sequence diagrams and flow diagrams from 5 different interviews, as well as an affinity diagram that contains our interpretations from all 5 interviews.

4.4.1 *Affinity Diagram.* See Appendix C.6.

User frequently uses their phones with one hand throughout the day. The contexts in which individuals need to use their phones with one hand is usually when multitasking. Individuals are usually doing something before or while needing to type with one hand. Some requires typing with one hand while multitasking on daily tasks. For instance, the user needed to brush their teeth while messaging their friend about a homework assignment (e.g., U05-22). The user could also be walking (e.g., U05-01) or eating (e.g., U02-01). These activities often required users to switch their attention back and forth between the phone and their environment.

Keyboard layout hinders the user's ability to enter text. Almost all participants faced challenges with existing keyboard layouts that hinder their ability to enter text. The user often finds it difficult to reach keys on the far edges or opposite side of the phone display while operating one-handed. This results in constant readjustment to their grip of the phone, which in turn causes accidental clicks on unintended keys, reducing typing speed and efficiency. In addition, the placement of certain keys is difficult to access with only one hand. This is particularly the case for those buttons on the corners of the screen, where the thumb is unable to reach. Moreover, the user often struggled to locate characters like the question mark (e.g., U02-08), which they expected to find at the bottom right key, mirroring a standard physical keyboard layout, but instead, they had to switch to a different punctuations keyboard first in order to enter such character. This mismatched mental models and the need of keyboard switching drastically slowed down the typing efficiency.

Talking about keyboard switching, all punctuations, special symbols (like "#"), and emojis. which are characters the user frequently enters, are all located on different keyboards, being another major issue that was identified. Accessing specific these characters not only requires the user to switch keyboards, but also the switch key is inconveniently placed on the far edge of the phone screen. Most participants reported that they had to use a handful of punctuations frequently, implying that the current design might not be optimized for usability of the majority's needs. Some even skipped punctuation or capitalization entirely because of the difficulties and inconveniences of the keyboard layout (e.g., U03-12, U05-37).

Users usually find some assistive keyboard technologies unhelpful. Almost all participants also used some assistive keyboard technologies like autocorrect and slide-to-text. However, many participants found that the assistive technology is sometimes not effective. In the case of autocorrect, it appears counterproductive when the user is using one hand to enter text while multitasking. The default autocorrect feature on the phone would automatically change the typo word to a word that the algorithm believes is what the user intended to type, unless the user manually select a suggested word displayed at top of the keyboard (U05-09, U05-11). However, more often than not, the autocorrect feature got it wrong (U05-09), and the resulting word would sometimes be completely different from what the user intended, especially when it was an acronym (U05-10). Moreover, the user could not use the autocorrect feature on this word anymore because the word was now considered as "correct" by autocorrect; consequently, the user had to manually delete and retype the entire word (U05-12). Forgetting to manually select from the suggested words was especially common when the user was using one hand and distracted by the surroundings (U05-11), causing the autocorrect to be in fact counterproductive and making the user very annoyed and frustrated (U05-13). In contrast, the user was able to use the autocorrect feature much more effectively and efficiently when typing with both hands (U05-30).

Users make many slips (typos) texting one handed but is not always an issue to complete the task. When typing one handed, users have a higher incidence of making slips and typos compared to typing with both hand (e.g., U05-07). This may be due to many things, but one major reason is that, the user has to make larger jumps between key presses when entering text with one hand compared to entering text with both hands (e.g., U05-29). Moreover, the user often struggles with fixing these typos with one hand, especially when the autocorrect feature is not working (e.g., U05-13, U05-26). When the user becomes so annoyed and frustrated with these typos, they would switch to typing with both hands even if that means stopping the task they are doing with the other hand (e.g., U05-12, U05-27), and sometimes, they would even completely ignore the typos (e.g., U03-02). Although most of the times, the user believe the accuracy of their messages is important, they occasionally disregard typos when sending error free messages is unnecessary, specifically if they believe that the recipient would be able to understand their intentions (e.g., U03-03). In context of performing a Google search with one hand, the user is also aware that platforms such as Google search can still understand their intent, even with typos (e.g., U05-35).

The size and weight of the phone sometimes causes physical discomfort to the user to text with only one hand. The size and weight of the phone has also been attributed to the cause of physical discomfort while texting with one hand. Prolonged use often causes thumb fatigue, and holding the phone in certain positions, such as while lying on the user's back, can lead to arm soreness (e.g., U04-04). In addition users also face challenges with stability while gripping the phone with one hand. The instability when typing with one hand often requires the user to constantly readjust the phone's position in the hand. Some users even had an incident where they felt like dropping their phone due to instability (e.g., U03-07). As a result, to solve this discomfort, many users even switched to typing on both hands for better stability. When typing long texts or when accuracy is needed, users forgo typing with one-hand and switch to two hands, indicating there is a clear preference to minimize typing with one hand.

4.4.2 Consolidated Sequence Diagram. Our consolidated sequence diagram (see Appendix C.4) contains all steps that each participation took to input text with one hand. Although the participants may begin with slightly different contexts (top region of diagram), once they have begun text entry, their procedures begin to fully or partially merge. The participant may encounter certain situations as they enter the main sub-task loop of text entry (center region of diagram), and they would repeat this loop until they have finished their task and jump out of the loop (bottom right corner of diagram). Given our consolidated sequence diagram, the participant usually runs into breakdowns when they: switch to a different keyboard, make a slip (typo) and attempt to correct it, and physically adjust the phone in their hand. The participant would attempt to deal with one of these issues before they begin another new iteration of text entry sub-task and encounter one of the issues again, or not. There is also a case when the participant gives up on entry with one hand and switches to two-hand entry (bottom region of the diagram). In this case the participant would also jump out of the main loop and would go through a different set of procedures before their task is done.

The participants' breakdowns can be generally grouped by the different subtasks in the main loop. During a keyboard switch, the participants experience many issues related to the difficulty of reaching and locating specific keys with only one hand due to the layout of the keyboard. The participant usually makes a typo as some fashion of slip due to muscle memory, proximity of key placements, and awkward hand postures. When the participant attempts to fix a typo, they would encounter issues with assistive features, placement of the delete key. The participant would also have the phone slip out of their hand during input due to larger phone display sizes mandating moving the phone in their hand.

4.4.3 Consolidated Flow Diagram. The consolidated flow diagram (see Appendix C.5) shows the flow of information from the user (participant) to a recipient that is either another person or another service/platform. The information flows from the user, through their phone (artifact), and towards the recipient, and the recipient would also return information back. The information takes form as a text message, an email, or some form of query request. The breakdown takes place during the input of these information that are detailed in our sequence diagram.

5 USER REQUIREMENTS AND FUNCTIONAL CONSTRAINTS (IMPROVED)

The user requirements were originally created based on our understanding of context of use and analysis of contextual inquiry results. Later, we carefully revised each user requirement, attempting to improve its testability, objectivity, and overall wording, based on our new knowledge gained from lectures about user evaluation methods and feedback received from GSI. These improved requirements are listed from Section 5.1 to Section 5.4, separated by the area of breakdown it corresponds to. Then, as a group, we selected three user requirements against which the qualitative user evaluations (specifically the simplified user testing) tested our design; these requirements were selected based on whether they could be feasibly evaluated using qualitative methods and our low-fidelity paper prototype (further discussed in Section 5.5 and Section 7.2.2).

We identified the user requirements based on the general themes that emerged from the affinity diagram and breakdowns discovered in consolidated diagrams, while referencing specific interpretation notes. Thus, the user requirements are grounded in the contextual inquiry results and when considered all together, are comprehensive and cover all relevant aspects (breakdowns) related to the context of use. (The headings in this section each represent an important theme from the affinity diagram that describes/summarizes some potential breakdowns.) We also tried our best to meet the other rules/principles for each user requirement, including testability, objectivity, and may not imply a solution. However, we realized that there would always be a tradeoff between the rules/principles that a user requirement must satisfy, especially between the requirement's objectivity and whether the requirement implies a solution. Being more specific would increase objectivity, but would also increase the likelihood of implying a solution. On the other hand, being more abstract would less likely imply a solution, but would also be less objective. Thus, we tried our best to balance these two rules, and occasionally had to prefer one over the other if the requirement made more sense for the context of use in one way over the other.

5.1 Typos

User Requirement 1: When entering text on mobile devices, user should not make more typos (slips) when operating with one hand compared to operating with both hands (U05-07, U5-08, U05-13, U05-23, U05-27, U05-29, U05-35, U01-14, U01-15, U03-04).

The user frequently made typos, in almost every other word, while typing on the phone with one hand (U05-07 to 08, U05-23, U05-35). They expressed frustration and annoyance towards making these slips (U05-13) and switched to typing with both hands when they became too irritated with making typos, even if that meant putting down the artifact that originally occupied their other hand to focus on typing (U05-27). This truly shows the level of frustration of the user with making typos. Moreover, when typing with both hands, the user experienced fewer typos compared to typing with one hand (U05-29). Taken together, it is clear that there is a salient breakdown in the existing method of one-handed text entry on mobile devices, hence this user requirement.

5.2 Keyboard assistive feature

User Requirement 2: User should not need to manually fix any results produced by the system's keyboard assistive features (e.g., autocorrect) when entering text on their mobile device with one hand (U05-09, U05-10, U05-11, U05-12, U05-13, U05-25, U05-27, U05-30).

The autocorrect feature appears to be ineffective and even counterproductive when the user is using one hand to enter text on the phone while multitasking (U05-09 to 13, U05-25). The default autocorrect feature on the phone would automatically change the typo word to a word that the algorithm believes is what the user intended to type, unless the user manually selected a suggested word displayed at the top of the keyboard (U05-09, U05-11). However, more often than not, the autocorrect feature got it wrong (U05-09), and the resulting word would sometimes be completely different from what the user intended, especially when it was an acronym (U05-10). Moreover, the user could not use the autocorrect feature on this word anymore because the word was now considered "correct" by autocorrect; consequently, the user had to manually delete and retype the entire word (U05-12). Forgetting to manually select from the suggested words was especially common when the user was using one hand and distracted by the surroundings (U05-11), causing the autocorrect to be in fact counterproductive and making the user very annoyed and frustrated (U05-13). In contrast, the user was able to use the autocorrect feature much more effectively and efficiently when typing with both hands (U05-30). Thus, it is clear that there is a breakdown in the existing autocorrect feature when entering text on the phone with one hand and multitasking.

5.3 Keyboard layout

User Requirement 3: User should be able to remove any text entered on the mobile device using one hand without accidentally modifying any other characters or words they did not intend (U05-25, U05-26, U05-27, U02-06, U04-17).

Autocorrect could only take the user so far: It was difficult for autocorrect to produce the right suggested words every single time, and when it did not, the user struggled to manually fix typos with just one hand, specifically in removing the typo (U05-25, U05-26, U05-27, U02-06, U04-17). In the case when the user's other hand was occupied, they had no choice but to remove the typo by first holding down the delete key, which frequently ended up over-deleting the word before it, rather than just up to and including the letter that differed from the right word (U02-06, U04-17, U05-26). As a result, the user had to spend more time retyping more letters than necessary. Moreover, the fact that the user would use both hands to fix typos when the other hand was not occupied (U05-12) and even when the other hand was occupied if they became so frustrated with fixing typos (U05-27) truly indicate the difficulty of fixing typos with just one hand when autocorrect did not produce the right suggestions which caused them to manually remove and retype the word. Therefore, this user requirement was created.

User Requirement 4: User should be able to interact with all keyboard elements using one hand without slips and without straining (overextending) their thumb (hand) or repositioning the mobile device (U04-02, U04-03, U04-05, U04-06, U04-12, U01-05, U01-07, U01-13, U02-02, U02-03, U02-08, U02-17, U02-20, U03-06, U03-08, U03-12, U04-06, U04-07, U04-09, U04-10, U04-11, U05-37).

The ability to comfortably reach the edges of the keyboard appeared to be a major breakdown in one-handed text entry on mobile devices. These interpretation notes all showed that, when operating their phone one-handed in utilizing various social media and messaging services, the user had to constantly strain their thumb to reach for different keyboard elements (and other general UI elements) located at the opposite side of the users' thumb, which was uncomfortable and frustrating. For example, some of these keyboard elements include the keyboard-switch key (U03-06, U05-37), the send/enter key (U02-02, U04-10, U04-11), capitalization key (U03-12), all of which are located on the edges of the keyboard, opposite to the user's thumb (depending on the thumb they were using to type). Moreover, as the user tried to reach the opposite end of the keyboard, they often misclicked on different keys (e.g., U02-17, U03-08, 04-11). To alleviate the discomfort, the user was sometimes forced to switch from one hand to two hands (e.g., U02-20) or reposition the device (e.g., U01-07). Users who enter text on the phone with one hand almost always need to access all keys on the keyboard with one hand. Thus, ensuring the user can comfortably and accurately interact with all elements on the keyboard one-handed is absolutely essential.

User Requirement 5: User should be able to locate and enter emojis, punctuations, and numbers on the mobile device with one hand faster than their default (i.e., current or existing) method of entry (U05-15, U05-16, U05-17, U05-37, U04-09, U04-18, U02-08, U03-06, U03-18, U03-19).

There appears to be a breakdown in entering non-letters with one hand, such as emojis (U05-15, U05-16, U05-17, U04-09, U04-18), punctuations, and numbers (U02-08, U03-06, U03-18, U03-19, U05-37). To enter these characters, the user had to first switch the keyboard (e.g., switching from the default letters keyboard to the emojis keyboard) by locating and pressing the keyboard-switch key, with which the user struggled, since this key is located at the edge of the keyboard (U05-15, U05-37, U03-18, U03-19). Then, the user had to find the desired characters on the new keyboard, which was also hard to do: For example, to select to desired emoji, the user had to consciously look at the keyboard rather than relying on muscle memory because the emojis keyboard has way more keys (i.e., emojis to choose from) compared to the letters keyboard and it also dynamically changes based on the frequently used emoji of the user (U05-16). A similar experience applies to the punctuations and numbers keyboard (U03-18, U05-37). Furthermore, because the user had to frequently enter these characters, almost in every message they sent, this process drastically slowed down the overall text entry speed (U05-17) and the user eventually stopped entering these characters due to its complexity (U05-37).

5.4 Physical discomfort

User Requirement 6: User should be able to hold their mobile device one-handed in a secure manner without inflicting physical discomfort (U01-07, U01-08, U01-09, U01-20, U01-21, U02-09, U02-10, U03-07, U03-13, U03-14, 04-04).

The user experienced difficulty in holding the phone with one hand, as indicated by all of these interpretation notes above. The trouble of holding the phone with one hand often led to fatigue (e.g., U02-10), which consequently resulted in the user either dropping their phone or worried that they might drop their phone (e.g., U01-08, U03-07). Sometimes, the user used their pinky as a rest stand to support their mobile devices; however, it did not take long before the phone's weight tired out the user's pinky (e.g., U02-09). Dropping a phone can damage it and even make the phone malfunction, which hinders the users' ability to enter text. Thus, it is important to ensure that the user is able to securely and comfortably hold their phones one-handed.

This concludes all of the user requirements we identified and improved.

5.5 Selected user requirements for qualitative user evaluation

We collectively decided which user requirements to evaluate our prototype against based on two criteria: The user requirement must be able to be tested (1) using qualitative user evaluation methods, and (2) using our current low-fidelity paper prototype with functionality being wizard-of-oz. This section first discussed the user requirements we eliminated because they did not meet at least one of these two criteria, then discussed the user requirements we selected, explaining how and why they would meet both criteria.

First, user requirements 1 and 6 were eliminated because they did not meet the second criterion. Evaluating user requirement 1 would require measuring the number of typos made by the participants while typing a piece of text on the prototype, which was definitely not feasible on our current paper prototype that did not have any "real" functionalities. Similarly, testing user requirement 6 would also need a high-fidelity prototype with appropriate size, weight, and surface that imitates that of a real mobile device and can stimulate the user's senses. Thus, we did not choose user requirements 1 and 6 for the current stage of user evaluation.

While user requirement 5 could be (at least somewhat) tested using a paper prototype with wizard-of-oz and we in fact, originally included it when designing the user goal and subgoals/tasks for the simplified user testing, after the lecture on quantitative user evaluation, we realized that requirements that have to do with comparison between designs (showing one design to be better than another) and measurement of speed, like that of user requirement 5, would best be tested by quantitative user evaluations. Thus, we did not include this user requirement as part of our analysis of the results of the current stage of user evaluation. The tasks that were originally designed to test this requirement were not removed because despite not being able to test this requirement, seeing how users perform on those tasks would still give us valuable insights on whether those features were usable or could be potentially helpful, such that we would possibly iterate on them and better evaluate them in the future using quantitative methods.

On the other hand, the remaining three user requirements (2, 3, 4) were indeed possible to be evaluated in the current stage of user evaluation. Successfully meeting user requirement 2 would mean that the user should not consciously notice any changes made by the prototype's keyboard assistive features. Thus, by introducing subgoals/tasks in the simplified user testing where keyboard assistive features, specifically autocorrect, would play a role, we would consider our design prototype to successfully meet this user requirement as long as the user did not explicitly mention anything that seemed off or counterproductive while doing those tasks. Moreover, our paper prototype already had features with appropriate frames/interactions for wizard-of-oz designed to address this requirement. Thus, this user requirement met both criteria, consequently, it was selected as one of the three requirements we tested our design against.

Likewise, user requirement 4 could also be evaluated using simplified user testing by creating various subgoals/tasks that require the user to interact with as many elements at different locations on the keyboard as possible, and if the user did not explicitly mention any difficulties (while doing think-aloud) nor show any actions that would imply any difficulties, such as long pauses, negative emotions, etc., we would consider our prototype successfully meeting this requirement. Just like previously, features that attempt to address this requirement were already implemented in the current paper prototype. Therefore, this requirement was selected for the current stage of evaluation.

To evaluate user requirement 3, we could use a task that requires the user to accurately delete some text, such as exactly one word, and in order for it to be considered as successfully met, the user should be able to use the new feature of the delete key of our prototype to delete the designated text. In contrast, using other methods or interactions such as repeatedly tapping the delete key or holding down the cursor to select the text would all be considered as not successfully meeting (or failing) this requirement since they did not use the new feature in our design, thus providing inconclusive evidence for whether our design prototype would meet this requirement. Additionally, this new feature with the relevant frames and interactions was available on the prototype and ready to be used. Hence, this requirement was also selected for the current stage of evaluation.

It is important to note that although user requirements 2 and 3 were selected because they technically met the two criteria we defined, they would still be evaluated with a better validity and more completely using a high-fidelity prototype, since the subgoals/tasks we created for simplified user testing with discounted, low-fidelity paper prototype would be simple and restricted, which might not fully represent the user requirements. However, even with discounted usability tests like simplified user testing, we could still gain some initial insights into the usefulness of our prototype and provide at least some evidence for whether or not the user requirements were successfully met. Thus, even when we say that a user requirement was considered to be met in the results section of simplified user testing or the discussion section, it does not indicate that the prototype has no issue or room for improvement regarding features that attempt to address that requirement, it only provides evidence that we were at least somewhat in the right direction and by implementing those features in a high-fidelity prototype, we could truly better evaluate the design against those requirements.

6 INITIAL DESIGN AND LOW FIDELITY PROTOTYPES

6.1 Design Critique

Jerry's Design

Switching to one-handed keyboard
Detail:

- Allows the user to easily switch between two-handed keyboard and one-handed keyboard, by sliding the thumb from left to right (note the keyboard currently designed for right-handed users).

Critiques:

- This feature may interfere with the swipe-to-text feature
- This feature indeed satisfies the requirements for left-handed users, as well as switching between one-handed keyboard and two-handed keyboard. I like the novel design idea. However, some apps might have preset functionality for sliding the thumb from left to right, this might need to mistouch. For example, in the Instagram home page, when you slide the thumb from left to right, it might go to the "post a new story" page rather than switching keyboards.

Shifted keyboard
Detail:

- The new keyboard is now shifted to the right, allowing users to easily reach all keys on the keyboard. This addresses User Requirements 5 and 6.

Critiques:

- Could also add a way to adjust how far the user wants the keyboard to be shifted
- I like the design idea, it can allow users to easily reach all keys on the keyboard without scratching the finger. This can reduce the fatigue to the user. However, this 'shrink' keyboard might have a smaller key size than usual, users might come across more mistouch and misspelled words.

Improved Autocorrect
Detail:

- Autocorrect can sometimes be counterproductive [User Requirement 2]. To prevent this, autocorrect no longer automatically changes the typo; it only underlines the typo and the user can choose to manually change it or not. If the user would like to change the typo, he simply clicks on the word and autocorrect will display a list of 9 words on the keyboard for him to select from; he can also scroll down the list of words to see more suggestions. If he finds the right word, he simply clicks on the word and Autocorrect will replace the typo with the word clicked. If cannot easily find the right word, he simply clicks on the "delete" key which deletes the entire typo, and he can then just retype it.

Critiques:

- I like how they provide more than one correction. Can probably order words from most likely to least likely based on user's habit
- I liked the idea, it improves the current autocorrect feature. For further improvement, how about combining Issac's idea to use ML features to show the 'most possible' word in the first place.

Improved Delete Key

Detail:

- Delete can be counterproductive, such as accidentally deleting words not part of the typo [User Requirement 3]. The "delete" key keeps the original functionalities, but also includes a new functionality: Pressing the delete key followed by sliding left will delete an entire word. This allows the user to quickly and accurately delete the typo itself and nothing else.

Critiques:

- If the user wants to delete multiple words in a row, does the user hold down their finger after the swipe? Or is it not allowed to do so? As long as the user has the need to hold down the finger to multi-delete, there will be accidental deletes of entire words.
- This design really expands the functionality of the current delete button, which might be useful in typing. But when the user slides left to delete the entire word, he might mistouch some keys. My suggestion is to let the user long press(or double click) and then slide left the key to avoid mistouching.

Punctuation keys

Detail:

- Users should easily enter punctuations [User Requirement 4]. The new punctuations key at the bottom right corner allows the user to conveniently enter four frequently-used punctuation marks with just one step, which is pressing on the corresponding punctuation key, eliminating the excessive steps of keyboard switching.

Critiques:

- The design is great. It expands the feature of punctuation. However, a user who is familiar with the original punctuation keyboard might not get accustomed at the first time. So maybe clarify that the user can choose either the original punctuation keyboard or the improved punctuation keyboard on the setting.

Emojis Key

Detail:

- Users should easily enter emojis [User Requirement 4]. The emojis key at the bottom right corner allows the user to conveniently enter any emojis. Clicking this key displays a list of frequently-used emojis at the top of the keyboard (replacing the suggested words of autocorrect), which the user can easily click on to enter them. If the user wants to enter some emoji not in the list, he can easily search for the emoji by typing the words that describe this desired emoji using the letters keyboard below (without doing any switching), and the list will be updated to reflect the search.

Critiques:

- This design may cause issues when a right handed individual needs to use their left hand to text from time to time

Special Characters Key
Detail:

- Users should easily enter some special characters [User Requirement 4]. Clicking the“star” key at the bottom row displays both the numbers and 10 frequently used characters at the top of the keyboard (replacing the suggested words from autocorrect) serving as an extension to the existing keyboard. This allows the user to easily enter any combination of letters, numbers, and special characters without switching keyboards back and forth.

Critiques:

- Features 6 and 7 are a little redundant. This is because the keyboard being squished to the right can already address the user requirement of easily entering numbers and emojis, as the user can just simply switch to the new keyboard and then enter them.
- Users had difficulty before mainly because they could not reach those keys in the first place, by switching to the new keyboard, they can now easily do so.

Daniel’s Design

Mini Mousepad
Detail:

- A transparent, square region near the dominant side of the phone. When activated, the mousepad will show a transparent circle to indicate a cursor which is controlled by the thumb motion. To select, the user needs to double tap any part of the box region. Once finished, the user can minimize it by tapping the arrow button.

Critiques:

- The mini mousepad might be too small so it would be difficult to accurately select UI elements. Also the thumb is not as dexterous as other fingers such as the index finger, so it might be even harder to accurately manipulate the cursor.
- An intuitive idea for navigating UI elements. However, depending on the position and size of the activation region as well as what widget this activation region exists in, it is possible to easily accidentally activate this if this “box” always persists on screen.

Rotating Keyboard
Detail:

- The rotating keyboard will allow users to shift the keyboard horizontally. There is a scrollbar near the bottom of the keyboard to prevent users from over rotating the keyboard. However, to rotate, the user can swipe anywhere on the keyboard.

Critiques:

- Users may accidentally rotate the keyboard. For instance, if the user wants to press a key but then his fingers accidentally slides, the keyboard might rotate. Also, there should be a quick access button that automatically resets the keyboard to its default layout.
- A solution for reaching edge keys. However, this can introduce extra steps of sliding back and forth with the thumb to input a single, which can make text entry a bit tedious. The user would have to lift their finger from a key to touch a scroll bar at the bottom between key inputs

Franklin's Design

Perspective keyboard
Detail:

- Similar to the perspective zooming feature on the iPhone's wallpaper, when the user tilts their phone during text entry on their virtual keyboard, the keys would shift left and right and enlarge and/or shrink based on the user's tilting action. There should also exist a deadzone where if the phone is somewhere near leveled, the tilting effect should not happen or is drastically toned down. The option to configure these should be available.

Critiques:

- Unintentional tilting. For instance, if the user is walking, there will be a small amount of tilting, which can cause the keys to shift as well.
- Difficult to type fast since the position of the keys will always be changing. So the user can't use their muscle memory to memorize where each key is. Depending on the tilting angle, keys will be in different positions, so the user has to find the specific keys.
- The design definitely helps to address the User Requirement 4, as the user will be able to comfortably reach the keys on the left edge and on the right edge of the keyboard without repositioning their hand, and 6, since they are now able to securely hold their phone.

Moving functional keys to top corners
Detail:

- The newly designed keyboard layout places more functional keys like the numeric keyboard, keyboard switch, and uppercase key to the upper right corner of the keyboard. This is a very natural location where Thomas would place his thumb, without having to shift the phone in his hand too much and compromising the hand's hold on the phone. Placing these keys too low would induce the thumb to move down towards the palm, and the palm would open up, losing surface contact with the back of the phone.

Critiques:

- If you place too many functional keys on top, it will become very cluttered. Maybe the user can have an option to customize that area of the screen.
- This design definitely helps to address User Requirement 4, as it allows the user to comfortably reach the numeric-keyboard-switch key, language-switch key, and uppercase key, and it also helps to address some of User Requirement 5, since it allows easy switching between keyboards. However, I think the user might still struggle a bit with entering let's punctuations/numbers/emojis because after all, they still have to switch to a different keyboard to enter those non-alphabetical characters, and then switch back the keyboard. I think we can try to find a way to allow users to enter frequently-used non-alphabetical characters without even switching the keyboard at all.

Isaac's Design

Predictive key enlargement
Detail:

- A ML model will analyze the current app that the user is using, the context, the text/sentence content, and current word that the user is typing. The ML model will generate predictions of the next letter that the user is going to press. Those prediction keys will be enlarged. Larger than the other keys.

Critiques:

- This is actually a very interesting idea. It helps address User Requirement 4, since it ideally will allow the user to comfortably reach the keys they want to enter. There might be too many possibilities initially (after the user only typed 2 or 3 letters), but after a certain threshold of letters are entered, the probability of correctly predicting the letters the user is about to enter and the correspondingly enlarged keys will be more accurate. This feature can be more helpful to help predict non-alphabetical keys (like punctuation, special characters, and emojis), and display their keys on the top of the keyboard or something.

Yichen's Design

Quick switch function key that can switch among different keyboards
Details:

- When switching between traditional language keyboards, people often find it difficult to control it with one hand, so I designed a quick-switch button. The key sits atop several existing alphanumeric keys, making it easy to reach with one hand. When you long-press, the quick switch will show up. The left half of this button is the shortcut key for the emoji keyboard, and the right half is the shortcut key for switching between language keyboards.

Critiques:

- I like how these buttons are now more accessible. However, there should also be a way to change which keyboard each shortcut key maps to. For example, if the user uses more punctuation and emojis, the user should be able to configure the quick switch to open the punctuation keyboard instead of the language keyboard.

One-handed mode that allow users to type in one hand easily
Details:

- I designed a one-handed operation mode to allow people to easily type with one hand. When using this one-handed operation mode for the first time, you need to manually turn on the one-handed keyboard in Settings-Convenient Mode. After turning on this action, the keyboard will automatically switch to one-handed mode. Unlike ordinary fixed keyboards, the keyboard can slide left and right in one-handed mode. You can easily slide the keyboard with your thumb and use other fingers for one-handed typing. This design can solve the trouble of often having to switch hand postures to touch different buttons.

Critiques:

- I like how this feature lessens the stress on users' hands when texting on one hand. We should make it easier for users to toggle on/off this feature, so the users do not accidentally slide the keyboard.

6.2 Personas

For the final primary persona, we tried to merge some of our individual personas into one single entity. We only kept the most important features and needs into our final persona, and left the other ones as secondary personas (not shown here). This final persona also is closely grounded in the contextual inquiry results and also incorporated the user requirements into our persona.
Name: Jack
Occupation: College student
Age: 21 years old

Fig. 27. Picture of Jack

Jack is a 21 year old college student who is currently studying business management. He is in his fourth year of college, so his daily life is extremely hectic and busy. Not only does he want to get good grades in class, but he also needs to juggle his spare time between studying, his social life, and job searching. Therefore, more often than not, during his free time, he finds himself consistently multitasking between many tasks. His primary goal right now in life is to find a job before he graduates. Therefore, he is always looking for potential emails, text messages, messages on social media from companies to see if he got rejected or moved on to the next round of interviews.

Like many others his age, he uses an iPhone 13 and is extremely addicted to it. He finds himself constantly checking his phone and getting sidetracked. In addition, his phone is always next to him because he wants to be notified as soon as possible whether someone texted him or when he receives an email. He wants to be constantly updated with the most recent news and feels bad if he leaves someone on read, so he will reply as soon as possible. Jack is also bilingual, he speaks both Chinese and English; he occasionally speaks Chinese to his parents, and mainly uses English during the day. Usually, Jack uses his phone with both hands when he is exclusively focused on just the phone. However, whenever Jack uses his phone when he is performing other tasks or when one hand is occupied, such as eating, brushing his teeth, walking outside, on public transportation, and lying on his bed, all of which also happen very frequently, he tends to operate his phone with just one hand.

In the situations when he is using the phone with one hand, Kevin is either messaging his friends and family on iMessage or WeChat, checking messages on Gmail or LinkedIn, browsing social media like YouTube and TikTok, or searching things on Google. All of these activities involve entering text using the virtual keyboard on the phone with just one hand, which Jack significantly struggles with, specifically in terms of:

(1) the high frequency of typos;
(2) difficulty in fixing the typos caused by counter productivity of auto correct and delete;
(3) inconvenient location of certain keys (like punctuations, emojis, and other special characters);
(4) uncomfortable size of keyboard and phone leading to difficulty to reach all keys and physical discomfort

Thus, Jack has the following specific desires/needs.

(1) Jack wants to minimize the amount of typos he makes when he is entering text with one hand, which should not be more than the amount of typos when he is typing with both hands [User Requirement 1].
(2) Jack wants to easily fix any typos using one hand without doing any redundant actions (i.e. more than what is necessary) such as
    (a) correcting mistakenly-corrected words by keyboard assistive features like autocorrect [User Requirement 2], or
    (b) accidentally deleting correctly-spelled letters/words not part of the typo [User Requirement 3].
(3) Jack wants to be able to easily locate and enter non-alphabet characters with one hand, including emojis, punctuations, and numbers [User Requirement 5].
(4) Jack wants to be able to comfortably reach all keyboard elements while securely holding the phone with one hand [User Requirement 4, 6].

6.3 Sketches

The final design (sketches & storyboards) was created based on the feedback received from design critique. Each team member actually had very different designs, meaning that our user requirements were well-written as they do not imply a specific solution. First, we tried our best to merge the different features of our designs that attempted to resolve the same need of the persona (i.e., addresses the same user requirement(s)), but if we struggled to do so, we collectively voted to keep the best one and discard the others. For example, some of Jerry's, Yichen's, Daniel's, Franklin's features all addressed the same need of being able to reach all elements on the keyboard (Jack's 4th need, User Requirements 4, 6), and they were vastly different designs. After careful consideration, we decided to keep the one that is the most feasible (for the paper-prototype and wizard-of-oz) and adequately (in our opinion) resolves the persona's needs. Next, for each of the features we decided to keep in the final design, we redrew/modified the sketches and also tried to improve on them based on the critique (see each of the following sketches to see the things we improved based on the design critique). Lastly, the storyboards were modified to illustrate the final primary persona and the context of use for our final design.

Sketch 1 presents the new one-handed keyboard as a whole, which attempts to address all of Jack's needs. Sketch 2 shows the process of transitioning between the existing two-handed keyboard (current design) and the new one-handed keyboard (my design). Each of the remaining sketches (3-6) shows a specific part/feature of the overall keyboard, explaining how it attempts to address one (or more) of Jack's needs.

6.3.1 Sketch 1. This sketch (see Figure 28) illustrates the overall one-handed keyboard layout and general description of its features/keys. One major difference compared to the two-handed keyboard is the shift of the keyboard to the right side of the screen. By doing so, Jack should be able to easily reach all keys, including those that are on the left edge of the keyboard, thus accounting for Jack's 4th need, where he would like to comfortably reach all elements of the keyboard while holding his phone securely. I did not make any significant changes to the layout of the letters because Jack is already so familiar with typing in the "QWERTY" layout, consequently, switching to a drastically different layout not only takes a long time to learn, but also may not even make any meaningful difference in meeting the user requirements. Thus, I just went with making gradual improvements to the existing layout instead of coming up with a breakthrough. One additional feature we added to this was the ability to adjust how much the keyboard is shifted or squished towards the right (see Sketch 3 for more details).

Another noticeable difference is that the "return" key at the bottom right corner is replaced with two separate keys. Before discussing what those keys are, I would like to elaborate on why the return key is removed. Jack is usually in informal situations when using his phone with one hand, such as texting, Google searching, or browsing social media. This means that the "return" key does not function as creating a "newline"; instead, it serves the purpose of "send" (sending his message, his search query, or his comment). However, these apps which Jack uses to text, search, or browse social media, already have a separate "send" key apart from the keyboard, so the "return" key in these situations are actually redundant. Therefore, it is removed and I believe that it does not cause any reduction in functionality. As for the two new keys, they attempt to address Jack's 3rd need which is to conveniently enter certain characters; specifically, these two keys are for punctuations (see Sketch 6 for details).

Originally, this keyboard also had 2 additional special keys that attempted to address the remainder of Jack's 3rd need, which was to easily enter numbers and emojis. However, based on the design critique, I realized that these keys not only would make the keyboard's bottom row too squished, but they were also a little redundant since the fact that the keyboard being shifted towards the right already implicitly addressed the need to easily enter numbers and emojis, because Jack would now be able to easily reach the emoji/number-switch key on the left side of the keyboard, which was the main breakdown we identified from our contextual inquiry results. Thus, these two keys were removed in the final design.

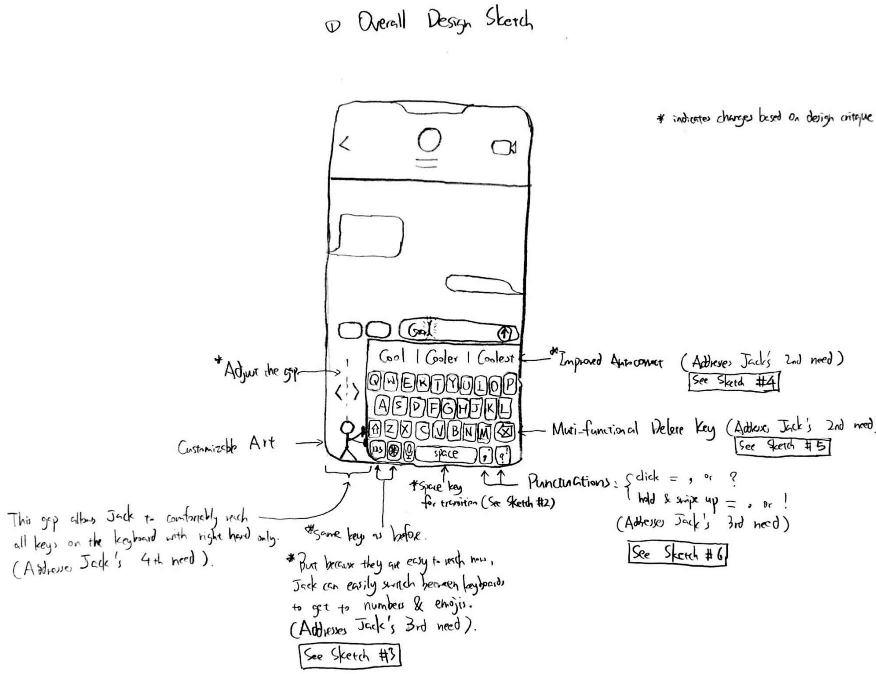

Fig. 28. Sketch 1

Despite not having any salient differences in terms of their appearances in this sketch, the autocorrect feature (see Sketch 4) and delete key (see Sketch 5) both have improved functionality, in order to resolve Jack's 2nd need, which is to easily fix any typos.

Lastly, although Jack's 1st need is not explicitly addressed by any particular key/feature, I believe that when considering all of the features together as a whole, this new one-handed keyboard would be able to reduce the frequency of typos, thus resolving Jack's 1st need.

6.3.2 Sketch 2. This sketch (see Figure 29) shows the process of switching from the existing two-handed keyboard to the new one-handed keyboard (my design), and vice versa. To do so, Jack simply needs to slide his (right) thumb from the left side on the keyboard's space key to the right side (as shown in the sketch). To switch back, Jack simply needs to swipe in the reverse direction (from right to left). Note that, since Jack is right-handed, the one-handed keyboard and this transition motion are designed for people who type with the right hand (thumb). In the future, we plan to design for people who mainly type with their left hand as well.

This feature was improved based on the design critique. Originally, this feature allowed the user to swipe anywhere on the keyboard to switch between two-handed keyboard and one-handed keyboard. However, during the design critique, we pointed out that although this allowed a convenient transition, swiping anywhere on the keyboard might interfere with other features on the keyboard and the screen/app, for instance, the swipe-to-text feature and the "back" feature (which are existing features on the iPhone). Thus, we decided to change the area which can pick up this interaction to just the space key, which does not have any existing features when the user swipes on it.

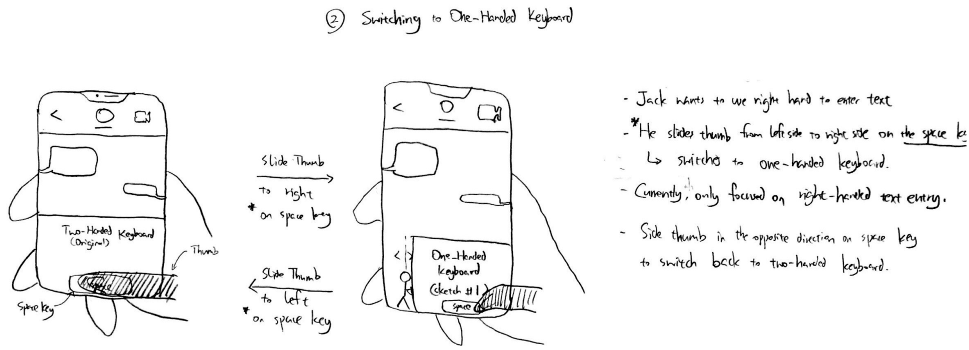

Fig. 29. Sketch 2

6.3.3 Sketch 3. This sketch (see Figure 30) is a newly added feature based on the design critique, which allows Jack to easily adjust the width of the new keyboard, or how much the keyboard is "squished". This gives him the flexibility since Jack might be holding the phone with different hand positions from time to time, so the need of how well he needs to reach all keys on the keyboard may change. Thus, the addition of this feature addresses Jack's 4th need more optimally.

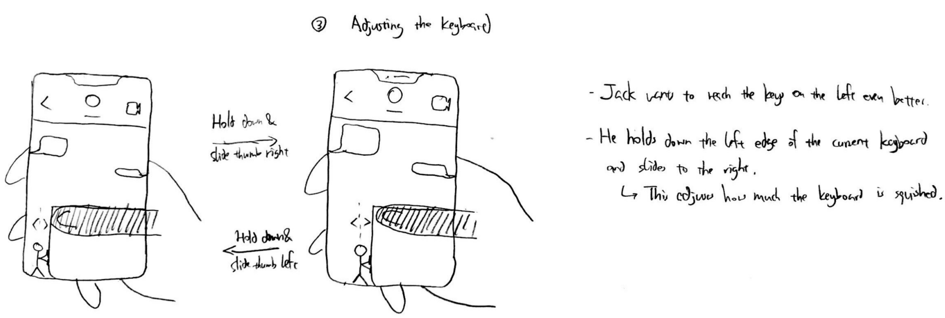

Fig. 30. Sketch 3

6.3.4 Sketch 4. The autocorrect feature in the original two-handed keyboard can be counterproductive, specifically when it automatically changes a typo to a correctly-spelled word but is not what Jack intended, which happens frequently. In order for autocorrect to not automatically change the typo, Jack has to manually select one of the three suggested words displayed at the top of the keyboard, which he tends to forget especially when he is multitasking or distracted, resulting in autocorrect being counterproductive when Jack is entering text with one hand.

In our new one-handed keyboard (see Figure 31), autocorrect no longer automatically changes the typo; it only underlines the typo and Jack can choose to manually change it or not. If Jack would like to change the typo, he simply clicks on the word and autocorrect will display a list of 9 words on the keyboard for him to select from; he can also scroll down the list of words to see more suggestions. If he finds the right word, he simply clicks on the word and Autocorrect will replace the typo with the word clicked. If cannot easily find the right word, he simply clicks on the "delete" key which deletes the entire typo, and he can then just retype it.

Based on the design critique, we also decided to incorporate Machine Learning models that learn from the patterns which Jack types to produce a better list of words that have a really high chance

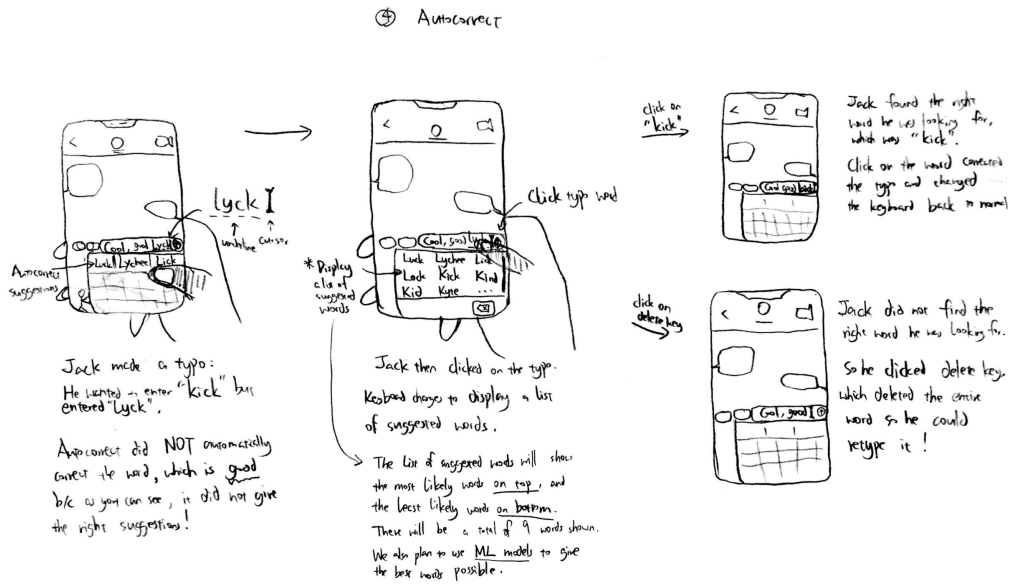

Fig. 31. Sketch 4

of giving the right corrections (merging with another feature by a different team member). We also changed the order in which the words appear to display the most likely words (the algorithm believed to be correct) at the top, and the less likely words at the bottom. The list would also dynamically update based on Machine Learning.

This improved functionality takes care of Jack's 2nd need (specifically part a), because he no longer needs to fix any mistakes made by autocorrect, therefore, the autocorrect feature is not counterproductive.

6.3.5 Sketch 5. In the original two-handed keyboard, holding down the "delete" key will continuously delete one character at a time at a fast pace. This functionality is very difficult to control, as reflected in Jack often deleting not just the typo, but also the correctly-spelled word(s) in front of the typo. Thus, the delete key is originally counterproductive.

In the new one-handed keyboard (see Figure 32), the "delete" key keeps the original functionalities, but also includes a new functionality: Pressing the delete key followed by sliding left will delete an entire word. This allows Jack to quickly and accurately delete the typo itself and nothing else, thus eliminating any counterproductivity caused by delete, resolving Jack's 2nd need (part b).

This feature was also improved based on the design critique, which we realized that the swiping left interaction may result in the user accidentally clicking on the keys next to the delete key. Thus, to fix this, whenever the user clicks on the delete key, all of the remaining keys automatically are disabled (i.e., clicking them would not result in any text entry / functionality). A visual would also be shown that the keys cannot be clicked. As such, the delete key would accurately delete the words without causing more slips, better resolving Jack's 2nd need.

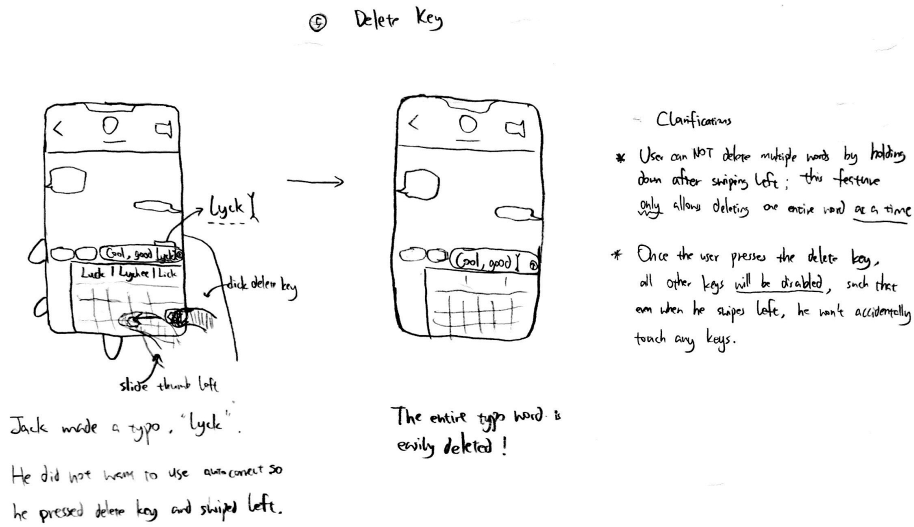

Fig. 32. Sketch 5

6.3.6 Sketch 6. In the original keyboard, whenever Jack has to enter a punctuation, including the frequently used ones like comma and question mark, he has to switch to the numbers/punctuations keyboard by pressing the keyboard-switch key at the bottom left, locate/enter the punctuation, switch back to the letters keyboard. He has to perform these three steps whenever he wants to enter any punctuation, which is complicated and difficult, especially when operating the phone with just one hand.

In the new one-handed keyboard (see Figure 33), the punctuations key at the bottom right corner allows Jack to conveniently enter four frequently-used punctuation marks with just one step, which is pressing on the corresponding punctuation key, eliminating the excessive steps of keyboard switching. These two keys have default values of ',' and '?' respectively, which can be entered by a simple click. They also offer two alternative values of '.' (period) and '!' respectively, which can be entered by clicking and sliding up, which despite requiring one extra motion, is still much simpler than switching keyboards back and forth. Thus, this new addition of punctuation keys will account for part of Jack's 3rd need, which is to enter non-alphabetical characters, in this case, the punctuations, conveniently.

Like mentioned previously under Sketch 1, this keyboard also had 2 additional special keys that attempted to address the remainder of Jack's 3rd need, which was to easily enter numbers and emojis. But based on the design critique, we removed those two keys as they were redundant as the remainder of Jack's 3rd need would be resolved by the shifted keyboard already (Sketch 1, 3). Thus, this was changed after the design critique for this feature.

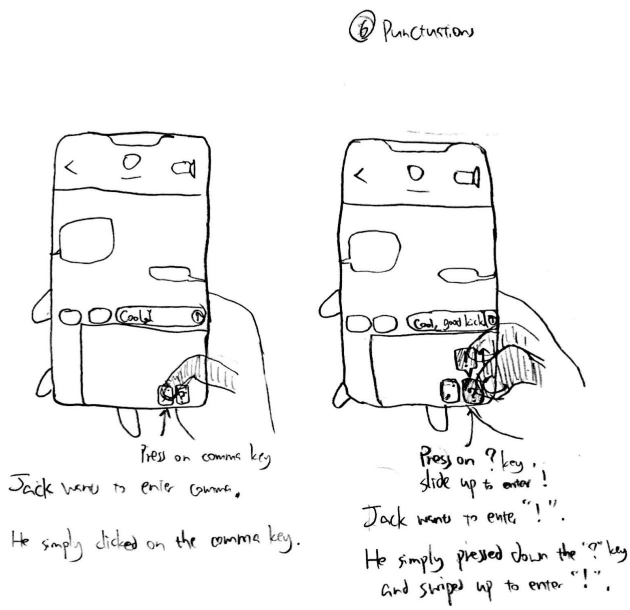

Fig. 33. Sketch 6

6.4 Storyboards

Storyboard 1 extends on Sketch 1, 2, 3. Storyboard 2 extends on Sketch 4, 5. Storyboard 3 extends on Sketch 6.

6.4.1 Storyboard 1. This storyboard (see Figure 34) depicts how Jack, who is trying to reply to a message to his friend with one hand (specifically, his right hand) while brushing his teeth with his other hand, initially struggles with entering text on the current keyboard on his phone, because his thumb could not reach the keys on the left edge of the keyboard.

However, by changing (Sketch 2) to my new one-handed keyboard design (shown in Sketch 1 in detail), he is able to comfortably reach all keys on the keyboard without readjusting his hand position thus securely holding the phone. Consequently, he is able to effectively complete his message and send it to his friend. Thus, this successfully addresses Jack's 4th need (and also part of Jack's 3rd need).

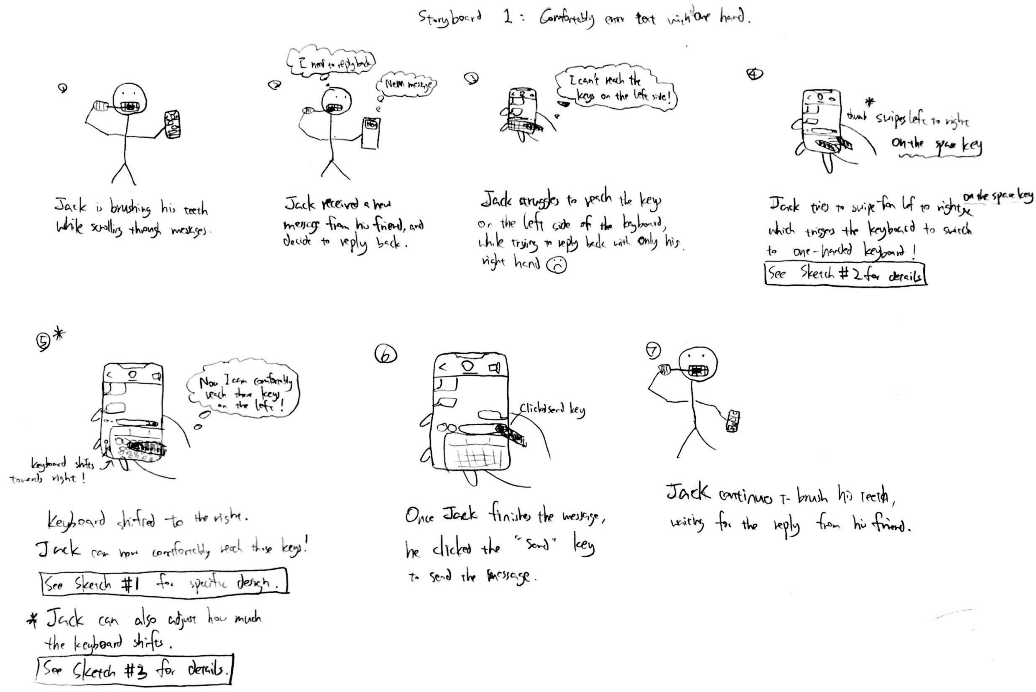

Fig. 34. Storyboard 1

6.4.2 Storyboard 2. This storyboard (see Figure 35) depicts how Jack, who is trying to reply to a message from his family member with one hand while walking to class, struggles to fix typos on the current keyboard design because the autocorrect incorrectly changes his typo to a word he did not intend to type, and while deleting this word, he accidentally over-deletes the previous correctly-spelled word too.

However, by changing to the new one-handed keyboard design, Jack is able to easily fix typos because the autocorrect and the delete key are no longer counterproductive, as the former allows Jack to manually pick from a listed of suggested word (Sketch 4), whereas the latter allows Jack to accurately and efficiently delete a word (Sketch 5). Consequently, he is able to efficiently finish the message and send it to his family member. Thus, this successfully addresses Jack's 2nd need.

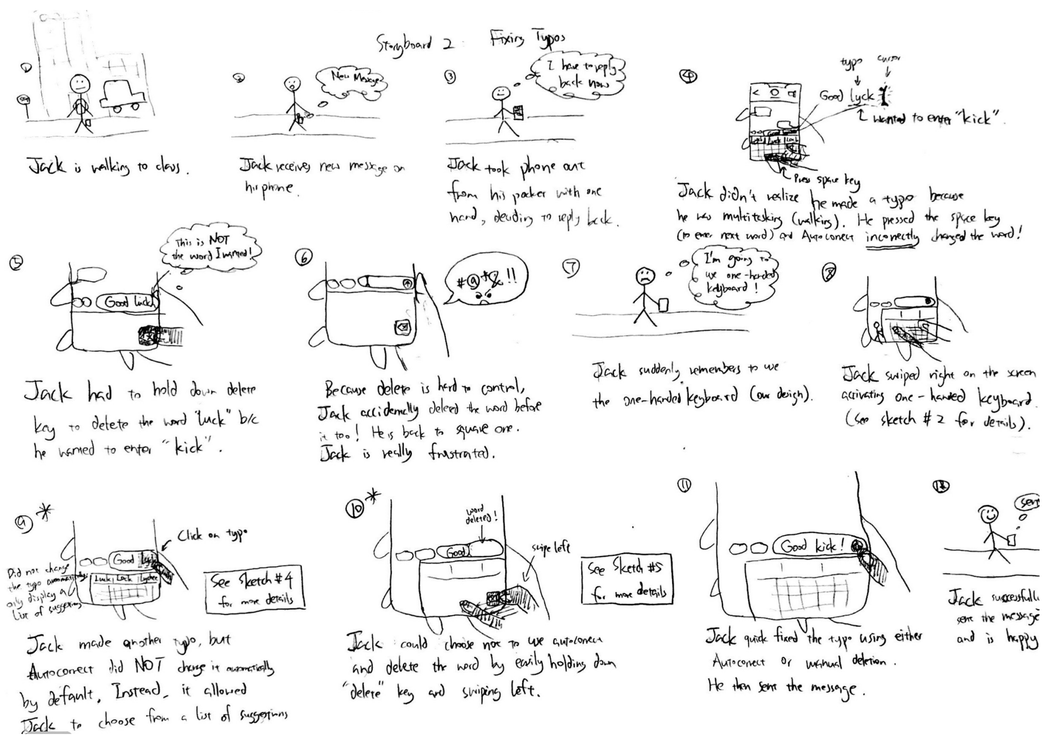

Fig. 35. Storyboard 2

6.4.3 Storyboard 3. This storyboard (see Figure 36) depicts how Jack, who is trying to comment on YouTube with one hand while eating food with the other hand, struggles to enter punctuations marks and emojis, because everytime he wants to enter one of them, he has to switch to a different keyboard, find the desired element, and switch back to the letters keyboard to enter the next word.

However, by changing to the new one-handed keyboard design, Jack is able to conveniently enter frequently-used punctuation marks, including comma, period, question mark, and exclamation mark (Sketch 6), and also easily enter desired emojis (and numbers) because he can now easily reach all elements on the keyboard. Consequently, he is able to efficiently finish the comment and publish it on YouTube. Thus, this successfully addresses part of Jack's 3rd need.

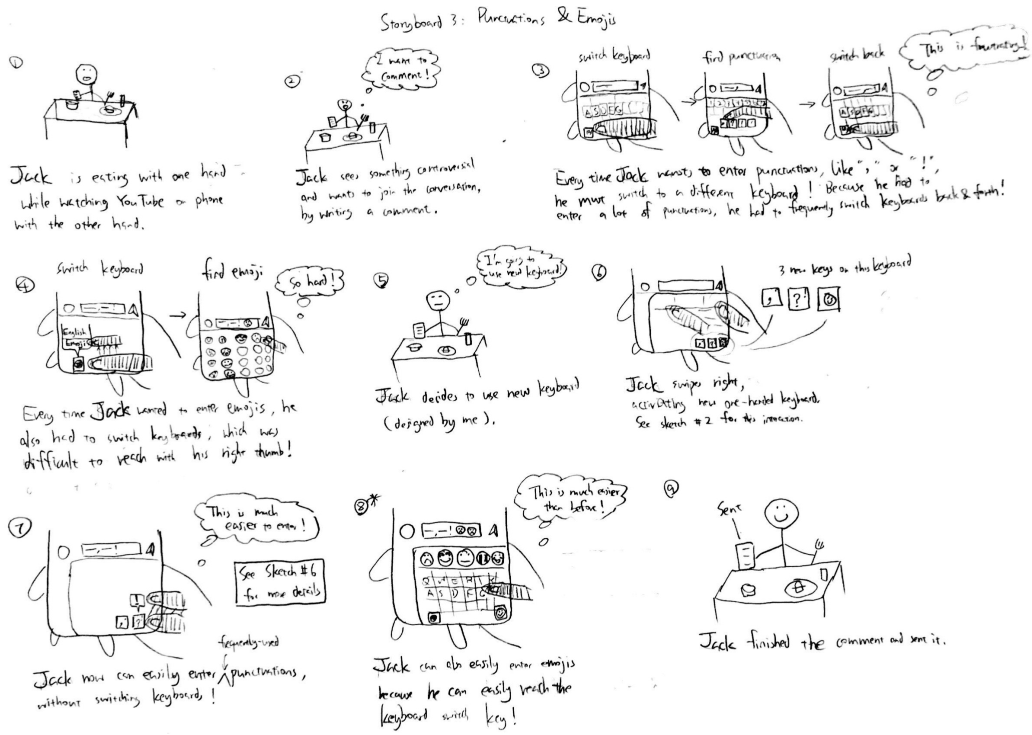

Fig. 36. Storyboard 3

6.5 Paper Prototype

Our final paper prototype was implemented closely following the features and interactions shown in the final sketches and storyboards (see Figure 37 or Appendix D.7 for higher definition). Because we attempted to satisfy all of Jack's needs (primary persona) when creating our final sketches and storyboards, our paper prototype should also address those needs, and by extension, the six user requirements. To reiterate what we mentioned in the Final Sketches and Storyboards section, we have also recorded other feature design ideas that are not incorporated into this iteration of our prototype, mainly due to conflicting designs for a certain need/requirement. Our rationale for choosing one feature over the other was that the selected feature should adequately address Jack's needs (and by extension, the user requirements) while being conservative with current keyboard layouts and being feasible for a Wizard-of-Oz. We did not yet want to incorporate features that drastically overhaul current widely-accepted solutions, but we may revise our prototype in the future and incorporate other features consisting of relatively more radical changes, given our user evaluation feedback.

The paper prototype consisted of 2 main parts: (1) frames and (2) flow between frames. Each frame, or a snapshot of the interface, in the prototype is built upon a sketch; it clearly shows the context for which the user is using the phone, namely, messaging a friend, and the keyboard design with its newly added features. The flow between frames, indicated by the red boxes and arrows, explains how/where the user should interact with the prototype to go from the current frame to the next frame, which is also built upon the sketches and the storyboards. For example, to go from the first frame, which depicts the current/existing two-handed keyboard, to the second frame, which depicts our new one-handed keyboard, the user should press the space key and slide right, as indicated by red boxes and arrows. All of the remaining frames and flows follow the same standardization, so it is easy to understand how the prototype works. It is redundant to go over each frame and flow of the prototype one-by-one (since these are similar to the sketches and storyboards), so we mainly focused on discussing the changes we made and how they were implemented in the final paper prototype (from sketches/storyboards) based on the design critique.

6.5.1 Changes based on design critique. First, the design critique raised concerns that the activation of the one-hand keyboard may interfere with the default swipe-to-text feature of the keyboard if the user were to just swipe at an arbitrary location on the keyboard. To counter this issue, the activation for this feature was moved to swiping the spacebar instead, which served no purpose on the default (current) keyboard (see *1 in Figure 37). Moreover, to increase the customizability of the one-handed keyboard, the user can now manually adjust how much they want to shift the keyboard depending on the magnitude of their swiping-action; the keyboard can now be squished until the characters symbols on the keys have zero padding (see *2 in Figure 37).

Based on the design critique, the autocorrect feature is now utilizing a context-sensitive language model to make good predictions for misspelled words and the user is given a list to choose from (see *3 in Figure 37).

The new delete feature is revised by deactivating all other keys on the keyboard while the user is pressing and swiping the delete key, so that the unwarranted pressing of other keys would not interfere with this action (see *4 in Figure 37).

Lastly, having the emoji and number keyboard switch key at the bottom row of the keyboard was deprecated in the final paper prototype because this was deemed to be a redundant feature in the design critique. The original purpose was to help the user locate and enter numbers and emojis with ease; however, since the one-hand keyboard is already making edge elements easier to reach with one hand, the user could simply switch to emoji and numbers keyboard with the conventional way.

6.5.2 Wizard of Oz. Because the nature of our paper prototype is broken down into frames and flow between frames (arrows), it can be easily adapted to a Wizard of Oz. To Wizard of Oz our prototype, we first need to print out each frame in the prototype on paper, where each frame should be about the size of an iPhone. We tell the participant in the simplified user testing that their goal is to send a message, “Cool, good kick!”, to their friend using the prototype with one hand, which is what our paper prototype is built around. The participant begins by physically holding the very first frame shown in Figure 37 with one hand. Then, based on the participant’s interaction with the current frame, we can present different frames to the participant by quickly switching the current frame out from their hand with the next frame, according to the flow of frames, namely the arrows, shown in Figure 37. Because the participant is trying to do a specific goal, there are only a certain set of interactions (tasks) they could do with the keyboard, which would inevitably have them go through most if not all of the frames in the prototype. As such, we can use the paper prototype to conduct a Wizard of Oz that allows the participant to experience all of the features and functionalities of our design.

Fig. 37. Paper Prototype

7 USABILITY EVALUATION

7.1 Heuristic Evaluation

7.1.1 Purpose. The purpose of the heuristic evaluation was to get some initial feedback on our design, specifically regarding its usability, from usability experts via our narrow-down, low-fidelity prototype. The feedback we received can be used later to iterate on the prototype and to address its salient usability issues.

7.1.2 Method. First, as a group, we picked one realistic, overarching user goal from the context of use that could be feasibly achieved using our low-fidelity paper prototype: "Message a friend on a mobile device using just the right hand". This goal was then broken down into subgoals or tasks which the users would perform to accomplish the overall goal. We attempted to make these subgoals/tasks cover all three user requirements we were going to test and (almost) all of the available features of our prototype. Thus, by performing these tasks one by one, the user would inevitably explore all parts of the prototype, thus allowing us to later evaluate whether the user requirements were met as a group. The specific tasks are listed in the following section 7.1.3 and further elaborated in section 7.2.2. The same goal and subgoals/tasks were used for both heuristic evaluation and simplified user testing.

Next, each team member conducted a heuristic evaluation with a (different) usability expert, a student (not from our team) in EECS 593, to identify usability issues with our prototype and our design in general. During the heuristic evaluation, we demonstrated to the expert how we expect the user to interact with our prototype to accomplish the goal, and the expert rated the usability of the prototype and identified its issues with a list of usability heuristics (see more details in the following section 7.1.3).

After each team member concluded their individual heuristic evaluation, as a group, we consolidated the findings of the individual evaluations into one table, showing all usability problems identified and their relevant information including which expert identified the problem, the heuristic violated by the problem, and the severity of the problem. We also analyzed the table to determine which issues should be addressed immediately, so we can resolve them by iterating on the prototype in the future.

7.1.3 Tasks and Procedures. To ensure the quality and consistency of our heuristic evaluations, each team member closely followed the protocol we created as a group, which was a bare-bones, bulleted outline regarding the overall user goal and corresponding subgoals/tasks that we planned to demonstrate using the paper prototype, and the process for conducting the heuristic evaluation. This section elaborated on the protocol, discussing it in detail while maintaining its core aspects, such that it accurately presented what each team member did to conduct the heuristic evaluation.

Each team member (referred to as the investigator in the remainder of this section) contacted their own usability expert via email to set up the date and time for a meeting (either online or in person).

After arriving at the meeting, the investigator and the usability expert first exchanged some information, including uniqname and group name. The investigator then briefly explained our focus and context of use ("one-handed text entry on mobile devices") and the user goal we were going to demonstrate for this heuristic evaluation ("message a friend on a mobile device using just the right hand"). The expert was instructed to record any usability issues/problems they identified in the design/prototype with respect to the 10 usability heuristics, during the demo. For each usability issue identified, the expert also indicated to which heuristic the issue corresponds and rated the severity of the issue on a scale of 0 (*not a problem at all*) to 4 (*usability catastrophe*). The 10 usability heuristics that the expert used to evaluate the prototype are listed below.

(1) Visibility of system status
(2) Match between system and the real world
(3) User control and freedom
(4) Consistency and standards
(5) Error prevention
(6) Recognition rather than recall
(7) Flexibility and efficiency of use
(8) Aesthetic and minimalist design
(9) Help users recognize, diagnose, and recover from errors
(10) Help and documentation

After the above information and instructions were clearly conveyed to the usability expert, the heuristic evaluation then began according to the instructions, with the investigator demonstrating the prototype on how the user would use it to achieve the goal, while the usability expert recorded down any usability problems they identified. Specifically, the investigator demonstrated the following subgoals/tasks one by one using the paper prototype:

(1) Switch to one-handed keyboard and adjust keyboard width
(2) Enter text "Cool, good"
(3) Enter "luck"
(4) The user realized that they accidentally made a typo, i.e., entered "lyck", fix the typo
(5) The user changed their mind and wanted to enter "kick" instead, replace "luck" with "kick"
(6) Enter "!"
(7) Enter FACE WITH TEARS OF JOY emoji
(8) Send the message

At any time during the demo, the expert was allowed to ask clarification questions about each subgoal/task, or the design/prototype. After the demonstration ended, the expert gave their notes, the usability problems with their corresponding information, to the investigator and briefly explained each problem. The investigator also had a chance to ask any clarification questions about the issues. The entire heuristic evaluation lasted about 30 minutes. Afterward, the investigator thanked the expert, and the meeting was concluded (Remark: Technically, the investigator and expert then switched roles because the other student had to demonstrate their prototype to us, but this was not important for our study, thus not being presented in the report).

7.1.4 Participants. There were a total of 5 usability experts, each participating in a heuristic evaluation which a team member conducted. Their information is listed in Table 8.

Encoding	Usability expert	Group
E1	esing	18
E2	lkurek	3
E3	tonytang	20
E4	pinhan	11
E5	zixiangz	20

Table 8. Usability expert information.

7.1.5 Results. Listed below are the design issues and the respective scores noted by the usability experts. As mentioned previously, the design issues are based on Nielson's Usability Heuristics, and the scores range from 0, not an issue, to 4, a usability catastrophe.

Usability problem	E1	E2	E3	E4	E5
No indication for how to switch to our new one-handed keyboard design (#10, #6, #1, #7; 4)	2	3		4	3
No indication of the extended delete key feature (#1, #10; 2)	1		4		
The user still needs to touch the very left of the screen to fix the beginning words (#3; 1)	2				
Need to allow users to delete characters, not just words (#7; 1)	1				
Deviates from the standard keyboard layout (#4; 1)		3			
No indication users can press on the typo word to be taken to the table/list of suggested words. (#1, #6, #7; 2)		3		2	
The user cannot return to the 'ABC' keyboard from the emoji and number/punctuation keyboard. (#4, #3; 2)		3			3
The user is locked to comma, and question mark as default for the bottom right keys (#3; 1)		2			
The user cannot toggle the keyboard cosmetics/typo lists off. (#3, #5, #9; 4)		1	3	3	3
The user cannot continuously delete words with the new delete feature. (#3; 1)		1			
The user can mistakenly minimize the keyboard for one handed use by accidentally dragging the keyboard. (#5; 1)			1		
The user does not know you can swipe up to type an exclamation mark. (#10; 1)			4		
No apparent limit to the keyboard minimization feature. (#3; 1)			2		
The animation of extended delete key and disabled letter keys being triggered every time the delete key is pressed is not only initially confusing to new users, but also annoy experienced users (#1, #7; 1)				2	

Table 9. Heuristic evaluation results.

7.2 Simplified User Testing

7.2.1 Purpose. The purpose of the simplified user testing was to get some initial feedback on our design, specifically regarding both its usefulness and usability, from real stakeholders via our narrow-down, low-fidelity prototype. The feedback we received allows us to not only realize the usability issues regarding our prototype, but also to better understand the context of use and whether our design addresses the user requirements, so we can later iterate on the prototype to improve it.

7.2.2 Method. The same user goal and the corresponding subgoals/tasks from heuristic evaluation were used for the simplified user testing (see more in the Tasks & Procedures section). To briefly reiterate, we collectively picked one user goal ("Message a friend on a mobile device using just the right hand") and broke it down into subgoals/tasks, which were realistic, concrete, and grounded in the context of use. By doing the tasks, the user would test out different features of the prototype, giving us the information we need to evaluate our design prototype against the three selected user requirements. The tasks are listed below with the explanation on how they were created and the corresponding user requirement(s) it attempts to measure.

(1) Switch to one-handed keyboard and adjust keyboard width
(2) Enter text "Cool, good"
(3) Enter "luck"

(4) You realized that you accidentally made a typo, i.e., entered “lyck”, fix the typo
(5) You changed your mind and wanted to enter “kick” instead, replace “luck” with “kick”
(6) Enter “!”
(7) Enter FACE WITH TEARS OF JOY emoji
(8) Send the message

To begin, the purpose of Task 1 was not to test any specific user requirements, but to make sure that the remaining tasks would be carried out using our one-handed keyboard design rather than the existing two-handed keyboard, since we would like to evaluate our new design not the existing one.

Task 2, 6, and 7 were originally designed to test user requirement 5, which involves the user entering punctuation and emoji efficiently. However, because we later realized that this requirement could not be evaluated in the current stage as it would involve measurement of speed, which works best with quantitative evaluation, these tasks were no longer considered to be testing this requirement. However, knowing how the participants perform on these tasks could still give us some initial insights on whether these new features of our design prototype would be promising and may potentially meet this requirement in the future when tested with quantitative methods.

Task 3 and 4 attempted to evaluate our prototype against user requirement 2: “User should not need to manually fix any results produced by the system’s keyboard assistive features (e.g., autocorrect) when entering text on their mobile device with one hand.” These two tasks imitate the scenario where a user accidentally made a slip resulting in a typo. This is a common scenario frequently found in our contextual inquiry. In this case, autocorrect would be counterproductive by automatically changing the word to another word which the user did not intend to type. Our design attempted to address this requirement by offering the user the freedom to conveniently choose whether to use autocorrect or not, and this user requirement would be considered fulfilled if the user were able to find and select the word they want to replace the typo with in the suggestion table, had they chosen to use this feature. This would indicate that our design met user requirement 2, at least to a certain extent given the current user testing with discounted prototype.

Task 5 attempted to evaluate our prototype against user requirement 3: “User should be able to remove any text entered on their mobile device using one hand without accidentally removing any other characters or words they did not intend.” This task involved the user to accurately delete one word on the prototype using one hand, which exactly parallels the user requirement itself. In order to consider our design to meet this requirement, the user would need to correctly utilize the new feature to delete the desired word. If the user used any existing features like repeatedly pressing the delete key or holding down the cursor to select the word, it would be considered not satisfying this user requirement. This is because we wanted to evaluate our new design, not the existing technology.

Lastly, all tasks except task 1, when taken together, attempted to evaluate our prototype against user requirement 4: “User should be able to interact with all keyboard elements using one hand without slips and without straining (overextending) their thumb (hand) or repositioning the mobile device.” By having the user interact with various elements at different locations on the keyboard, we can see whether the user explicitly mentioned or showed any actions that would imply any difficulties, such as long pauses or negative emotions, during the think-aloud. If none of them were present, we would consider our prototype to meet this user requirement, at least to some extent.

After creating the tasks for the users to perform, each team member then conveniently recruited a participant from the stakeholder group for the simplified user testing according to the inclusion/exclusion criteria we defined as a group: The participant 1) must be at least 18 years old, 2) right-handed or primarily use their phone with the right hand, and 3) preferably uses an iPhone (Note: the last criterion is preferred but not required).

Each team member then individually conducted simplified user testing with their participant following the protocol we designed collectively as a group (see specific details/steps of the protocol in the following section 7.2.3). Participants used the prototype, with its functionality being wizard-of-oz by the team member, to do each subgoal/task which all taken together accomplish the user goal, while performing a think-aloud by constantly talking and verbalizing their thoughts as they move through the user interface. The think-aloud allows investigators to understand not only what the participants were doing, but also what they were thinking, which may reveal additional insights on the prototype's issues, leading to a better evaluation of the prototype's usability and usefulness.

Each team member took notes during (and after) their simplified user testing, regarding the interpretations of the participant's actions and utterances during the think-aloud in relation to the prototype's usability and usefulness. Then, as a group, we consolidated the notes to produce high-level findings demonstrating our design's usability and whether it met the three user requirements we selected for the evaluation. With the analyses of the findings, we can improve our design by iterating on the prototype in the future.

7.2.3 Tasks and Procedures. To ensure the quality and consistency of our simplified user testing, each team member closely followed the protocol we created as a group, which was a bare-bones, bulleted outline regarding the process for conducting the simplified user testing, and the overall user goal and the corresponding subgoals/tasks which the participants had to perform. This section elaborated on the protocol, discussing it in detail while maintaining its core aspects, such that it accurately presented what each team member did to conduct the simplified user testing.

Each team member (referred to as the investigator in the remainder of this section) contacted their own participant online, by sending a message inviting them to participate in a user testing study on a design prototype. The message roughly described what the study was about, using a prototype to perform some tasks while narrating their thoughts; estimated duration, 30 minutes; required qualifications (inclusion/exclusion criteria defined early); and that the study had to be recorded (if the investigator needed to). After obtaining the participant's initial confirmation that they met the qualifications and were willing to do the testing, the participant and the investigator arranged an in-person meeting.

Upon arrival at the meeting and before starting the user testing, the participant was once again informed about the purpose of the study, to evaluate the design prototype's usability and usefulness, and their responsibilities, to use the prototype to execute a series of short tasks, while performing think-aloud. They were assured that their participation was fully voluntary as they may quit the study at any time. They were also told that the study would be recorded for further analysis (if needed by the investigator) and the data collected from the study would be completely anonymous and confidential. The participant then gave their verbal consent to participate in the user testing had they agreed to all of the conditions mentioned, which they did.

Starting the user testing, the investigator first clearly conveyed and explained the following instructions to the participant. Their overall goal was to "message a friend on a mobile device using just the right hand" (the same goal as heuristic evaluation). They would be given one subgoal/task at a time, which they need to execute on the prototype using just their right hand; completing all of the tasks would result in accomplishing the goal. While using the prototype, they had to perform think-aloud, or in other words, constantly talking and verbalizing their thoughts as they moved through the user interface. A short video was played to the participant to show them an example of think-aloud, for them to better understand what is expected (Link to video: https://www.nngroup.com/articles/thinking-aloud-demo-video/).

After the participant indicated that they understood the instructions clearly, the investigator gave the participant (the first frame of) the paper prototype and their first task (while starting the recording if the investigator needed to).

The paper prototype (built and discussed in the previous stage of user-centered design) had 16 frames, each resembling the size of a regular-model iPhone printed on paper and placed on a portable hard surface (with the same/similar size as the frame) such that the participant may hold it with their right hand just like how they would normally hold their phone. As the participant correctly interacted with the prototype (e.g., pressed the expected key on the current frame), the investigator would switch the current frame with the next frame, successfully giving functionality to the paper prototype with wizard-of-oz (the order/flow of frames was presented in Section 6.5).

As the participant tried to execute the first task on the prototype with their right hand while doing think-aloud, the investigator also took notes on the participant's errors, long stalls, confusion, unexpected paths, statements of distress, and unexpected events and use, indicated by both their actions and the utterances of their thoughts. If the investigator recorded the user testing session, they would also be able to watch the recording later to pick up the details they potentially missed the first time. The investigator also wrote down any questions that they wanted to ask the participant (e.g., Why did you do this action?); these questions were deferred until the end of the user testing (after all tasks were completed). The investigator did not interrupt the participant or ask any questions during the think-aloud because that may introduce extra cognitive load to the participant. Whenever the participant reduced in frequency of think-aloud, or in other words, they stopped talking but they continued to interact with the prototype, the investigator would prompt the participant to "please keep talking".

After the participant completed the current task, or when the participant had spent a long time on the current task yet still could not figure out how to complete it, the investigator would give them the next task to do. This process continued until all of the tasks had been given to and completed by the participant. The subgoals/tasks that the participant executed on the prototype were the following (listed in the order they were given to the participant).

(1) Switch to one-handed keyboard and adjust keyboard width
(2) Enter text "Cool, good"
(3) Enter "luck"
(4) You realized that you accidentally made a typo, i.e., entered "lyck", fix the typo
(5) You changed your mind and wanted to enter "kick" instead, replace "luck" with "kick"
(6) Enter "!"
(7) Enter FACE WITH TEARS OF JOY emoji
(8) Send the message

After all tasks were given to and executed by the participant, the investigator asked some follow-up questions they had during the think-aloud to the participant. In total, the simplified user testing study lasted about 30 minutes. Once the study was concluded, the investigator thanked the participants for their time.

7.2.4 Participants. A total of five participants were conveniently selected, one for each simplified user testing we conducted individually. Each team member sent out invitations to participate in the interview online to close friends who met the inclusion/exclusion criteria for the stakeholder group (defined in the Method section). No monetary incentives were provided for the completion of the study. The demographic information about the participants are listed in Table 10. For the results and analysis, each participant was assigned with an encoding from U1 to U5 respectively.

User encoding	Age	Gender	Race	Employment status	Dominant hand	Phone brand	Multi-lingual	Disability
U1	20	Man	Asian	Student	Right	iPhone	Yes	No
U2	23	Man	Asian	Student	Right	iPhone	Yes	No
U3	22	Man	Asian	Student	Right	iPhone	Yes	No
U4	22	Man	Asian	Employed part-time	Right	iPhone	Yes	No
U5	22	Woman	Asian	Employed full-time	Right	iPhone	Yes	No

Table 10. Simplified user testing participants' demographic information.

7.2.5 Results. This section discussed the summarized and consolidated notes written down by the investigators during their simplified user testing's think-aloud, including how well the participants performed in each task and when the design failed to help the participants meet the user requirements.

Task 1: Switch to one-handed keyboard and adjust keyboard width
All five participants struggled to complete this task. The participants either gave up on finding the one-hand keyboard or found the feature by randomly interacting with the keyboard. Most participants expressed their frustration via think-aloud by commenting that there was no indication of how to switch to the one-handed keyboard. Additionally, they demonstrated their confusion via random interactions on the keyboard (e.g., clicking on the language-switch key, dragging on the keyboard from left to right, going to settings, etc.) and eventually turning into long stalls.

Although these results do not imply the design failed to meet any specific user requirements, not being able to switch to our one-handed keyboard design would prevent the user from utilizing any of its features. Thus, in the next iteration, our team plans to add some indicators to help improve the discoverability of the switching to one-handed keyboard interaction to ease this process for users.

Task 2: Enter text "Cool, good"
All participants completed this task without experiencing any significant difficulties. Everyone was able to use our one-handed keyboard design to successfully enter the words. Nobody made anything about the smaller key size being an issue, except one participant (U5) who said that they tried to be more careful by typing slower because of the smaller key size. But this does not directly indicate any issues with the key sizes.

As for the punctuation (the comma), four out of five participants immediately found the new comma key located at the bottom right or the keyboard and proceeded to use the new comma key to enter the comma, with one participant (U4) explicitly mentioning that they really liked this new feature because it made the process of entering punctuation marks much simpler. Only one participant (U5) went for the punctuation keyboard out of habit, but they later found the new comma key. The participant mentioned that if he had seen the comma key earlier, he would have definitely used the comma key. Despite not being able to assess our fifth user requirement, we see that the addition of the punctuation keys was overall well-received and made text entry easier for the user.

These results also provide some partial evidence for meeting user requirement 4, as the participants did not struggle with accessing and interacting with any keyboard elements utilized in this task. This demonstrates some preliminary insight into how the design might be allowing users to comfortably interact with different elements on the keyboard, potentially addressing this user requirement. However, results from other tasks would still be needed, serving as more evidence, in order to have a more confident conclusion that our design prototype really meets this requirement.

Task 3: Enter “luck” (but accidentally typed “lyck”)
All five participants were able to complete this task without any issues.

Task 4: User realized that they made a typo (i.e., entered “lyck”), fix the typo
Four out of the five participants immediately resolved into clicking on the typo trying to correct it, because they mentioned how the dotted underline for the word indicated that they should click on it. Only one participant’s (U4) initial reaction was to immediately delete the word with the backspace key. For those who tapped on the underlined typo to open the list of nine possible words, they were able to easily locate and click on the correct word using one hand. Overall, all participants were able to fix the typo without any issues. Additionally, those who chose to use our new autocorrect feature were also able to find the desired word without having to change any results produced by the system/keyboard. Thus, by what we defined in section 7.2.2, user requirement 2 is met.

Task 5: Replace “luck” with “kick” by deleting the word “luck”
All participant’s initial action was to delete the word “luck” letter by letter. With the exception of only one participant (U3), everyone pressed the delete button four times to delete the word “luck”. Despite the delete key being elongated to show the user that swiping left on it could be used to delete an entire word, all participants did not know that this feature existed. Some participants (U2, U4) even found the blank keyboard to be confusing and thought that it was a glitch. Therefore, there was no conclusive evidence on whether user requirement 3 was satisfied because none of them was able to utilize the feature in our prototype to complete the task. Thus, we could not say that user requirement 3 was satisfied, and we have to improve this feature’s usability issue regarding its discoverability in the future.

Task 6: Enter “!”
There were mixed results for task 6. One participant found that it’s more time consuming to hold the bottom right key and swipe up when compared to just switching to the numbers keyboard. All participants initially did not know that this feature existed, and needed to examine the keyboard to search for the “!” button, as it was quite a small symbol on the keyboard. This is an issue that needs to be worked on in next iterations.

Task 7: Entering an emoji
Most participants found the new shifted emoji keyboard to be more comfortable and efficient to use. However, one participant (U2) found that the emoji search bar might not be useful. This indicates a potential area for improvement in the future.

Task 8: Send the message
All five participants were able to complete this task without any issues. All participants were able to recognize that there was still a “send” key located in the messaging app and was able to click on it to send the message. This meant that removing the return key from the keyboard did not interfere with the user’s ability to accomplish their goal of sending the message to a friend.

This concludes the results of simplified user testing. Further in-depth analysis on these results and those of heuristic evaluation are discussed in section 10.4.

8 FINAL DESIGN AND FUNCTIONAL HIGH-FIDELITY PROTOTYPE

8.1 User Requirements Tested Against

User Requirement 5: User should be able to locate and enter emojis, punctuations, and numbers on the mobile device with one hand faster than their default (i.e., current or existing) one-handed method of entry (U05-15, U05-16, U05-17, U05-37, U04-09, U04-18, U02-08, U03-06, U03-18, U03-19).

There appears to be a breakdown in entering non-letters with one hand, such as emojis (U05-15, U05-16, U05-17, U04-09, U04-18), punctuations, and numbers (U02-08, U03-06, U03-18, U03-19, U05-37). To enter these characters, the user had to first switch the keyboard (e.g., switching from the default letters keyboard to the emojis keyboard) by locating and pressing the keyboard-switch key, with which the user struggled, since this key is located at the edge of the keyboard (U05-15, U05-37, U03-18, U03-19). Then, the user had to find the desired characters on the new keyboard, which was also hard to do: For example, to select the desired emoji, the user had to consciously look at the keyboard rather than relying on muscle memory because the emojis keyboard has way more keys (i.e., emojis to choose from) compared to the letters keyboard and it also dynamically changes based on the frequently used emoji of the user (U05-16). A similar experience applies to the punctuations and numbers keyboard (U03-18, U05-37). Furthermore, because the user had to frequently enter these characters, almost in every message they sent, this process drastically slowed down the overall text entry speed (U05-17) and the user eventually stopped entering these characters due to its complexity (U05-37).

We selected this user requirement to evaluate our functional, high-fidelity prototype against because this is one of the two user requirements (with the other one being User Requirement 1) that could only be tested by quantitative user evaluations, which is the focus of the current assignment. Specifically, testing this requirement requires a study that manipulates keyboard design (our design vs. existing design) as the independent variable and measures typing speed as the dependent variable, then compares the mean typing speed on each keyboard design using statistical tests to determine the statistical significance and effect size. If the resulting difference showed our design with significantly faster typing speed than the existing design, then we would consider our prototype to meet this requirement. Thus, User Requirement 5 was selected for this assignment.

8.2 High-Fidelity Prototype

Before we built our high-fidelity prototype in Figma. We first had to identify which features of our design we needed to implement in order to evaluate the user requirement we selected. This is because a prototype should just be implementing some, rather than all features of a design, and a prototype with high-fidelity just means those implemented features have to be actually functional. Then, using those functional features should allow us to evaluate our design prototype against the user requirements, achieving the goal of the user study. Therefore, picking the right features to implement was very important. To do so, we had to think ahead a bit by considering the study design, specifically, the tasks which we would ask the participants to perform on our prototype (see Tasks and Procedures section for more info); we designed and created the high-fidelity prototype in Figma with all the functional features that allow the user to execute those tasks.

Based on the feedback we discovered from the qualitative user evaluation, we realized that our biggest issue with our low-fidelity prototype was the discoverability of the features, e.g., how to switch from the current keyboard to our new shifted keyboard, how to enter the alternative values of the punctuation keys, etc. Thus, to improve this, for each feature we implemented in the high-fidelity prototype, we also built a corresponding popover tutorial describing how to interact with them that would appear upon first opening the prototype in Figma. As such, the user would be able to easily discover the features and be able to use them effectively during the testing.

In the remainder of this section, we discuss the implemented features in detail and their corresponding interactions, along with screenshots of those features in our prototype.

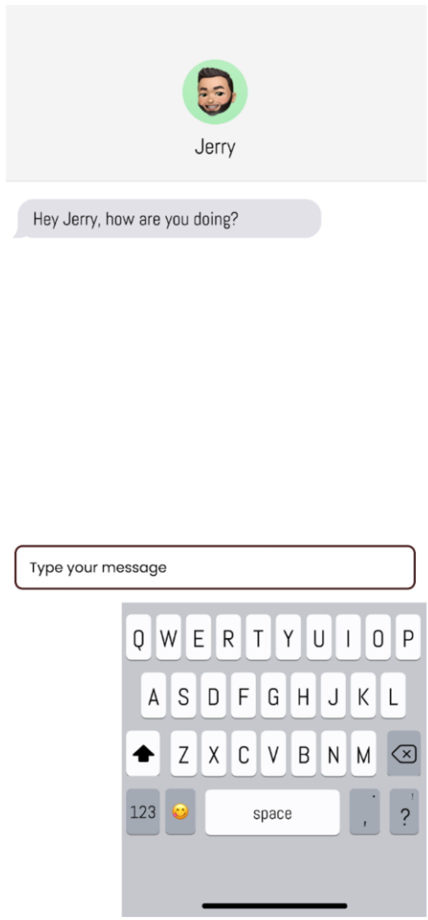

Fig. 38. Frame 1: Our Design for Punctuation Keyboard

This first screenshot shows the first frame (the one-handed letters keyboard) of our prototype created in Figma. All features on this frame are listed below.

- All of 26 letter keys on the keyboard.
  - Pressing on a letter key will enter the corresponding letter in the textbox above the keyboard.
  - Currently, all letters are in upper-case.
- Delete key:
  - Pressing on the delete key will delete everything the user has entered.
- Space key:
  - Pressing on the space key will enter a space in the textbox.
  - Holding down on the space key and then sliding left will switch the keyboard from this design (i.e., our one-handed keyboard design) to the current/existing keyboard design.
- 2 Punctuation keys (at the bottom right):

  - Pressing on these two keys will enter "," and "?" respectively.
  - Holding down on these two keys and then sliding up (or in any direction) will enter "." and "!" respectively.
- Emojis keyboard switch key (at the bottom left):
  - Pressing on this key will switch to the emojis keyboard (frame 2).
- Numbers keyboard switch key (at the bottom left):
  - Pressing on this key will switch to the numbers keyboard (frame 3).

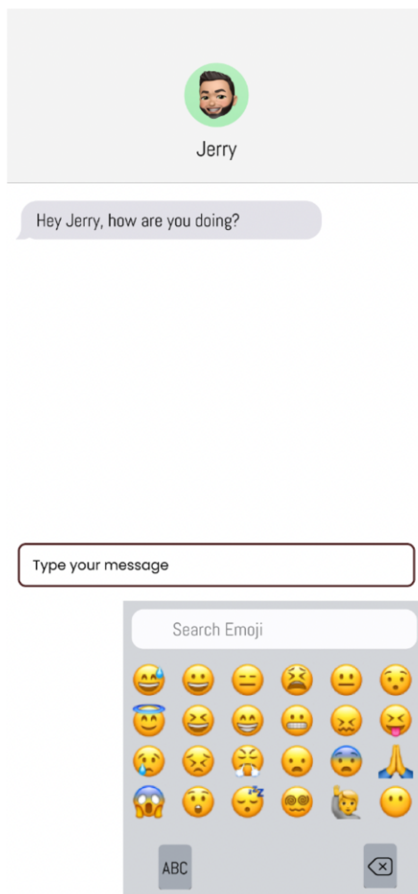

Fig. 39. Frame 2: Our Design for Emoji Keyboard

This screenshot shows the second frame (the one-handed emojis keyboard) of our prototype in Figma. The features on this frame are listed below.

- 4 emoji keys:
  - Pressing on any of the following 4 emoji keys will enter the corresponding emoji in the textbox:
    * Neutral Face Emoji
    * Grinning Squinting Face Emoji
    * Folded Hands Emoji

    * Face with Spiral Eyes Emoji
  - Remark: the reason why we only implemented these 4 emojis keys to be functional is because these are the only 4 emojis used in the tasks performed by the participants.
- Letters keyboard switch key (bottom left):
  - Pressing on this key will switch back to the letters keyboard (frame 1).
- Delete key (bottom right):
  - Pressing on this key will delete everything the user has entered in the textbox.

Fig. 40. Frame 3: Our Design for Number Keyboard

This screenshot shows the third frame (the one-handed numbers keyboard) of our prototype in Figma. The features on this frame are listed below.

- All 10 number keys (at the top of the keyboard):
  - Pressing on any number key will enter the corresponding number in the textbox.
- 4 punctuation keys (third row of the keyboard):
  - Pressing on the 4 punctuation keys will enter the corresponding punctuation:
    * , (comma)
    * . (period)

        * ? (question mark)
        * ! (exclamation mark)
- Delete key:
    – Pressing on this key will delete everything the user has entered in the textbox.
- Letters keyboard switch key (bottom left):
    – Pressing on this key will switch back to the letters keyboard (frame 1).

Our prototype also contains the following 3 frames, which imitate the current/existing keyboard design on an iPhone. These 3 frames have the same functionality as the previous frames respectively. The reason why we had to implement these 3 frames as well is because in our study, we needed to compare the typing speed on our design vs. on the existing/current design. But typing on Figma and typing on an actual phone may introduce confounds, since Figma has slight delays for each interaction, unlike a real final product of our keyboard. An interaction of all pages in our final prototype is also attached at the end of this section.

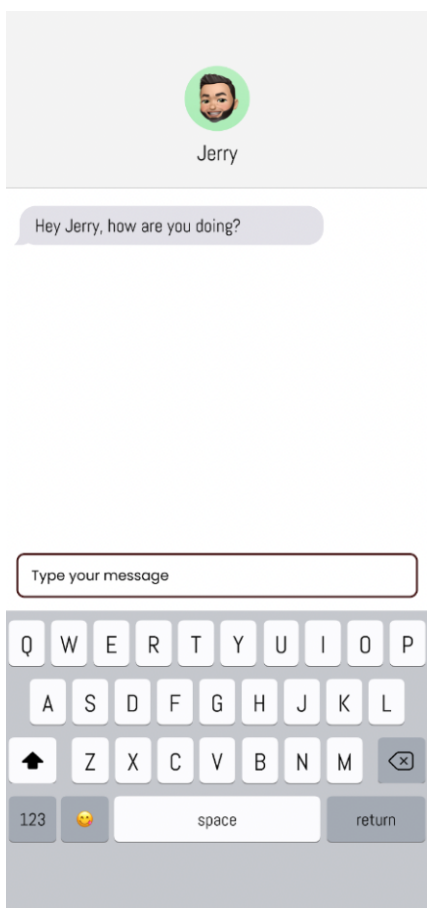

Fig. 41. Frame 4: Current/Existing Design Corresponding to Frame 1

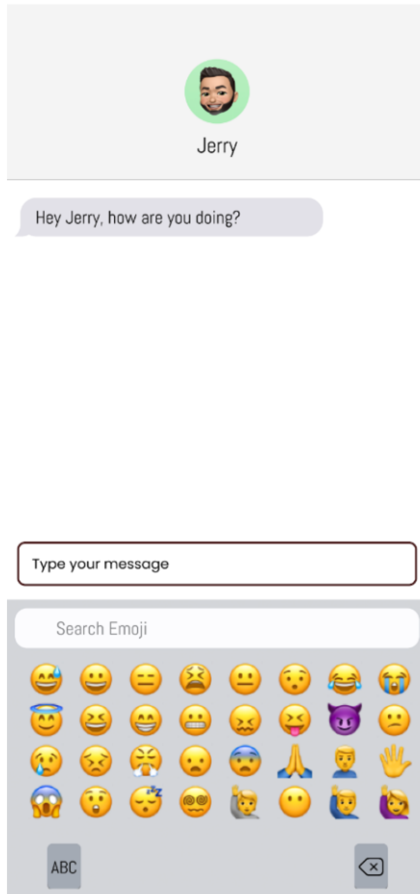

Fig. 42. Frame 5: Current/Existing Design Corresponding to Frame 2

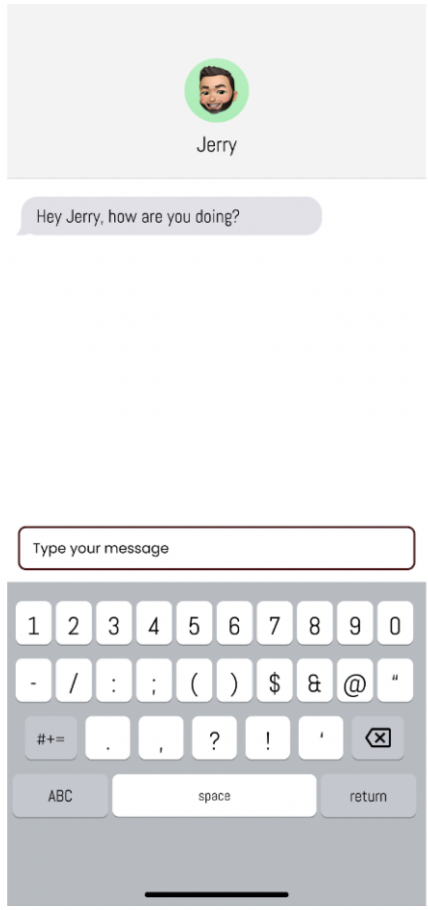

Fig. 43. Frame 6: Current/Existing Design Corresponding to Frame 3

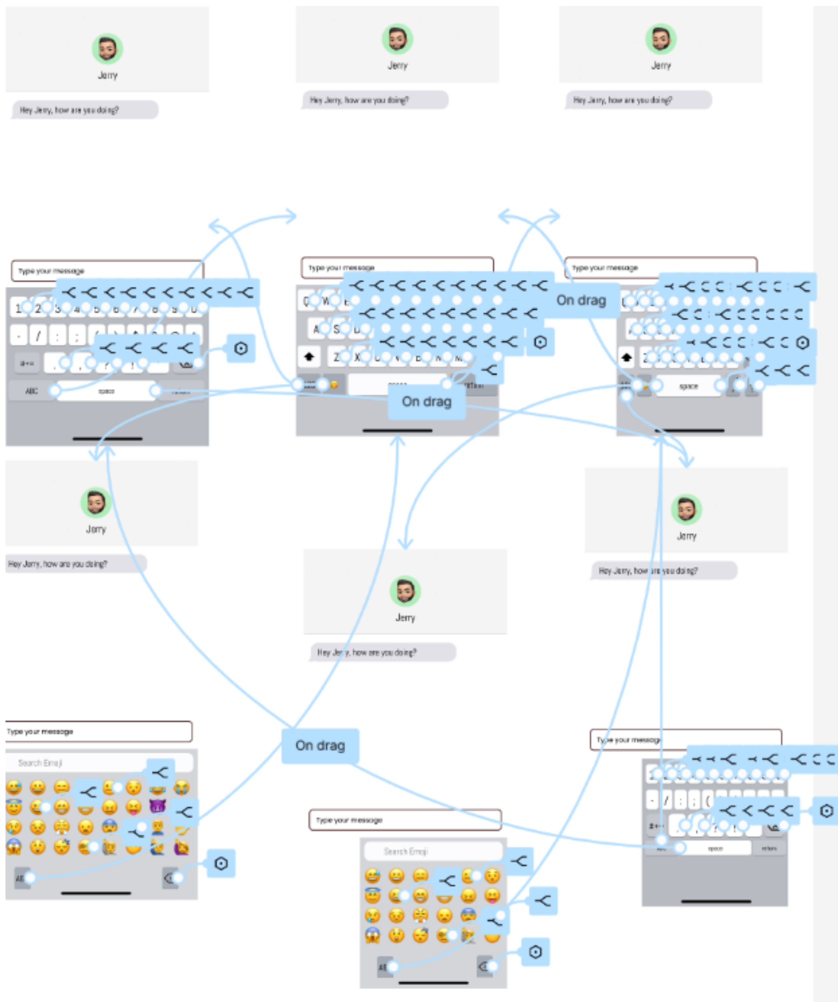

Fig. 44. Interactions with Different Frames

9 QUANTITATIVE USER EVALUATION

To ensure the quality and consistency of our user testing, we first created the study protocol as a group, defining its purpose, method, tasks/procedures, which each team member closely followed when conducting the study individually. This section elaborates on the protocol, discussing it in detail while maintaining its core aspects, such that it accurately presents what each team member did to conduct the user testing.

9.1 Purpose

The purpose of our study is to evaluate our new keyboard design against User Requirement 5, which we selected together. Specifically, we wanted to test whether entering punctuation, emojis, and numbers on our keyboard design with one hand is faster than entering them on the existing/current keyboard with one hand.

9.2 Method

First, we picked a user goal for the study that closely reflects User Requirement 5: to enter punctuation, emojis, and numbers using the keyboard with one hand. Based on this, we designed our study to employ multiple typing sessions, and within each session, participants would be performing tasks involving typing texts with different punctuation, emojis, or numbers. The specific sessions and tasks are described in the Tasks/Procedures section.

Next, we carefully designed the details of the study method. There was 1 independent variable, being *keyboard design*, with two conditions: our new one-handed keyboard design (i.e. intervention, or experimental), and the current/existing keyboard design (i.e. baseline, or control). There were 3 dependent variables, *time to enter punctuation*, *time to enter emojis*, and *time to enter numbers* (matching the three constructs in User Requirement 5). The reason why we split the time into three dependent variables is because we wanted to separately analyze *keyboard design x punctuation time*, *keyboard design x emojis time*, and *keyboard design x numbers time*. This increased level of specificity allowed us to pick up any effects of keyboard design to determine whether our design at least partially met User Requirement 5.

We decided to employ a paired or within-subject design for our study, such that each participant would be exposed to both conditions of the independent variable, i.e. using both keyboard designs to enter the texts. The motivation for this was that we did not have enough time and resources to recruit a large sample of participants, thus making a within-subject design work better in this case. Because of the nature of a within-subject design, we attempted to counterbalance the effect of the order of the condition to which the participants were exposed, i.e. whether they first typed on the our keyboard design or on the existing keyboard design, by randomly selecting one of the two keyboards designs for each participant to use first.

Since our study was a paired design, and we wanted to separately analyze *keyboard design x punctuation time*, *keyboard design x emojis time*, and *keyboard design x numbers time*, where keyboard design, being the independent variable, only had 2 conditions, we decided to use 3 paired Wilcoxon tests, with one for each interaction, according to Koji Yatani's table. Consequently, we defined the 3 null hypotheses, one for each test: For example, for the Wilcoxon test for the keyboard design x punctuation time interaction, the null hypothesis we defined is, there is *no significant difference* between the mean time to enter punctuation on the existing keyboard compared to our new keyboard when using one hand. We also defined 3 corresponding alternative hypotheses, one for each test: The alternative hypothesis for the same example is, the mean time to enter punctuation on our new keyboard is *significantly faster* than that of the existing keyboard, when using one hand.

After deciding on the statistical test, we used G*power to determine the minimum number of participants needed with parameters 0.8 for effect size, 0.05 for error probability (alpha), and 0.8 for power, which produced a minimum sample size of 12. Since we had 5 team members, each team member tried to recruit 3 participants, aiming for a total of 15. We also defined the inclusion/exclusion criteria for the participants, being 1) they must be at least 18 years old, 2) right-handed or primarily use their phone with the right hand, and 3) preferably use an iPhone (Note: the last criterion is preferred but not required).

With the protocol being completed, each team member then recruited 3 participants and conducted the user testing individually, while closely following the tasks/procedures described in the protocol (see next section). Afterwards, we collectively analyzed the timing results recorded by all team members by using the paired Wilcoxon tests to compute the statistical significance, to ultimately determine whether our keyboard design truly made entering punctuation, emojis, and numbers with one hand faster than the current design.

9.3 Tasks and Procedures

This section elaborates on the tasks the participants performed during the study and the procedure in which the study was conducted.

The study consisted of 2 sessions:

(1) Session 1, the practice session, where the goal was for participants to learn and adapt to the two Figma keyboards of the study, and the timing results did not count towards the statistical analysis later on, the results we recorded do not, and
(2) Session 2, the analysis session, where the timing results count towards the statistical analysis

Both sessions follow the exact same format. Within each session, participants were exposed to both conditions of our independent variable, i.e. they used both the existing keyboard and our new keyboard to enter text with one hand (which keyboard they started on was randomly determined for each participant to counterbalance the order effect); specifically, there were 3 pieces of text which the participants had to enter (on each keyboard) with their right hand only, one for punctuation, one for emojis, and one for numbers. Each piece of text consisted of 4 strings concatenated together in random order; the 4 strings for each piece of text are listed below.

- Session 1
  - Punctuation
    - HELLO,
    - HELLO.
    - HELLO?
    - HELLO!
  - Emojis
    - WORLD[Neutral Face Emoji]
    - WORLD[Grinning Squinting Face Emoji]
    - WORLD[Folded Hands Emoji]
    - WORLD[Face with Spiral Eyes Emoji]
  - Numbers
    - BREAD37
    - BREAD42
    - BREAD10
    - BREAD96
- Session 2
  - Punctuation
    - HORSE,
    - HORSE.
    - HORSE?
    - HORSE!
  - Emojis
    - WHALE[Neutral Face Emoji]
    - WHALE[Grinning Squinting Face Emoji]
    - WHALE[Folded Hands Emoji]
    - WHALE[Face with Spiral Eyes Emoji]
  - Numbers
    - CRUMB37
    - CRUMB42
    - CRUMB10
    - CRUMB96

For instance, the piece of text for punctuation which participants had to enter on the existing keyboard design in session 1 could be "HELLO?HELLO,HELLO.HELLO!", and it could be "HELLO.HELLO,HELLO!HELLO?" when they enter this on our keyboard design. The reason for why we randomized the order of the strings is because we want to prevent the effect of participants getting better from one design to the other, within one session. Concatenating the strings in a different order makes the piece of text appear somewhat novel to the participant when switching from one keyboard design to the other (within one session), but still keeping the actual content of the piece of text the same, so the comparison between timing results of one condition and the other using the paired t-test would still be valid. We also chose to switch the word part of the string between sessions to prevent the effect of participants getting better from Session 1 to Session 2. Since we did not plan to formally compare the timing results between sessions, having a different word would not affect the statistical tests.

Note that we used words to describe the emojis in the report here (because we couldn't figure out how to enter emojis in LaTex), but during the experiment, we showed the participants the actual emojis for clarity.

To sum up, at the beginning of each session, each participant was first randomly given a keyboard (i.e. either the existing keyboard or our new keyboard) they would type on first. Then, the participant used this keyboard with just their right hand to enter one piece of text for punctuation, one piece of text for emojis, and one piece of text for numbers, with each piece of text consisting of 4 strings listed above concatenated in a random order. The participant entered these 3 pieces of text again on the other keyboard with their right hand; each piece of text they entered had the same 4 strings as before, but the order of the strings were re-randomized and re-concatenated. The same process was repeated for both sessions. The detailed procedure which each team member closely followed is listed below.

Sentences enclosed by quotation marks are what each team member said to the participant word for word. Those that are not in quotations are instructions for investigators on what exactly needed to be done.

(1) Obtain consent
    - Explain the purpose of the study: "In this study, we are testing the effect of keyboard design on typing speed."
    - Briefly explain the participant's responsibilities: "You will be using two Figma keyboards, which you have to download on your phone, to enter a series of short texts with your right hand (thumb). Specifically, you will need to enter a total of 12 pieces of text and the time for each will be recorded."
        - 12 = (1 for punctuation + 1 for emoji + 1 for number) * 2 (keyboards per session) * 2 (sessions)
    - "The participation of the study is completely voluntary and you may quit the study at any time. The data collected from the study would be completely anonymous and confidential. Should you choose to participate in this study, please respond with 'yes', or 'no' otherwise."

(2) If the participant agreed to participate, then proceed with the following procedure. If not, then terminate the study and thank the participant for their time.
(3) First, ask the participant to download Figma on their phone and sign up.
(4) Ask the participant to open the link to the prototype and experiment with the keyboard.
(5) While they are experimenting with the keyboard, clearly explain to the participants all of the features in our prototype.

- i.e. the different possible interactions on each frame described in the previous Final Design and Functional High-Fidelity Prototype section.

(6) After the participant is familiar with every feature of the prototype, explain the following instructions for the participants in detail.
  - "I (the investigator) will first assign you one of the two keyboards in Figma that you have to type on first."
  - "Then, I will show you a piece of text and you have to type this text using the keyboard with your right hand."
  - "Please type (i.e. press on) one character at a time; make sure the current character has been entered (i.e. shown on the screen) before typing the next character."
  - "Please type at your normal speed but try your best to not make mistakes. If you do make a mistake, we have to restart."
  - Make sure the participant clearly understands the instructions.

(7) For both sessions, follow the following instructions exactly.
  (a) First, randomly pick one of the two keyboards for the participant to start on & clearly tell them which one.
    - Make sure the participant currently has that keyboard opened in Figma on their phone.

  (b) Shuffle (randomly) the order of the 4 strings for punctuation, concatenate them into one string (i.e. piece of text), and show it to the participant (either on paper or on a laptop).
    - Make sure the piece of text is in a very large font so the participant can easily see.
      – e.g. hello,hello?hello!hello.
    - Ask participants to look at this piece of text and ask if they have any questions

  (c) Start the timer as soon as the participant begins typing, and end the timer as soon as the participant finishes typing the last character in the text.
    - Record this time in the corresponding cell of the table.

  (d) If the participant made a mistake, ask them to stop, give the participant a 60 seconds break, restart this test by repeating steps (b) to (c) (i.e. need to re-shuffle the order of the 4 strings).

  (e) Repeat steps (b), (c), (d) for emojis, and then for numbers.
    - By the end of this step, the participant would have entered 3 pieces of texts and you would have recorded 3 different times.

  (f) Repeat steps (b), (c), (d), (e) for the other keyboard.
    - By the end of this step, the participant would have entered 3 more pieces of texts, with now a total of 6 pieces of texts entered and 6 different times recorded (by you).
    - This ends the first session.
    - Take a 2 minute break.

  (g) Repeat steps (b), (c), (d), (e), (f) for session 2.
    - Note: please use the strings we created for session 2.
    - Session 2 would result in 6 new pieces of texts entered by the participant with 6 different times recorded.

(8) Thank participants for their participation.

9.4 Participants

A total of 15 participants were conveniently selected. Each team member sent out study invitations online to close friends who met the inclusion/exclusion criteria for the stakeholder group (defined in the Method section) and the first 3 who replied became the participants. No monetary incentives were provided for the completion of the study. The demographic information about the participants are listed in the table 11 below.

Table 11. Quantitative User Evaluation Participants Demographics

Participant	Age	Gender	Race	Employment status	Dominant hand	Phone brand	Multi-lingual	Disability
J1	24	Man	Asian	Employed half-time	Right	iPhone	Yes	No
J2	21	Man	Asian	Student	Right	iPhone	Yes	No
J3	22	Man	Asian	Student	Right	iPhone	Yes	No
D1	20	Man	Indian	Student	Right	iPhone	Yes	No
D2	22	Man	Asian	Student	Right	iPhone	Yes	No
D3	21	Woman	Asian	Student	Right	iPhone	Yes	No
I1	22	Man	Asian	Student	Right	iPhone	Yes	No
I2	21	Man	Asian	Student	Right	iPhone	Yes	No
I3	22	Man	Asian	Student	Right	iPhone	Yes	No
F1	27	Man	Asian	Student	Right	iPhone	Yes	No
F2	33	Man	Asian	Employed Full-Time	Right	iPhone	Yes	No
F3	23	Man	Asian	Student	Right	iPhone	Yes	No
Y1	22	Man	Asian	Student	Right	iPhone	Yes	No
Y2	24	Man	Asian	Student	Right	iPhone	Yes	No
Y3	23	Woman	Asian	Employed Full-Time	Right	iPhone	Yes	No

9.5 Results

9.5.1 Data Description. We only used Session 2's data for conducting the statistical analysis, since the purpose of Session 1 was for practice, as stated in the Methods section. Each participant's data were recorded in a 2 by 3 table. The row category represents two conditions of the independent variable, "Our Keyboard Design" and "Existing Keyboard Design". The column category represents the three dependent variables, "Time for punctuation (s)", "Time for emoji (s)", and "Time for number (s)". An example table is listed below. The full results with all tables of every participant is in the Appendix.

Session 2 (counts towards statistical analysis)

	Time for punctuation (s)	Time for emoji (s)	Time for numbers (s)
Our Keyboard Design	17.38	28.04	20.26
Existing Keyboard Design	20.93	30.31	22.45

Fig. 45. Sample Data Entry

This table records the participant's time to enter the three pieces of text (consisting of punctuation, emojis, and numbers) in seconds, respectively on our new keyboard design and the existing keyboard design. Each participant had a different 2 by 3 table recording their data (see Appendix for all 15 participants' individual data).

9.5.2 Statistical Test. To reiterate what we mentioned in the Methods section, we conducted a separate statistical test for the time of each piece of text (punctuation, emoji, and numbers) entered by the participants, since they were independent of each other. To better test these quantitative data, we incorporated the knowledge of hypothesis testing into our evaluation. We defined our Null Hypothesis (H_0) as 'There is *no difference* between the mean time to enter X on our new keyboard and the mean time to enter punctuation on the original keyboard' and the Alternative Hypothesis (H_a) as 'The mean time to enter X on our new keyboard is *faster than* the mean time to enter punctuation on the original keyboard', where X can be either of: punctuation, emoji, and numbers.

For each of the 3 measures, we first drew a box plot to see the data distribution. The results are as follows.

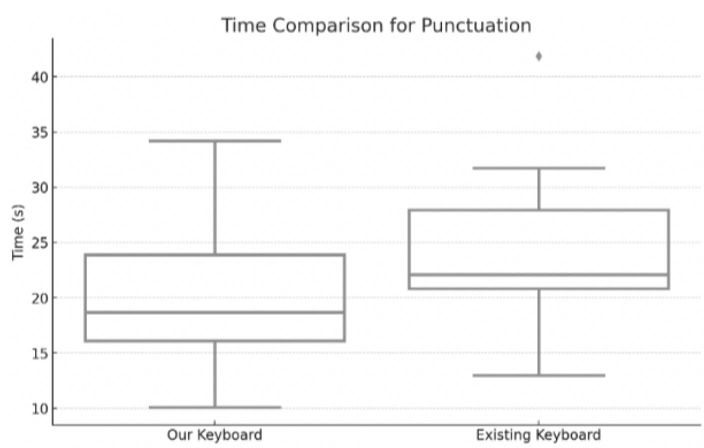

Fig. 46. Box Plot for Time to Enter Punctuation

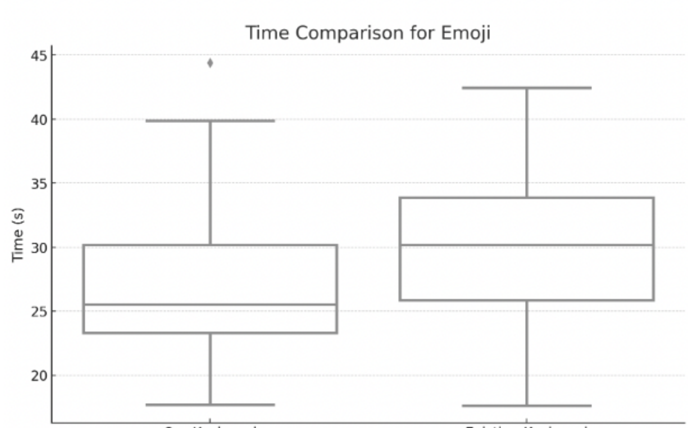

Fig. 47. Box Plot for Time to Enter Emojis

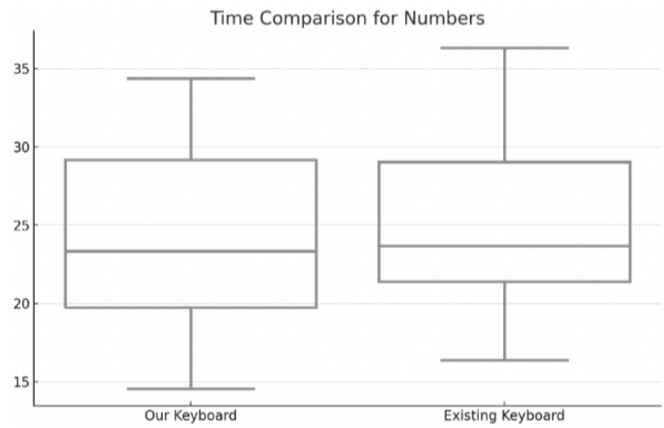

Fig. 48. Box Plot for Time to Enter Numbers

Based on the box plot, we can get some intuitive ideas about the results data. We can see that neither data follows normal distribution, since neither box is symmetrical with mean and median in the center. Most of the data does not follow normality and is right-skewed with few outliers. Furthermore, with only 15 participants in our study, it's not feasible to assert that our overall data exhibits normality. This limitation stems from the small sample size, which does not align with the Law of Large Numbers. Also, we noticed that our focus is to compare the time (seconds) the user performed on different keyboard designs, so this is a paired evaluation. Based on the information above, we used the Wilcoxon Single Rank Test for our statistical analysis. Based on the hypothesis definition, it's evident that we are dealing with a one-tailed paired test. This is because the alternative hypothesis is focused on assessing a 'faster' time (one-tailed), as opposed to a 'different' time (two-tailed).

We then calculated the Wilcoxon test using R and got t-values. We also transformed t-values into p-values to get a more readable result. Then we compared the t-value with 0.05 (95% confidence interval) to measure the statistical significance of our data.

Dependent Variable 1: Punctuation

Null Hypothesis: There is *no difference* between the mean time to enter punctuation on our new keyboard and the mean time to enter punctuation on the original keyboard.

Alternative Hypothesis: The mean time to enter punctuation on our new keyboard is *faster* than the mean time to enter punctuation on the original keyboard.

t-value: -2.92

p-value: 0.0056

Analysis: Participants were on average faster ($t(14) = -2.92$, $p = 0.0056$) with our newly designed keyboard (mean=20.01 seconds) than the original keyboard (mean=24.47 seconds) when entering punctuation. The p-value is 0.0056, which is less than the alpha level of 0.05. Therefore, we reject the null hypothesis for the punctuation dependent variable. This suggests that there is a strong statistically significant difference in typing speed between the two keyboard designs for punctuation, favoring "Our Keyboard Design".

Dependent Variable 2: Emojis

Null Hypothesis: There is *no difference* between the mean time to enter emoji on our new keyboard and the mean time to enter emoji on the original keyboard.

Alternative Hypothesis: The mean time to enter emoji on our new keyboard is *faster* than the mean time to enter emoji on the original keyboard.

t-value: -2.07

p-Value: 0.0285

Analysis: Participants were on average faster (t(14) = -2.07, p = 0.0285) with our newly designed keyboard (mean=27.74 seconds) than the original keyboard (mean=30.13 seconds) when entering emojis. The p-value is 0.0285, which is less than the alpha level of 0.05. Therefore, we reject the null hypothesis for the emojis dependent variable. This suggests that there is a statistically significant difference in typing speed between the two keyboard designs for emoji, favoring "Our Keyboard Design".

Dependent Variable 3: Numbers

Null Hypothesis: There is *no difference* between the mean time to enter number on our new keyboard and the mean time to enter number on the original keyboard.

Alternative Hypothesis: The mean time to enter numbers on our new keyboard is *faster* than the mean time to enter numbers on the original keyboard.

t-value: -0.66

p-Value: 0.2610

Analysis: Participants were not on average faster (t(14)=-0.66, p=0.2610) with our newly designed keyboard (mean=24.13 seconds) than the original keyboard (mean=24.94 seconds) in the test number section. The p-value is 0.2610, which is not less than the alpha level of 0.05. Therefore, we failed to reject the null hypothesis. This suggests that our keyboard design for numbers does not necessarily make typing numbers faster than the traditional keyboard.

10 DISCUSSION

10.1 Establishing Focus: Initial Survey

Before discussing the findings, it is important to note that we used a convenience sample for this survey, thus the results have low external validity. Consequently, all analysis and interpretations mentioned in Results and Discussion only apply to the specific sample of respondents of this survey. However, even without being able to generalize to a larger population, the results still do provide some initial insights into some existing issues with one-handed text entry on mobile devices.

Of the 27 participants who completed the survey, we found that one-handed text entry on smartphones is an existing interaction prevalent to users (as explained in Figure 6 and 7 in the Results section). Moreover, out of those who do type with one hand, users tend to find it less comfortable, harder, and slower to type with one hand compared to typing with both hands (as shown in Figure 9-11), implying that there are flaws in the current smartphone's keyboard for the interaction of one-handed text entry. This supports our original focus on one-handed text entry on mobile devices.

In attempting to narrow down the focus by examining specific areas for improvement on the existing smartphone keyboard, we found that users tend to make more typos and have a harder time to correct them when typing with one hand compared to with both hands (as shown in Figure 12 and 13). We also found that users find it more difficult to switch between letters and numbers, and also to switch the language of the keyboard with one hand than with both hands (as shown in Figure 17-19). These findings allow us to narrow the focus from "improving text entry on mobile devices with one hand", to "improving the ability to fix typos, switch between numbers and letters, and switch languages on a smartphone keyboard with one hand".

Furthermore, since the majority of participants use iPhone (Figure 5) and are right-handed (Figure 20), we decided to further narrow the focus to regarding just using the iPhone keyboard with the right hand. This may be beneficial to later stages in the UCD process since the iPhone has a uniform keyboard across all iPhone devices, unlike Android which has a wider variety of keyboards, thus limiting the potential confounding factors. Moreover, specifying the focus to be just the right hand also helps since the usability of a one-handed keyboard likely depends on the hand which the user uses to interact.

Thus, from the survey results, we successfully showed the validity of our focus and narrowed it down to "improving the iPhone keyboard to better support right-hand-only interaction, specifically on fixing typos, switching between letters/numbers, and switching languages."

10.2 Understanding Context of Use: Contextual Inquiry

After we conducted our contextual inquiry, our initial focus did not shift too much. However, we had to slightly broaden it to from "iPhone" and "right-handed interactions", to smartphone in general and left-handed interactions as well, because some of the team members were not able to find any participants using iPhone or right-handed.

However, given the results of the contextual inquiry and the following analyses, the breakdowns they have mostly experienced were still pertaining to keyboard operations we identified in our initial survey, including fixing typos, switching between letters/numbers and different languages. However, we were able to identify a few more areas where the breakdowns are coming from and also refine the existing areas.

Specifically, the breakdowns of one-handed text entry on mobile devices we compiled are related to the following factors: high frequency of typos, inconvenient keyboard layout, failure of keyboard assistive features, size of the mobile device, and the user's hand postures. In fact, some of these factors are also closely tied to one another. In fact, some of these factors are also closely tied to one another. A larger phone size and an inconvenient keyboard layout cause larger jumps between keys, and therefore give rise to uncomfortable hand postures while inputting text with a single hand. With an uncomfortable hand posture, the user is likely to commit text entry slips, have their precision and efficiency impaired, and it is even possible that they could drop their phone. Bad and unstable hand postures can also have a detrimental effect on fixing typos as well. Thus, with this new knowledge, we then refined our focus to be, "improving smartphone keyboard to better support one-handed-only interactions, specifically in entering commonly used characters, fixing typos, and usage of assistive keyboard features comfortably, accurately, and efficiently."

With our focus in mind, we can analyze our breakdowns with respect to our user requirements.

Our user requirement [1] is to tackle the breakdown with text entry slips as well as the issues of precision and efficiency that comes along with it. Making typos and fixing typos consist of a significant portion of our sequence and affinity diagrams, and is one of the main areas that the user was more frustrated with. We want to make sure that text entry with one hand is not at a great disadvantage when compared to entry with both hands. User requirement [2] attempts to mitigate the clunkiness and ineffectiveness of assistive features for typo correction. As many of our participants have reflected, these features are not as helpful as the user wanted them to be. These exist a situation where the user would rather leave the typos unfixed than attempt to fix them. Hence, we may need an approach that serves as an improvement or substitute to current assistive features to make fixing typos more convenient to conduct. However, user requirement [3] in particular seeks a solution to mitigate slips for a situation where the user forfeits the usage of assistive features and attempts typo correction manually.

On the other hand, user requirements [4] [5] [6] address the issue with key placement and hand straining which could also in turn give rise to issues related to typos already described above. When the human user interacts with a mobile device during text entry, the virtual keyboard serves as the sole interface that dictates the entire user experience. User requirement [4] essentially states that a bad keyboard layout or key placement can trigger a negative physical response from the user's hand in the form of an unstable and uncomfortable hand shape. User requirement [5] expanded on the observation that a more reasonable placement of the certain keys can also let the user locate keys faster and type more efficiently in addition to comfort. User requirement [6] is a follow-up to [5] and puts an emphasis on the physical side of the interaction which makes sure that the user should not have to hold their device in a compromised posture in general.

10.3 Initial Design and Low Fidelity Prototypes

Regarding the prototype for this iteration, our group was able to come up with diverse design ideas for addressing our user requirements.

The user requirement which we had the most discussion about is "comfortably reaching all key elements with one hand". Essentially speaking, this is deciding how we are designing our one-hand keyboard. The common element amongst all relevant designs is that we want to somehow move the edge keys closer to the user's thumb. Our approaches can be generally broken down into two types: directly moving the keys on the keyboard via some form of active user motion input, and relying on some other technology, such as the gyroscope in this context, to move the keys' positions with the help of gravity in a passive manner.

For the former type, we have two designs using the swiping motion, one being that we "squish" the entire keyboard towards a direction when the user swipes, and the other being that we "scroll" the keyboard towards one direction. The former design will display all keys at the same time regardless how far it is pushed, whereas the latter would have a portion of the keys fade out of the area of display on the keyboard, but the hidden portion would reappear if swiped back.

The gyroscope design, "the perspective keyboard", is to utilize the perspective phenomenon that metaphorically tilts the keyboard towards a direction when the mobile device is tilted by the user. The intent is to "tilt" edge keys towards the screen, as well as the proximity of the user's hand. The keys closer to the edge would gradually enlarge in size as well in correspondence with the perspective effect. This motion is controlled entirely by the user's wrist so there is no need for the user to conduct any gestures on the screen. However, this design would also have a portion of the keyboard hidden from the user when the device is tilted to a certain degree, and requires the device to be at a relatively more leveled configuration for the hidden portion to reappear again.

Eventually we chose the "squish" design since the "scroll" design is deemed logically simplistic and tedious to maneuver (constant scrolling between inputs), and the gyroscope design is too radical and sophisticated, and may introduce complications during evaluation. The "squish" is a more straightforward and effective solution that moves keys to a reachable area while giving the user access to all keys at the same time. However, the tradeoff is that the keys do become narrower as the keyboard is squished, thereby raising the likelihood of text entry slips, which inevitably goes against another user requirement regarding reducing user error. Yet we accept this tradeoff because this design gives the user control over how much the keyboard is shifted in the first place, and our requirements regarding typo fixing could potentially mitigate the importance of the requirement about user error.

There was also a debate as to whether functional switch keys (upper-case switch, language switch, and numeric/punctuation keyboard switch) should be placed in a designated region, separated from other keys, near the user's hand, or be shifted along with other keys on the one-hand keyboard. If we were to place these keys in a specific region, the tradeoff would be that although we can place them at wherever that is ergonomically comfortable for the user at our discretion, their positions may conflict with other app-specific UI elements, and may warrant some extent of overhaul of the original keyboard's layout, which is what we want to avoid for this iteration. We can avoid this problem altogether if we just let them shift together with the other keys, at the expense of not being able to place them at more ideal locations on the keyboard interface.

When tested against the problems of text entry described in the related work section, our design addresses some of these issues. One main concern in the designs proposed in the research papers was the high learning curve to use efficiently. This is due to the designs differing from the standard QWERTY keyboard. On the other hand, our design preserves most of the QWERTY layout that many people are familiar and comfortable with. Therefore, the learning curve for our design is much lower. The only exception is the addition of new punctuation keys on the bottom right that replaces the return key. However, none of our participants mentioned using this key during the contextual inquiry, so the new keys should not have much impact on the learning curve. In addition, our design can help prevent injuries described in the related work section. The shifting feature allows users to "squish" the keyboard so that all keys are within reach of their thumbs. Users will no longer need to strain their thumbs or wrists to compensate for their lack of reach, which is one of the leading causes of injuries.

10.4 Qualitative User Evaluation

We can conclude that the general ideas of our keyboard features are deemed promisingly usable. No user or usability expert expressed any severe concern with the core ideas of our design. However, there were still many loose ends that we did not tie up with this iteration.

The evaluations we conducted revealed that there were many small details that our paper prototype glossed over, which ended up becoming relatively serious usability issues. We were missing some signifiers and metaphors for our keyboard features which are currently lacking discoverability, and we were missing some features regarding user agency (missing a few "go back" buttons, etc).

The heuristics evaluation results show a variety of usability issues that can be boiled down to missing signifiers for specific features, lacking user agency due to missing keys in our design, and some features' effects being ambiguous and lacking refinement. Among these, the missing signifiers/indicators are generally ranked with the highest severity of '4', whereas the other ones mentioned usually yield a '3' or occasionally a '2'. These are the issues that we should address for our next iteration.

Our simplified user testing results also reflected these issues by instantiating these scenarios. All of our participants had trouble discovering some of our features due to missing signifiers and it complicates our ability to evaluate whether our features satisfy our requirements since they were not able to use it without us intervening. In the case where they did accidentally discover our feature (e.g. the delete key), they were confused by its visuals and its vague effects. Both of these two scenarios imply a great gap of execution and a great gap of evaluation respectively.

Features attempting to satisfy user requirements 2, 3 and 4 all had usability issues from the heuristics table. The participants were able to accomplish the task related to user requirement 2 despite having a usability issue (e.g. not able to exit from a certain state after the task is done), and consider requirement 2 to be partially satisfied. For user requirement 3, we consider it not satisfied because the feature is not discoverable, and therefore the users could not even initiate the task. For user requirement 4, the feature is also not discoverable but they were able to finish the tasks because those tasks are premised on the feature being activated already. Although we consider requirement 4 to be satisfied, we need to continue working on fixing the visibility issues in our next iteration.

For our next iteration, what we need to do is fixing the most blatant visibility/discoverability issues, and run an audit on the frame states of our prototype to make sure the user can return to the default state of the keyboard regardless which other state they are in. We also want to refine the effects of some features to make them more usable and less ambiguous as to what they are supposed to do.

10.5 High-Fidelity Prototype and Quantitative Statistical Analysis

Our study aimed to evaluate a new keyboard design, focusing on its efficiency in entering punctuation, emojis, and numbers with one hand, as compared to a standard keyboard design. This evaluation was essential to determine if our design could meet specific user requirements, particularly User Requirement 5. Our findings offer insightful conclusions about our high-fidelity prototype, designed and tested using Figma.

One of the challenges we faced was the usability of Figma for creating a functional and interactive keyboard prototype. To mitigate this, we employed a two-pass session strategy, focusing our analysis on data from the second session. This approach helped us filter out initial adaptation and learning curves, ensuring the results reflected users' true proficiency with each keyboard design.

Our statistical analysis revealed that our keyboard design significantly improved typing speed for punctuation. This outcome aligns with our expectations, as we had specifically revised and improved the punctuation aspect of the design. In contrast, the number entry did not show a statistically significant improvement in speed. This result was anticipated since we did not modify the number functionality; we only shifted the layout without altering its core design.

The most intriguing outcome was observed with the emoji keyboard. Although we did not intentionally modify its functionality—merely shrinking and shifting the entire keyboard layout—our design yielded a significant improvement in typing speed. Several factors could explain this unexpected result based on our discussion:

1. Adaptation to One-Handed Typing: Participants might have become more accustomed to the one-handed typing approach, which could have facilitated quicker emoji entry, even without specific functional changes to the emoji keyboard.

2. Reduced Emoji Set: The shifted keyboard design featured only 24 emojis compared to the original 32. This reduction likely simplified the selection process, minimizing visibility issues and decision-making time for users when choosing the correct emoji.

3. Ergonomic Benefits: The keyboard shift might have unintentionally resulted in a more ergonomic layout for one-handed use, especially for emoji selection, contributing to the observed increase in typing speed.

11 CONCLUSION

Our previous survey into one-handed text messaging on mobile devices provided foundational insights that set the stage for our subsequent research. Among the 27 participants, it was evident that one-handed typing is a prevalent method on smartphones. However, it came with its challenges: users found it less comfortable, more challenging, and slower than two-handed typing. This highlighted potential limitations in current smartphone keyboard designs and emphasized the practical importance of our research.

Building on this foundation, our detailed contextual inquiry further illuminated the differences of one-handed text messaging. We discovered that the existing keyboard layout often poses challenges, especially when users are multitasking. The challenges users face with current keyboard layouts, autocorrect features, and the act of switching between different keyboard modes have informed 6 specific user requirements. These requirements will be crucial in guiding the design of future mobile keyboards optimized for one-handed use.

Based on the specific requirements, we derived our initial design as the low-fidelity paper prototype. We also designed several design critiques and iteratively optimized our prototype. Our design mainly established and focused on the ‘squash’ and ‘scroll’ methods so users can comfortably reach all key elements with one hand. Our design also retains most of the traditional QWERTY keyboard layout, ensuring users face a minimum learning curve as well as solving main challenges in one-handed typing.

We conducted heuristic evaluation and simplified user testing to evaluate our prototype. Both heuristics evaluation and simplified user testing reveal nearly the same issues. Based on the result, we promise to say that there are nearly no severe usability issues with the core ideas but there are still some features that lack discoverability and visibility. Though all of our user requirements have some usability issues, we at least pass two-thirds of user requirements based on our discussion. We are going to fix those issues in our next iteration.

In the final step, we built our high-fidelity prototype and evaluated it with 15 participants. Based on the quantitative statistical analysis result, we can conclude that our new design for entering punctuation decreases the typing speed, which fits our expectations. For emoji and number typing, we can surprisingly see that users can still decrease typing speed by entering emojis, and this is probably based on our novel shifted keyboard design.

In conclusion, through the initial survey, contextual inquiries, low-fidelity paper prototype, interactive system evaluation, and high-fidelity prototype, we can build our novel keyboard design to increase one-handed typing efficiency. It not only provides insights and potential solutions into the current challenges users face but also guides as a robust foundation for future work. In the future, we are going to improve our high-fidelity prototype with more functionality, such as designing the number and emoji keyboard and creating the shifted keyboard for left-handed users. Besides, we are optimistic that future researchers and designers will leverage these findings, leading to innovations that cater to the evolving needs of one-handed mobile device users.

ACKNOWLEDGMENTS

We would like to thank Professor Nikola, GSI Chen Liang, and GSI Jaewoong for your help and support. Thank you for a great semester. Happy holidays.

REFERENCES

[1] James Clawson, Thad Starner, Daniel Kohlsdorf, David P. Quigley, and Scott Gilliland. 2014. Texting While Walking: An Evaluation of Mini-Qwerty Text Input While on-the-Go. In *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services* (Toronto, ON, Canada) *(MobileHCI '14)*. Association for Computing Machinery, New York, NY, USA, 339–348. https://doi.org/10.1145/2628363.2628408

[2] J. Diamond. 1997. The Curse of QWERTY. *Discover* (1997).

[3] D. Duncan. 1985. Typists are Turning to Seattle Man's Invention - Faster Keyboard Gains Converts 50 Years Late. *The Seattle Times* (1985).

[4] Damian Frej, Marek Jaśkiewicz, Miloš Poliak, and Zenon Zwierzewicz. 2022. Smartphone Use in Traffic: A Pilot Study on Pedestrian Behavior. *Applied Sciences* 12, 24 (2022). https://doi.org/10.3390/app122412676

[5] Lawrence Leung and Parham Aarabi. 2014. Mobile circular keyboards. In *2014 IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE)*. 1–4. https://doi.org/10.1109/CCECE.2014.6900941

[6] Edgar Matias, I. Scott MacKenzie, and William Buxton. 1996. One-Handed Touch Typing on a QWERTY Keyboard. *Hum.-Comput. Interact.* 11, 1 (mar 1996), 1–27. https://doi.org/10.1207/s15327051hci1101_1

[7] Jorge J. Nicolau, H. 2012. Touch typing using thumbs: understanding the effect of mobility and hand posture. *CHI '12* (2012). https://doi.org/10.1145/2207676.2208661

[8] Pierre Rimoldi Bixio Tarniceriu, Adrian Dillenbourg. 2012. Single-Handed Typing with Minimal Eye Commitment: A Text-Entry Study. (jan 2012). https://www.researchgate.net/publication/266483153_Single-Handed_Typing_with_Minimal_Eye_Commitment_A_Text-Entry_Study

[9] Dennis L. Caruana Christopher Wilhelm Maxwell Modrak Elizabeth C. Gardner William M. McLaughlin, Erin Cravez. 2023. An Epidemiological Study of Cell Phone-Related Injuries of the Hand and Wrist Reported in United States Emergency Departments From 2011 to 2020, Vol. 5. Journal of Hand Surgery Global Online, 184–188. https://doi.org/10.1016/j.jhsg.2022.11.009

[10] A. Oulasvirta X. Ren X. Jiang, J.P.P. Jokinen. 2022. Learning to type with mobile keyboards: findings with a randomized keyboard. *Comput. Hum. Behav.* 126 (Jan 2022). https://doi.org/10.1016/j.chb.2021.106992

[11] Kunpeng Zhang and Zhigang Deng. 2022. A Comparative Study on Single-Handed Keyboards on Large-Screen Mobile Devices. In *Proceedings of the 2022 International Conference on Advanced Visual Interfaces* (Frascati, Rome, Italy) *(AVI 2022)*. Association for Computing Machinery, New York, NY, USA, Article 4, 9 pages. https://doi.org/10.1145/3531073.3531075