WatchMeType: Typing in the air on a smart watch using the LEAP Motion

Jente Insing, Sander Ronde

[1] LIACS, Leiden University, Niels Bohrweg 1, Leiden, The Netherlands
jente.insing@gmail.com, awsdfgvhbjn@gmail.com

supervised by Jelle Talstra and Fons Verbeek.
jelletalstra@hotmail.com & f.j.verbeek@liacs.leidenuniv.nl.

Smartwatches are becoming more powerful and autonomous, but still have limited methods for text-entry. Often they make use of handwriting or voice dictation. But those methods can be very inconvenient or awkward to use in public. therefore a good text-entry method needs to be created. In this study we present our text-entry solution for a smartwatch using a motion tracking device, LEAP Motion, where users can interact with a new gesture-based keyboard by making simple mid-air gestures. After two user evaluations we have shown that our solution has potential, but the current techniques to track hand or finger movement has to be improved to make it more efficient and convenient to use.

1 Introduction

Smart watches are mostly used as a communication device, but most smartwatches do not have a (good) solution for text entry. The small screen provides limited screen space for touch input and a full QWERTY keyboard will become really small, making it impractical and inconvenient to use. Smartwatches like the Apple Watch or Android Wear 1.0 devices do not make use of a virtual keyboard, instead they rely on voice dictation (speech-to-text) to insert text, but this method can be awkward to use in public or is impossible to use in noisy areas. With smartwatches becoming more autonomous in the future, people should not have to rely on their phone as an input device to input text, therefore a new sort of text-input system on a smartwatch needs to be created. In this paper we will outline a new approach for text-input and answer our research question: *Does integrating a motion tracking and a gesture-based interface increase the speed of text input and make it more convenient for a Smartwatch user?* We will define the problem, describe our approach, solution and discuss the results of the evaluation of the prototype.

2 Background and problem definition

Different methods of text-input are used on the different smartwatch platforms on the market. We will list some of the most popular methods used by the two largest smartwatch operating system developers and take a look into alternative methods made by independent developers or researchers.

Methods of text input on smartwatches

The two biggest smartwatch Operating System developers active on the market are Apple and Google. Google introduced their smartwatch OS Android Wear 1.0 in 2014 and uses voice dictation as the main text-entry method.[1] In Android 2.0, which will be released early 2017, a touch based full QWERTY keyboard will be introduced. With a full QWERTY keyboard the keys will become very small, but by swiping instead of tapping, they make it easier to type. Another method introduced in this version of Android Wear is handwriting. With handwriting the user can write individual letters on the screen and this will be translated to text. Both methods included in Android Wear, are touch based and this means the hand of the user will block the screen, causing that users do not clearly see the feedback the screen provides. Apple's WatchOS introduced in 2015 also makes use of handwriting and voice dictation as the main text input method.[2] No touch based QWERTY based keyboard is provided.

In earlier research, alternative touch based approaches have been evaluated with regards to typing speed and ease of use. Zoomboard provides a full QWERTY keyboard, making the keys very small to fit the screen, but when tapping the screen, the keyboard will zoom into that specific area which will make the keys larger and a second tap will select the touched letter.[3] This method does not suggest words and two interactions are needed to select a letter, but still this method provides an average typing speed of 9.3 words per minute (WPM). In comparison, on a smartphone the average typing speed is approximately 25 - 35 words per minute.

Another research by Komninos and Dunlop provides a touch based six button keyboard with multiple letters on a single key using a standard alphabetical layout.[4] This method reduces the amount of interaction needed. Word suggestion helps to complete words to achieve a faster typing speed. With this method a typing speed of 8.1 WPM could be achieved.

The TouchOne keyboard is a smartwatch keyboard also based on touch.[5] The keys of the keyboard is circular and surround the screen which makes the buttons larger and it easier to type with. This solution also makes more room for the actual text. Still, because of the small screen it can make it inconvenient to use.

A research by Katsuragawa and Wallace and Lank shows instead of using touch a mid-air gestural keyboard by using a smartwatch to control other devices like a keyboard in an internet-of-things computer environment.[6] They compared a QWERTY keyboard with a Cirrin keyboard, making use use of a circular keyboard layout. With their implementation of the QWERTY keyboard a typing speed of 10WPM was achieved and with the Cirrin keyboard 6.4 WPM

We see that current smartwatch keyboard solutions are focusing mainly on touch. Based on the limitations of using touch based text input methods existing in actual consumers products as well in alternative methods that are explored in research, we believe a novel paradigm could be considered: mid-air gesture based interactions.

3 Swipe in the air to type

Initial design

To answer our research question we decided to develop a gesture based keyboard using the hand tracking motion controller LEAP Motion which can recognizes mid-air motion of the hand and individual fingers.[7]

When we created the first sketches and conducted research on existing smartwatch keyboards, we discovered the keyboard named TouchOne that looks similar to our first sketches and because we think this layout works best with mid-air gestures we decided to base our design on this keyboard and improve this interface to make it compatible for mid-air gesture interactions. The TouchOne keyboard layout already provides some design elements suitable for gesture interactions. For example: we figured out that when using mid-air gestures a centre is needed to reach all buttons and because the keyboard will work with hover-over and not with actual clicks it is not possible to hover over a button without activating it.

Our keyboard makes use of the predictive-text technology T9, which means “Text on 9 keys”, and the key layout of a dumb phone in order to reduce the amount of keys on the screen and maximize the size of the buttons (see figure 1a). According to Fitt’s law using larger buttons makes it easier and faster to select one.[8] Many people already have experience with the T9 layout which makes it easier to learn and to find the correct letter.

The layout is dived into 9 keys which surround the screen (see figure 1 & Appendix I). Letters A, B, C share a button, D, E and F another button etc. The middle area will

Figure 1. *Initial keyboard design and individual button lay-out*

be used to show the text entered and for touch based buttons that provide additional functionalities: A button to send text and an button for choosing special characters or numbers.

To type, the user selects letters by pointing the index finger in the air towards a button. A cursor on the screen will show the user where he is pointing to. Furthermore, we defined the following interactions:

1. When pointing with a finger to a key, a letter will be selected. The correct letter is predicted and hereby, words are suggested.
2. When swiping and hitting a key, the user pushes the key a little bit out of the screen to give feedback that the key is activated and to give an indication how hard the user has hit the key.
3. When the user pushes the key completely out of the screen, the letters on that specific key will be displayed as separate buttons. This enables the user to select a specific letter.
4. Swipe gestures with a flat hand provide additional functionalities: space (→, moving the hand from left to right), backspace (←, moving the hand from right to left), change suggested word (↓↑, moving the hand up and down to scroll through proposed words).

System implementation

To test the initial design we developed a web based prototype which could run on multiple platforms and devices. The web platform is supported on pretty much any platform so it should really help in our prototype. Of course when and if our system will transform from a prototype to an actual input method, a native implementation is preferred.

For motion tracking we decided to go with the LEAP motion. This is because the LEAP motion currently is one the best motion tracking device specifically for hands and not the whole body. As our idea is just a prototype that is built on the idea that in the future motion tracking will be better and more compact than it is now, which will make our keyboard more viable and more accurate.

The system works by analyzing the position of the user's finger movement. The user has to move their finger instead of moving the whole forearm. If a single finger is detected (and only a single finger), its direction is read. Given that the average person can only move their finger X degrees to any direction, the system then calculates how many degrees the user moved their finger and based on that sets the position of … on the screen (X/2 degrees to the right means the pointer is moved to 50% to the right of the center). This then allows the user to select one of the keys and the T9 system calculates the word the user probably meant.

4 Results and Evaluation

During this project, two user evaluations are conducted with people (n = 6, 4 male and 2 female μ = 25,2 years old) high educated and and a regular smartphone user, with no or almost no prior experience with a smartwatch. The prototype was tested on a computer (see figure 2), because we did not have the resources to test it on a smartwatch. After the first user evaluation the prototype was improved with the feedback we gathered. To test if our solution works well, usability specifications were made (see Appendix II & IV). Speed is the most important factor for our project to be considered as a success. Therefore we consider 8 words per minute as a good performance for inexperienced users. Second, it has to be easy to understand, it needs to have a small learning curve and the keyboard and the interactions have to be convenient. To test this, we will use the the System Usability scale (SUS) and want to achieve an average score of minimal 68.

Evaluation 1: Method

The first user evaluation is conducted after the first real working prototype was made. Three participants were asked to perform a few tasks. The first task was to make the participant explore the keyboard. The participant had to think out loud so we knew what the participant thought and which part he did not understand. After this exploration task, the user was given a task to type a sentence and afterwards a System Usability Scale (SUS) was filled in by the participant to measure the usability and learnability. (see table 1 & Appendix III) Also, a small non-directed interview was conducted to get some deeper insights about the experience (positive or negative) the user had.

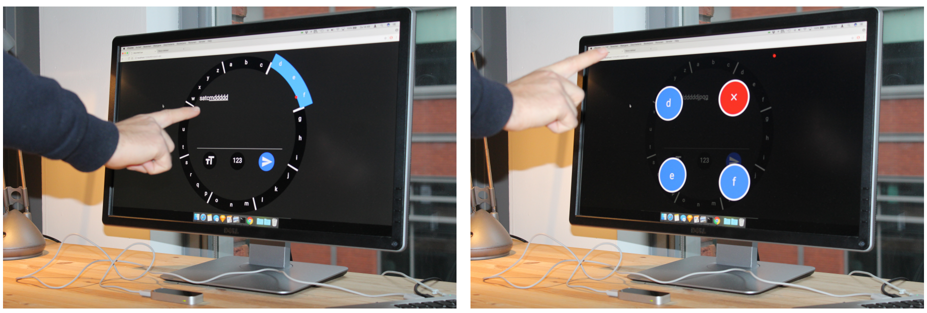

Figure 2. *User evaluation test setup*

Evaluation: Results

All participants liked to use the keyboard in combination with the LEAP Motion. The interface and the gestures the users had to learn were easy to understand, although they liked to see a tutorial with an explanation. They all could imagine this solution could be something they could find on a smartwatch in the future. On the downside the prototype was not very easy to use. The gestures were sometimes too sensitive. The individual letters showed up too fast when people did not want to use this function, which sometimes caused frustration. Also some users did not understand how to hide the individual button interface. Another problem we found was that the field of tracking of the LEAP Motion was too small. The participants wanted to move their whole hand through the air, but then their hand got out the tracking area. Because of the high sensitivity of movement of the LEAP Motion some of the users found it easier to not use the word-prediction mode, but instead they used the individual letters more to type. Hand swipes, to activate functions like space or backspace, did not work very well. The LEAP Motion did not always understand the type of gesture and therefore we used the Wizard of Oz method to simulate those gestures. Another major issue we found was that some participants had difficulties to point their finger to the the right of the screen with their right hand. After a while some of the participants experienced pain in their finger or wrist. The average SUS score was 51,66 (see table 1). This is below our expectations. Because of the high sensitivity, some bugs, and the error sensitivity of the LEAP Motion, the users did not think it was easy to use.

	P1	P2	P3	Average
Total score	50	47,5	57,5	51,66

Table 1. *outcome of the System Usability Scale (SUS). Score of 68 is average (P stands for participant.*

The typing speed was also not fast enough (see table 2). This can be explained, because people have to get used to the keyboard and the mid-air gestures and because of the problems mentioned earlier. The word per minute measurements are not completely valid, because the participants only had to type one sentence. More sentences or words have to be typed in a limited time frame measure the actual words per minute.

Participant	Time to complete task	WPM
P1	2:54 minutes	1.86
P2	3:04 minutes	1,76
P3	2:26 minutes	2,21
Average	2:48 minutes	1,92

Table 2. *Outcome of the time to complete task 2: Write "vanavond kan ik niet helaas" (P stands for participant.*

Changes after first evaluation
After this evaluation we have made some changes to the prototype. Because users mentioned that the movement of the cursor and some typing interactions were too sensitive, we have improved this and made the typing interactions less sensitive. Bugs in the word prediction software were fixed and a tutorial is now included, explaining some parts of the interface and gestures. Last, in the overlay with the individual letters a close button is included, making it more clear on how to close the overlay.

Evaluation 2: Method
After the improvements mentioned earlier a second user evaluation was conducted with three participants (see appendix IV). the participants had to perform the same tasks as the first user evaluation and fill in a SUS. Besides these tasks we measured the typing speed by conducting a speed test. The participant was given two minutes to type as many words as possible from a list of dutch words.

Evaluation 2: Results
During this user evaluation, we first showed the participants a tutorial, which revealed the functionality of the keyboard to the user. Most of the participants did understand the explanation in the tutorial immediately and they did not need extra information to get started with the system. The sensitivity of the typing interactions was improved. The participants were less likely to accidentally activate the individual buttons layover, which made typing with T9 much easier and faster.
In the individual letter layover it was unclear how to exactly close the layover with the new close button. Furthermore it was unclear that when the participant added an individual letter in for example the middle of a word the predictive text functionality deactivated. Therefore they did not know they could not change the word anymore by swiping up or down (↓↑) causing that they had to remove the whole word and start over. The average SUS score was 70 (see table 3). This is an increasement of 18,66 points. From this we can conclude that the overall experience and convenience of the prototype was much improved.

	P1	P2	P3	Average
Total score	62,5	80	67,5	70

Table 3. *outcome of the System Usability Scale (SUS). Score of 68 is average (P stands for participant).*

The typing speed in task 2 also improved greatly (see table 4). In comparison with the first user evaluation the average time to complete the sentence decreased with 75 seconds (44,65%).

Participant	Time to complete task	WPM
P1	1:58 minutes	2,7
P2	1:23 minutes	3,9
P3	1:20 minutes	2,7
Average	1:33 minutes	3,48

Table 4. *Outcome of the time to complete task 2: Write "vanavond kan ik niet helaas".*

When conducting the speed test the participants typed on average 66,3 characters in two minutes which is computed 6,63 words per minute (standard five characters per word, including a space). See figure 3 for the results. Also, we measured backspace usage to get an indication how many mistakes were made during the test.

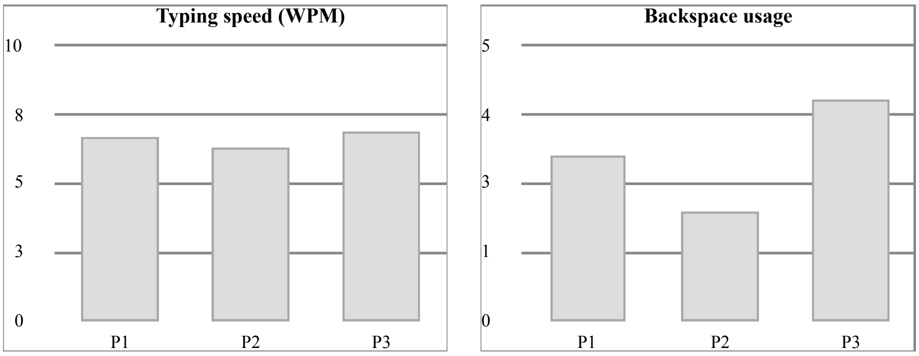

Figure 3. *Logged performance of participants during the speed test, split by words per minute and backspace usage.*

We see a big increase in the amount of words per minute during the speed test in comparison with task 2. This is mainly caused because they were more focused on speed and also because they got more experience during the test finding the correct letters.

5 Conclusion and Discussion

This study presented a new way of text entry using the motion tracking device LEAP Motion. After conducting two user evaluations we can conclude a motion-controlled keyboard can be a good solution for text entry on a smartwatch if motion controllers decrease in size and could be implemented in a smartwatch. However, the current techniques available, like the LEAP Motion, are too error prone and have to be improved in order to make the keyboard more convenient to use. Also our implementation of the LEAP API could be improved to make it work better. Some people mentioned during the experiments that their finger or wrist was hurting. We think this is caused by the limited hand tracking area of the LEAP Motion and the way people have to interact with it. A way to improve this is using mid-air gestures where the user also can move their forearm instead of performing mid-air gestures from the wrist where the user could not move their arm like we now had to implement, because of the constraints of the LEAP Motion. The participants in our first user evaluation did think the keyboard was not easy to use and the participants did not think they were not fully in control. With changing some of the interactions like adjusting the sensitivity we saw a big increasement in ease of use, control and most of all speed. The speed of typing was slower as expected, on avarage 6,63 wpm, and in comparison with other alternatives too slow. A reason for this is that some studies conducted multiple sessions with time spans of multiple hours. Therefore, when our participants have more experience we think a higher wpm could be achieved, but more research is needed. With the current prototype and techniques used, the requirements for learnability and ease of use were achieved.

6 Future Work

We believe motion tracking devices will improve in the future, making more possible and making it less error prone. Besides this, there is also room for improvements on the prototype itself. By implementing a smarter word-prediction system and a spellchecker we think we could improve the typing speed. Instead of using a standard alphabetical layout we also could study the effects of other layouts to improve speed and word predictions. Another point for future research is when a motion controller is embedded into a smartwatch. Now the LEAP Motion stood stationary on a desk, but when the arm of the user wearing a smartwatch moves it could be possible that typing would be less accurate and more difficult. Using sensors like an accelerometer and a gyroscope could solve this possible issue.

References

1. "Android Wear." *Android*. Google, n.d. Web. 16 Dec. 2016. <https://www.android.com/wear/>.
2. "Apple Watch - WatchOS." *Apple*. Apple, n.d. Web. 16 Dec. 2016. <http://www.apple.com/watchos/>.
3. Oney, Stephen, et al. "ZoomBoard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013.
4. Komninos, Andreas, and Mark Dunlop. "Text input on a smart watch." *IEEE Pervasive Computing* 13.4 (2014): 50-58.
5. Touchone Keyboard. TouchOne, n.d. Web. 16 Dec. 2016. <http://www.touchone.net/>.

6. Katsuragawa, Keiko, James R. Wallace, and Edward Lank. "Gestural Text Input Using a Smartwatch." Proceedings of the International Working Conference on Advanced Visual Interfaces. ACM, 2016.

7. "Leap Motion." Leap Motion. Leap Motion, n.d. Web. 16 Dec. 2016. <https://www.leapmotion.com/>.

8. Fitts, P. M. The information capacity of the human motor system in controlling the amplitude of movement. J. Experimental Psychology, 47, 6 (1954), 381-391.

Appendix I: Initial Design

Walkthrough

When the user opens the keyboard for the first time a walkthrough tutorial shows up to show the user how the gestures work. The user can swipe through the steps in the tutorial. the dots on the bottom of the screen indicates the amount of steps he has to take.

Keyboard

By swiping in the air with one finger the user is able to type. The letters are divided into eight buttons and there are 8 possible swipe directions possible. When user clicks on the big blue button the user sends the message. This will work with touch so the user can not mistakenly send the message when he accidently uses the wrong swipe gesture.

Swipe to type

When user swipes in the air to type with his finger, the button will be pushed in like a normal keyboard also does. The feedback will prevent that the user accidentally pushes the wrong key by giving the user some extra space to make mistakes. The word the user is typing is selected. When the user swipes with his whole hand down or up, he can change the word to the right one.
The user can add space between the words by swiping with his whole hand to the right and remove the letter by swiping to the left

Choosing a single letter

When the user swipes in the air to a key and hold his hand still at the end of the swipe gesture. the letters on the button will appear as individual buttons. The user can now select the individual letter he wants.

This functionality is needed, because the dictionary of the keyboard does not always know the right words, For example when the word is from another language or when typing a name.

Choosing a special character or number

When the user pinches in the air with his hand the keyboard will change to the special characters keyboard. Here the user can select one of the special character. By pinching back the keyboard will change again.

Appendix II: User evaluation 1

1. Research question and motivation

this document describes a plan for an usability test during the development of the WatchMeType smart watch keyboard. During this test we will focus on understandability, efficiency, satisfaction and learnability.

the test objectives are:

- Determine design inconsistencies and usability problems like navigation errors.
- Determine if the application meets the user requirements and contains all functionality the user needs to accomplish a task.
- Determine user satisfaction. What does the user think of the overall experience of the application.
- Determine if gestures are easy to learn.

Procedure

3 participants will take part in a usability test in a room at the University of Leiden. A laptop with the application connected to a Leap motion will be used to conduct the test. The participant's interaction with the website will be monitored and recorded via Quicktime screen capture and we will record the user via a camera pointed to the participant.

The facilitator will brief the participant before conducting the test. During the test the facilitator will read the participant's task one at a time and the participant will be asked to think aloud to better understand his/her mental model. The facilitator and data logger will observe the participant and log their behavior while interacting with the application. During the test, the time it takes to complete a task will be measured.

When the participant finished all the tasks he will be asked to fill in the SUS questionnaire to measure the user satisfaction. When finished, the participant will be interviewed by the facilitator to get his/her opinion and overall impression of the application.

Participants

For this test we will recruit 3 participants who represent our target user at Leiden University with the following characteristics:

- Smartphone- or smartwatch user.
- Age between 20 - 30.
- Mix of men and women.
- Mix of educational and technical background

Test methods

Observing

We will test the understandability by using observational techniques. The participant will be asked to perform some tasks. The user has to think aloud in order to understand what the user thinks of the product. The test will be recorded and analyzed to measure the user performance.

Non-directed Interviews

After the participant has taken the test, we will ask some questions about their experience with the product. If problems occurred or functionality was not clear during the observation the participant can explain it in more detail.

System usability scale

To measure the user satisfaction the System usability scale (SUS) will be used. We can compare the SUS scores to our second test to measure if the user satisfaction is improved.

Scenario and Tasks

The usability tasks will be derived from scenarios to set a context for the participant to interact with the application. The participant starts with an easy task and will become more difficult during the test. The scenarios and tasks used are:

Task 1

Explore and play with the application and think out loud.

Scenario 1

You have a smartwatch and a friend of you just messaged you. You do not have a smartphone with you to answer on his message so you decide to use your smart watch instead. Your friend asks you if you want to drink something tonight in the centre of Leiden. You realize you do not have time tonight and start messaging him.

Task 2

Type “ik kan vanavond niet helaas?” and send the message.

Measurements

To evaluate the participant's performance the following metrics will be collected.

Objective	Measures
Learnability	Count amount of user errors
Efficiency	Time to complete a task
Satisfaction measures	Measure SUS score.

Results

When all participants have completed the test, all data will be compiled and all usability problems encountered during the test will be documented and recommendations for further development will be provided to meet the user’s requirements. Because the prototype did not work very well with using hand gestures to add space or remove a lesson, the participant had to say the intended gesture and the facilitator would click on the right button on an ipad.

Understandability

Overall the participants understanded the interface immediately. The keyboard layout was clear and recognizable with the numeric keyboard they used on their dumbphone. Hand gestures were logical and no errors were made using them. Some participants could not find an exit when in the individual letter modal.

Participant	**understanding gestures**	**understanding interface**
Participant 1	0 errors made	1 error made
Participant 2	0 errors made	4 errors made
Participant 3	2 errors made	5 errors made

Efficiency

Outcome of the time to complete task 2.

Participant	**Time to complete task 2**
Participant 1	2:54 minutes to complete
Participant 2	3:04 minutes to complete
Participant 3	2:26 minutes to complete

Satisfaction.

To measure satisfaction the System usability scale was used.

	Participant 1	**Participant 2**	**Participant 3**	**Average score**
Total score	50	47,5	57,5	51,66

Participant 1: observing and interview results

- Does understand the keyboard layout. Its recognizable with numeric keyboard and understands T9.
- Gestures can recalled easily and are logical.
- Can find the letters easily.
- Gestures are really sensitive. Modal with Individual letters will show up too fast.
- Does not understand how to hide the individual letters modal.
- Hand Swipe gestures do not work very well. When trying to add a space, the system still measures the finger, causing that the system selects a letter or the individual letters show up.
- Participant comments that her finger is going to hurt when she moves it from the body away. (right hand pointing to the right on the screen.
- Word prediction does not work very well, because of the hand swipes not working very well, it is easier to type with the individual letters.

SUS

1 = 2, 2 = 2, 3 = 1, 4 = 1, 5 = 3, 6 = 4, 7 = 4, 8 = 4, 9 = 3, 10 = 2, Total score = 50

Participant 2: observing and interview results

- directly finds the individual buttons and likes that he can to push them outside the screen
- Does not understand T9 immediately. Needs some help with it.
- Is missing a tutorial in the beginning
- Gestures are easy to remember, Says that the gesture for space (swipe to right with hand) is more difficult than a swipe down, and because space is often used actions this will be more logical.
- Gestures are too sensitive. Modal with individual letters will show up too fast.
- Individual buttons are too far away, needs to put some effort to hit them.
- Participant think it is too difficult to go to the right with his finger when typing. To the left is also not as convenient as moving the finger down or up.

SUS results

1 = 2, 2 = 2, 3 = 2, 4 = 2, 5 = 3, 6 = 3, 7 = 4, 8 = 4, 9 = 2, 10 = 3, Total score = 47,5

Participant 3: observing and interview results

- Is missing a tutorial explaining how it works.
- Mentions the gestures are too sensitive.
- Does not know how to close the individual letter modal.
- Does not directly knows how to get to the individual button layout. Thinks he has to touch a key twice to let it appear. Figures out by himself how to do it.
- Mentions pain in finger. Buttons on the right are difficult to touch.
- mentions that if it works better, it would be a nice new way of typing. Could imagine this on a smartwatch.

SUS results

1 = 2, 2 = 2, 3 = 4, 4 = 2, 5 = 4, 6 = 2, 7 = 2, 8 = 3, 9 = 4, 10 = 3, Total score = 57,5

Appendix III: System Usability Scale (SUS)

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Appendix IV: User evaluation 2

1. Research question and motivation

this document describes a plan for an usability test during the development of the WatchMeType smart watch keyboard. During this test we will focus on understandability, efficiency, satisfaction and learnability.

the test objectives are:

- Determine design inconsistencies and usability problems like navigation errors.
- Determine if the application meets the user requirements and contains all functionality the user needs to accomplish a task.
- Determine user satisfaction. What does the user think of the overall experience of the application.
- Determine if gestures are easy to learn.
- Determine words per minute.

Procedure

3 participants will take part in a usability test in a room at the University of Leiden. A laptop with the application connected to a Leap motion will be used to conduct the test. The participant's interaction with the website will be monitored and recorded via Quicktime screen capture and we will record the user via a camera pointed to the participant.

The facilitator will brief the participant before conducting the test. During the test the facilitator will read the participant's task one at a time and the participant will be asked to think aloud to better understand his/her mental model. The facilitator and data logger will observe the participant and log their behavior while interacting with the application. During the test, the time it takes to complete a task will be measured.

When the participant finished all the tasks he will be asked to fill in the SUS questionnaire to measure the user satisfaction. When finished, the participant will be interviewed by the facilitator to get his/her opinion and overall impression of the application.

Participants

For this test we will recruit 3 participants who represent our target user at Leiden University with the following characteristics:

- Smartphone- or smartwatch user.
- Age between 20 - 30.
- Mix of men and women.
- Mix of educational and technical background

Test methods

Observing

We will test the understandability by using observational techniques. The participant will be asked to perform some tasks. The user has to think aloud in order to understand what the user thinks of the product. The test will be recorded and analyzed to measure the user performance.

Non-directed Interviews

After the participant has taken the test, we will ask some questions about their experience with the product. If problems occurred or functionality was not clear during the observation the participant can explain it in more detail.

System usability scale

To measure the user satisfaction the System usability scale (SUS) will be used. We can compare the SUS scores to our second test to measure if the user satisfaction is improved.

Tasks

The usability tasks will be derived from scenarios to set a context for the participant to interact with the application. The participant starts with an easy task and will become more difficult during the test. The scenarios and tasks used are:

Task 1

Explore and play with the application and think out loud.

Scenario

You have a smartwatch and a friend of you just messaged you. You do not have a smartphone with you to answer on his message so you decide to use your smartwatch instead. Your friend asks you if you want to drink something tonight in the centre of Leiden. You realize you do not have time tonight and start messaging him.

Task 2

Type “ik kan vanavond niet helaas” and send the message.

Task 3

The participant has to type a text to measure Words Per Minute (WPM). The amount of characters and the time will be measured to get an indication of the average typing speed. The participant gets 2 minutes to type as many words as possible. The words the participant has to type are:

Papa bos sneeuw herinneren anders noorden prettige kussen tussen ochtend koning dronken straat zoen missen reis programma buiten spreek bank krant genieten gebeuren voorstel keuken avond missen vakantie boerderij ook les buur contact kunnen dragen ander dochter knap ver links iemand man blijven programma heeft drie rechts sorry drinken gaat vertalen

Measurements

To evaluate the participant's performance the following metrics will be collected.

Objective	Measures
Learnability	Count amount of user errors
Efficiency	Time to complete a task and speed test
Satisfaction measures	Measure SUS score.

Results

When all participants have completed the test, all data will be compiled and all usability problems encountered during the test will be documented and recommendations for further development will be provided to meet the user's requirements. Because the prototype did not work very well with using hand gestures to add space or remove a lesson, the participant had to say the intended gesture and the facilitator would click on the right button on an ipad.

Understandability

Adding a tutorial helped the participants understand the gestures and interface better. People did not always know which part of the interface is touch based and which part of the interface is gesture based. Also some participant did not always know where to find the individual letter overlay. The close button in this overlay looked the same as the individual letters with a color difference, but instead of only hitting the close button they had to swipe over the button. This was not very clear to the participants.

Participant	understanding gestures	understanding interface
Participant 1	0 errors made	2 errors made
Participant 2	0 errors made	1 errors made
Participant 3	0 erros made	3 errors made

Efficiency

Outcome of the time to complete task 2.

Participant	Time to complete task 2
Participant 1	1:58 minutes to complete
Participant 2	1:23 minutes to complete
Participant 3	1:20 minutes to complete

Outcome of task 3. The amount of words is calculated by counting the amount of characters typed divided by the value 5 (Average word length).

Participant	Amount of characters	Amount of words	Words per minute
Participant 1	67	13,4	6,7
Participant 2	63	12,6	6,3
Participant 3	69	13,7	6,9

Satisfaction.

To measure satisfaction the System usability scale was used.

	Participant 1	Participant 2	Participant 3	Average score
Total score	62,5	80	67,5	70

Participant 1: observing and interview results

- Tutorial is clear. But does not know that she has to touch the screen to swipe.
- Does not understand she has to swipe over the close button. Thinks that it is inconsistent.
- found a bug where she activates buttons underneath the individual letter overlay. This should not happen.
- The sensitivity of the keys to get into the individual letter overlay is improved. Makes it harder to accidentally activate the overlay.
- When she has chosen an individual letter she does not see enough feedback that she hit the correct letter. Suggests an animation or something.
- It is possible to activate two letter keys. When that happens she does not know which letter she has chosen.
- Because the small vocabulaire of the keyboard, it is very difficult to see where you made a typing mistake.
- If you choose an individual letter, text prediction will be deactivate. User does not know that

SUS

1=2 2=2 3=2 4=2 5=2 6=1 7=5 8=4 9=4 10=1, Total score = 62,5

Participant 2: observing and interview results

- Does not understand she has to swipe over the close button.
- Does not know how to get to the individual letter overlay. Does happen sometimes accidentally.
- Predictive text is deactivated when she hits an individual letter. Does not know that. Thinks that she made a mistake.
- found a bug where the special characters are not correctly displayed.
- Mentions she does like the way of interacting with the keyboard.
- Thinks she has to swipe over the send/special characters/ or capital letters buttons. Does not know it’s touch based.

SUS

1=4 2=2 3=4 4=1 5=5 6=2 7=4 8=2 9=4 10=2, Total score = 80.

Participant 3: observing and interview results

- Tutorial is clear, Does directly understand how the keyboard works.
- Does directly understand how to get to the individual letters. Likes the animation of pushing the button out of the screen.
- Can not directly find the special characters. But thinks that it is underneath the 123 button.
- Likes to see letters with special characters on it. like é or â.
- Notices text prediction with individual letters does not work.

- Suggests a list when swiping through predicted words. it is difficult to see now if there is an word predicted or that the word is incorrectly spelled.
- He thinks that he is under control learned it fast. Notices that there is almost no learning curve.
-

SUS

1=2 2=2 3=4 4=1 5=4 6=4 7=4 8=2 9=3 10=1, Total score = 67,5.

Conclusion

When conducting the second user evaluation, we first showed the participants a walktrough. Here we revealed the functionality of the keyboard to the user. Most of the participants did understand the explanation in the walktrough immediately and they did not need extra information to get started with the system. The sensitivity of the typing interactions was improved. The participants were less likely to accidentally activate the individual buttons layover, which made typing with T9 easier and faster.
In the individual letter layover it was unclear how to exactly close the layover with the new close button. Furthermore it was unclear that when the participant added an individual letter in for example the middle of a word the predictive text functionality deactivated. Therefore they did not know they could not change the word anymore by swiping up or down (↓↑) causing that they had to remove the word and start over. The average SUS score was 70 (see table 3). This is an increasement of 18,66 points. With this second user evaluation we can conclude that the overall experience and convenience of the prototype was much improved, but there are some inconsistencies and improvements we still have to make. When conducting the speed test the participants typed on average 66,3 characters in two minutes which is computed 6,63 words per minute. This is below our expectations but when we look at the small amount of time they invested to learn to interact with the keyboard we think it is quite good.