Comparison between Text-input and Speech-input User Interfaces for Health Centers - Term Project Final Report

Eugene O. Casmin - 2528933, Abdullah I. S. Sulayfani - 2542603

Resumen— **With more and more tasks in our daily lives being computerised, human-computer interaction is a leading research topic among emerging trends owing to the need for seamless operations as well as growing calls for more inclusion of users who are at a disadvantage. In light of this, there is increasing need for the production of technologies with improved interfaces everyday. Our term project is one such example designed to be used in a healthcare environment. This study follows the design and implementation of a speech input based UI as well as the subsequent usability evaluation of the aforementioned UI juxtaposed with the current text input based UI. The findings are presented in the form of a heuristic evaluation reports by the experts as well as responses obtained from interviewing users after interacting with the system. The goal is to draw attention to the pros and cons of the use of speech-input user interfaces in performing daily operations within the health sector.**

Palabras clave— **human-computer interaction, text-input UI, speech-input UI, UI evaluation, health sector.**

I. Introduction/Motivation/Problem Statement

TILL this point in the course, we have seen how pervasive interfaces are in daily lives of their respective users. Additionally, we have been made privy to different types of interfaces with these differences ranging from appearances to the mode of interaction implemented. There are various reasons why an interface designer would choose one mode of interaction over another during the design phase. Our project seeks to explore such a process.

As we have mentioned in our abstract, processes are becoming more and more computerised worldwide. This computerisation spills over to one of the most important sectors of any community; healthcare. Healthcare officers (a group which comprises numerous professions from doctors to simple office clerks at the health center) need to interact with healthcare systems in a manner that is as seamless as possible to ensure that the data handling process does not consume more time than is necessary.

If you speak to a doctor or their assistant at the health center, chances are they will tell you that a good chunk of their work is filling forms about patient personal information or filling forms about data pertaining a patient's progress for use while making a diagnosis or filling forms requesting office supplies; I am fairly certain that you sense a pattern. Through out all this data entry, the users need to interact with a system that has an interface. It is therefore of paramount importance that these interfaces are made as simple to use as possible.

What if we could investigate which method of interaction yields the best net results?

Our project proposes a contrast and comparison between a system/UI for healthcare officials that implements traditional data entry methods or interaction methods for filling the aforementioned forms and a improved system that implements speech-based input as an interaction method.

II. Related Work

This section explores articles and resources whose inspiration we have sought in the literature review part of our proposed project. Each paragraph elaborates on a resource as well as the reason why we picked it.

The first resource we scrutinised was [1] which elaborates what exactly is meant by the term "Speech Input". Furthermore, the website explains the origins of speech input in the form of speech-to-text interfaces that have been implemented in the past to assist users with physical and/or cognitive impairments to interact with computers. The article also mentions some examples of commonly used speech input, citing Apple's Siri, Amazon's Alexa and Echo, and Google Assistant.

The second resource we considered was [2] which explored the pros and cons of speech-input based user interfaces. It is important that we understand the cost-benefit trade-offs of implementing an interface which speech input capabilities. On paper, it is evident that the pay off of this option far outweighs the traditional interaction methods especially in large scale implementations.

The third article considered is [3] which attempts to illuminate the trends, themes and challenges in speech-based human-computer interaction. Through a review of 68 research papers this work maps the trends, themes, findings and methods of empirical research on speech interfaces in HCI. The article goes on to explain that most studies are usability/theory-focused or explore wider system experiences. A thematic analysis of the research found that speech HCI work focuses on nine key topics: system speech production, modality comparison, user speech production, assistive technology & accessibility, design insight, experiences with interactive voice response (IVR) systems, using speech technology for development, peopleâs experiences with intelligent personal assistants (IPAs) and how user memory affects speech interface interaction. From this, gaps and challenges can be gleaned to determine what the development market will focus on next.

The next article studied is [4]. This resource seeks to elucidate the trade-off between net temporal profit and convenience/user acceptance of speech-based input. This paper reports the results of an experiment investigating word generation rates, word error rates, and user acceptance of a speech recognition program as compared to typing. Although the subjects made more errors when using the speech recognition software, they were able to generate more than twice as much text in the same amount of time. However, this relative efficiency was not enough to overcome the inaccuracy and annoyance in fixing so many errors.

The fifth article considered is [5] which elaborates on the application areas in which spoken interaction can play a significant role, assesses potential benefits of spoken interaction with machines, and compares voice with other modalities of human-computer interaction. It also discusses information that will be needed to build a firm empirical foundation for the design of future spoken and multi-modal interfaces. Additionally, it argues for a more systematic and scientific approach to investigating spoken input and performance with future language technology.

The last article explored in this section is [6] that explains some of the limitations of speech input. We have indeed covered this under our objectives to show that we do recognise the drawbacks of implementing this mode of interaction. The author goes on to claim that speech is slow for presenting information, is transient and therefore difficult to review or edit, and interferes significantly with other cognitive tasks. This paper affirms our knowledge on the disadvantage of speech interaction.

III. Design

This section details the design decisions for how speech recognition was integrated into the existing interfaces was heavily influenced by the nature of the environment where the application will be used, user backgrounds, and related to the HCI design principles we have learnt.

To showcase the design approach, Fig. 1 below depicts the login/registration form interface which would be used by the patients, and a receptionist interface where they can en-queue people. As the application will be deployed in a health center, we expect 2 types of users: patients, and the health center officers.

A. Patients' Form Interface

We cannot make any assumptions regarding the background, age, and education level of the patients. Thus, in our design, our goal for the patient form interface was to make it as clear and easy to use as possible.

Visibility of system status, and error prevention are two key points that we deemed profoundly important for this category of users. The interface should have clear indicators that show it started/stopped listening to speech input.

One noticeable design decision is the presence of a microphone next to each form field. This is to prevent errors and to make it easier for the user to recognize the utility of these buttons. This means that clicking a microphone and entering speech input will only lead to modifying the field it is attached to. Most people do not regularly visit health centers. We expect the majority of patients to be either new users, or ones that have not used the system in a long time. Having a microphone button attached to each field will make it easier for users to understand what pressing each one of them does. Figure 1 shows a scenario where the patient tries to register using the login/registration interface.

B. Health Officers' Interface

For the health officers, we were able to make the system a bit more robust. For example, the receptionist has an interface where they can en-queue patients. In the interface, they would have to enter the patient ID, and the name of the doctor who will see the patient. Instead of having a microphone button attached to each field (like we did for the patients' form), we have a single microphone button that can be used to enter/modify data in all fields. The receptionist can click on the microphone and then for example, say "Patient 1234 Doctor Ahmed", Or "Doctor Ahmed Patient 1234", or pick and modify a single field by saying "Patient 1234". The system will parse the speech transcript and recognize which fields the receptionist is trying to modify.

We found it important to add robustness to the system because we expect the receptionist to be exposed to the interface on a daily basis. We also expect that the receptionist will receive some training about the interface. Owing to this, they will know that the aforementioned microphone button can be used to modify all fields, and that the system gives them more freedom when it comes to structuring their sentences when issuing speech input. All this serves to make it less daunting for the receptionists' daily use of the interface.

Fig. 2 shows a scenario where the receptionist tries to enter data, and then modifies a field using speech input.

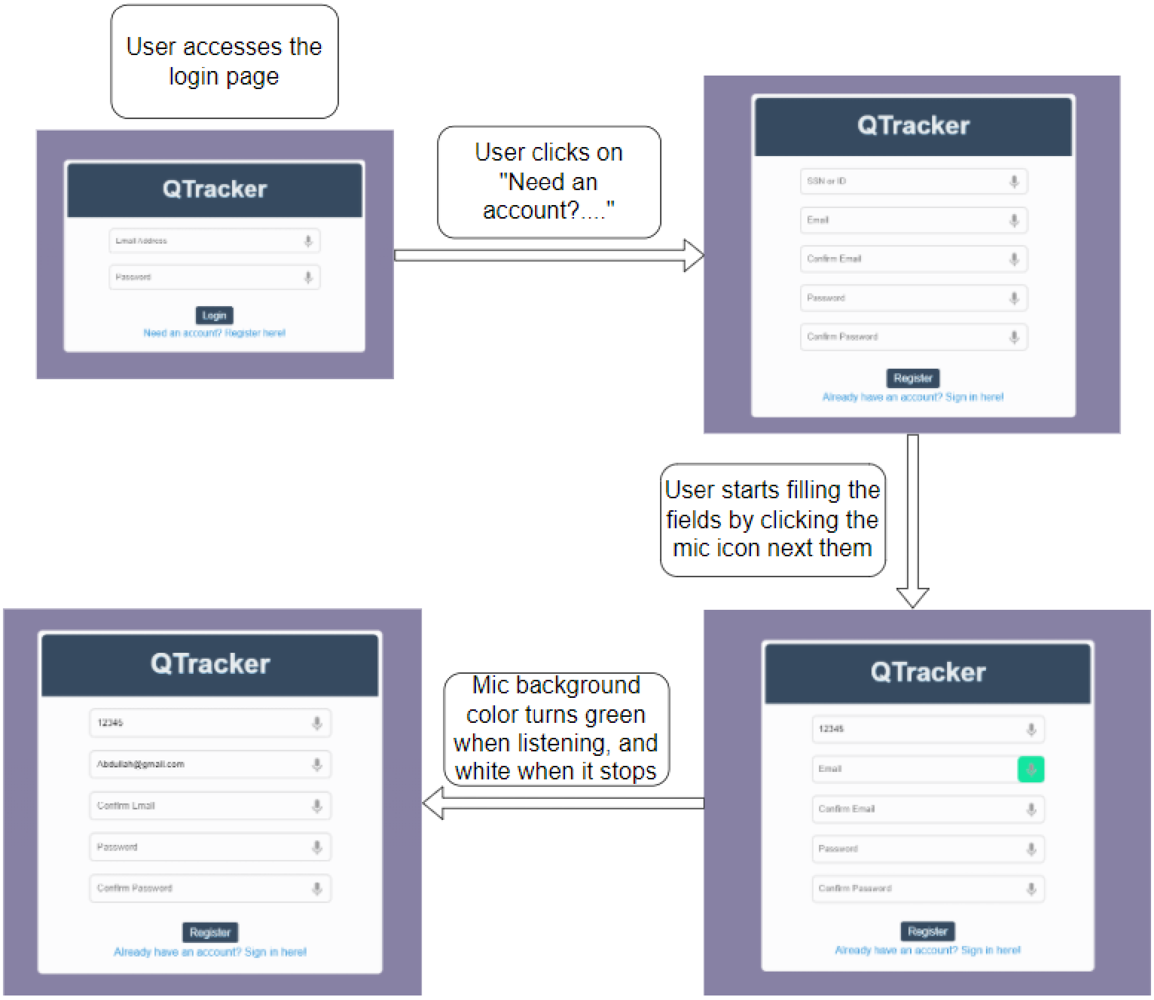

Fig. 1. Patient login/register interface usage scenario.

IV. Implementation

The discussion in the "Design" section is heavily reliant on the implementation. We did not want to separate the two, so we included the functionality details in the previous section. In this section, we will only mention the technologies we used to bring about the aforementioned functionality and interfaces.

The hitherto discussed interfaces are part of larger, complex, distributed system. The technologies used to implement the system as a whole will not be covered rather we shall touch on those directly used to integrate speech input into the interfaces

For the visual elements of the interface, PHP, HTML, CSS, and JavaScript were used. JavaScript was also used to integrate speech recognition [9]. We used the JavaScript "Web Speech API" to add speech recognition to our interfaces and to parse the input transcript.

It is worth mentioning that not only native JavaScript was used, but also one of its frameworks, JQuery. We used JQuery because it allows us to easily detect events and modify visual properties of HTML elements based on those events. It was mostly used in areas where we needed to add visual effects that show system status (For example, listening/waiting mic effects).

V. Experimentation Methodology: Evaluation environment setup & Evaluation

This section details the experimentation phase of our project. It would be prudent to restate the aim of the study which is to compare the efficiency of the current text-input based health workers' system/web application with the improved version that uses an augmentation of both text and speech input.

We will discuss the experimentation methods and subsequent procedures, participant details as well as the metrics we used for evaluation.

A. *Methods & Procedures: Evaluation*

In this section, we will elaborate the process by which we set up the environment to perform the evaluation via the two approaches we aim to compare.

The setup process involves selecting some fundamental tasks (with reasons given for each chosen

Fig. 2. Receptionist interface usage scenario.

task), we will introduce our test subjects (whose real identities shall remain anonymous but shall be referred to with pseudo-names) and explain the rationale for selecting them if any, we will continue to mention the evaluation techniques we have chosen to compare and finalise by describing any additional assumptions that we took into account for our evaluation environment.

B. Fundamental Tasks

For our evaluation process, we selected the two fundamental tasks elaborated below:

1. Accessing patient information: The QTracker web app, as any health office application ought to, allows the user to view the details of their patients. Patient details include their personal information e.g., name, address, telephone number; as well as crucial data such as appointment history or any pending appointments that the patient might have. This task is likely to be the most used by the health officers at the clinic thus, we have chosen this as our "easy" task.
2. Making an appointment: Since our goal is to test a new form of input, it is imperative that one of our tasks involves significant data entry to stress-test the system. For this reason, we have chosen the "Create Appointment" task as our "difficult" task for our evaluation.

C. Test subjects

In this subsection, we will introduce our test subjects and why elaborate the importance of each of them in our evaluation study.

1. Test subject **A** (as they will be referred to throughout this study): a Psychology student at METU NCC. The test subject is fluent in Turkish but only conversant with English language to a reasonable degree. The student uses a computer on occasion but not frequently enough for them to be considered an avid user. Furthermore, the test subject's use of their computer rarely entails dealing with forms; either entering or retrieving data. The test subject is also not very conversant with speech-input interfaces in their daily life. Thus, test subject A qualifies as our ***novice*** user/test subject for the user-based evaluation.
2. Test subject **B** (as they will be referred to throughout this study): Is a Computer Engineering student at METU NCC. The test subject is reasonably conversant with both Turkish and English languages. The student uses a computer frequently enough for them to be considered an avid user. However, although the test subject's daily use of their computer entails dealing with forms; either entering or retrieving data, they have little to no experience with speech-input based interfaces such as the one in this study. The test subject is conversant with speech-input interfaces (e.g., Cortana and Siri) in their daily life. Thus, test subject B qualifies as our ***intermediate*** user/test subject for the user-based evaluation.
3. Test subject **C** (as they will be referred to throughout this study): Is a Computer Engineering student at METU NCC. The test subject is reasonably conversant with both Turkish

and English languages. The student uses a computer frequently enough for them to be considered an avid user. Additionally, the test subject's daily use of their computer entails dealing with forms; either entering or retrieving data; and they have vast experience with speech-input based interfaces such as the one in this study. Thus, test subject C qualifies as our ***advanced*** user/test subject for the user-based evaluation.

4. Test subjects **E1 & E2** (as they will be referred to throughout this study): The developers of the system and hence, the user interface under study. The evaluators have adequate knowledge in human-computer interaction and the accompanying expertise about design heuristics vital for user interfaces. Furthermore, the evaluators have experience with development and use of speech-input based interfaces such as the one under study. Thus, test subject E1 & E2 qualifies as our ***expert*** user/test subject for the expert-based evaluation. E1 evaluated the text-input based UI while E2 evaluated the speech-input based UI. This allocation was done arbitrarily.

The table below summarizes the user demographics.

	TS A	**TS B**	**TS C**	**TS E1 & E2**
Text-input UI expertise	Low	Medium	High	High
Speech-input UI expertise	Low	Medium	High	High
HCI knowledge	Low	Low	Low	High
English proficiency	Medium	High	High	High

TABLA I
User demographics

D. Evaluation Methods

In this subsection, we will introduce and elaborate on the evaluation methods that are the subject of comparison in this study. We will further explain the rationale behind the evaluation methods we have chosen and their suitability to our test subjects and environment.

1. In **expert-based evaluation**, a designer or HCI expert assesses a design based on known or standard cognitive principles or empirical results. Expert-based evaluation techniques are also referred to as expert analysis techniques. Examples of expert analysis methods include: heuristic evaluation, cognitive walkthrough, and review-based evaluation.
With this is mind, we chose ***heuristic analysis*** as the method for expert-based evaluation.
The rationale behind opting for this method lies in the need for an approach that ensure that the expert makes use targeted design evaluation metrics to perform the evaluation within the shortest time possible. The rudimentary "questionnaire" employed entails the ten design heuristics by the Neilsen-Norman Group from [8] coupled with a YES/NO metric for the experts to encapsulate whether or not they deem the system/UI to have adhered to the corresponding heuristics.
As is a requirement of heuristic analysis, the experts then compile a report outlining why they evaluated the UI as having met the design heuristic or not. This report, however, will be an all-inclusive report of the usability of the speech-input based UI and not just based on the two tasks chosen earlier.
We would like to reiterate that the questionnaire is only a guidance tool to distill the expert's evaluation and not the primary findings of the experts' evaluation; rather this is in the reports compiled.
The assertions from the design heuristics which the experts were asked whether or not they agree with were framed as follows:

- Visibility of system status: The operations needed to accomplish the task were clearly visible and accessible every step of the way. With regard to the speech-input interface that is part of this study, this heuristic entails the system's ability to perform important communication tasks to the user such as feedback and confirmation. Does the system provide an auditory icon, for instance?
- Aesthetics and minimalism: The application presents UIs in an aesthetically pleasing way and the UI's degree of minimalism is optimum. The information provided on the UI is focused on the essential and is enough to guide the user without needing additional explanation but not too much to crowd the interface or overwhelm the user.
- User control and freedom: The application provides the user the chance to safely abandon an incomplete operation in the event of a misstep.
- Consistency and standards: The operations on every availed UI window throughout the task are consistent. There are no operations that produced conflicting outcomes. Performing the same task on different platforms entails the same set of operations.
- Error prevention: The UI notified the user of a possible error and allowed the user the chance to correct the error before proceeding. With regard to the speech-input interface, does the interface give specific error messages? is the interface adaptable/adaptive in the event of background noise? does the system allow the

user to turn off the input device? does the system provide for undo operations?

- Recognition rather than recall: The tasks on the UI are enough to guide the user to perform any operation they would like to. The user need not depend on their memory of previous use of the system or a similar system.
- Flexibility and efficiency of use: The UI provides the user the chance to customise the view. The user can easily and safely select a UI appearance, language or other user-preferred functionality using straightforward operations.
- Match between the system and the real world: The user is able to employ knowledge of daily life tasks/objects to perform a task or operation or interpret an otherwise unfamiliar task without producing an undo-able or unpredictable effect.
- Help user to recognise, diagnose and recover from errors: The UI notified the user of a possible error and allowed the user the chance to correct the error before proceeding. The UI allowed the user the chance to undo an erroneous operation.
- Help and documentation: The UI provides adequate, easily comprehensible documentation which assisted the user upon implementation.

The five-point (ranging from 'Strongly Disagree' to 'Strongly Agree') Likert scale was then provided underneath each of the assertions for the evaluation of each task.

2. In **user-based evaluation**, we also used a ***questionnaire***. The questionnaire provided to the test subjects was open-ended and more inclined toward measuring the users' experience as the target market.
The questionnaire is modelled based on the System Usability Scale (SUS) in [7] with ten items accompanied with a five point Likert scale ranging from Strongly Agree to Strongly Disagree for simplicity of use; and were as follows:
An important point to note would be that the user-based evaluation of the speech-input user interface was carried out in the school cafeteria (pastane) to simulate realistic background noise that would be present in the health facility for which the web application is intended.

- I think that I would like to use this system frequently.
- I found the system unnecessarily complex.
- I thought the system was easy to use.
- I think that I would need the support of a technical person to be able to use this system.
- I found the various functions in this system were well integrated.
- I thought there was too much inconsistency in this system.
- I would imagine that most people would learn to use this system very quickly.
- I found the system very cumbersome to use.
- I felt very confident using the system.
- I needed to learn a lot of things before I could get going with this system.

VI. Results

In this section, we will analyse the results of the expert and user based evaluation performed for the QTracker web application for health officers.

We will provide the results grouped by the evaluation approach (either expert or user based) and proceed to compare and contrast the findings.

A. Expert Based Evaluation

The findings of the expert-based approach are as shown below.

On the Yes/No scale, [1 - 'Yes', 0 - 'No']. It is important to note that in this case, the neutral value was primarily used for heuristics that could not be tested for in the corresponding task.

The table below elaborates the findings of the expert evaluation. The columns correspond to the the findings of each of the experts. These findings were then compiled in one report elaborated afterwards.

	E1 - Text	**E2 - Speech**
Visibility	YES	YES
Aesthetics & Minimalism	YES	YES
User control and freedom	YES	YES
Consistency & standards	YES	YES
Error prevention	YES	YES
Recognition vs. recall	YES	YES
Flexibility & efficiency	YES	YES
Mapping	YES	YES
Error recognition, diagnosis & recovery	YES	YES
Help & documentation	NO	NO

TABLA II
Findings from expert-based evaluation

The following reports was compiled from the findings of the expert evaluation as follows.

A.1 Expert's heuristic evaluation reports

The first report below entails expert E1's detailed comments explaining his decisions on whether or not the ten Nielsen Normann design heuristics were met during the design of the ***text-input*** user interface.

1. ***Visibility of system status***: The UI adheres to the visibility heuristic. The state of the system is conveyed to the users in a manner that is easy to comprehend. The system keeps the user

informed on what is going on through appropriate feedback within a reasonable time (discounting any network connection issues) to empower them to make decisions. The amount of information conveyed is not too much to distract the user and is just enough for the user to act on. The UI therefore achieves open and continuous communication with the user. A notable case of this is, for instance, when the user chooses the option to save entered patient's records, a spinning wheel appears to communicate to the user that the requested screen is loading.

2. ***Match between the system and the real world***: The system speaks the user's language; with words, phrases and concepts familiar to the user rather than system oriented terms. The UI follows real-world conventions and the events constituting [the chosen] operations and subsequent information appears in a logical order. The system uses direct plain language in availing options. Additionally, the sequence of operations that the second task entails, follow a logical order.
3. ***User control and freedom***: For the chosen tasks, the system UI provides clearly marked emergency exit for when the user performs an unwanted operation. The UI supports user undo and redo operations in line with this heuristic. This is exhibited by the undo (back arrow) button availed on the screen at each step of the tasks to allow the user to leave an unwanted state in the event that they select a wrong option as well as the redo (front arrow) button provided if the user would like to revert to the state they left, for instance. Though these are availed by the browser, they still enable the UI to provide the user with control.
4. ***Consistency and standards***: The UI is predictable and learn-able. The user need not wonder about conflicting outcomes for the same set of actions. Tapping on an option on the menu sends you to a sub-menu or opens up the interface to perform the operation you chose to do. This is maintained throughout the UI. This is in line with the expectations for internal consistency. Accessing the UI on iPad or an iPhone to perform the chosen operations requires the user to perform the same operations as on the PC. This is in line with the expectations for external consistency.
5. ***Error prevention***: The application provides adequate support for error prevention. For example, the user is prompted to confirm the entered patient's details in the second task. The application provides an unobstructed view of all the details before asking the user to confirm the records being saved in the database. Additionally, the UI provides clear and concise error messages for wrong inputs.
6. ***Recognition rather than recall***: The success of the operations entailing using the web application is clearly more dependent on the user recognising the label on the menu provided. The tasks involved (entering and retrieving patient details) require short and straightforward procedures, thus reducing the users' cognitive load.
7. ***Flexibility and efficiency of use***: The web application provides adequate support for the user to perform simple but crucial operations such as copying and pasting patient information (for instance, addresses that would otherwise long and erroneous) to the UI which can be edited further. These operations are the most error prone when one is creating a patient record. Provision for such tasks improves user experience by providing flexibility and sharpening efficiency during use.
8. ***Aesthetics and minimalism***: The UI design is focused on the essentials. The information provided on the UI is focused on the essential and is enough to guide the user without needing additional explanation but not too much to crowd the interface or overwhelm the user. The labels are clear and concise.
9. ***Help Users Recognize, Diagnose and Recover from Errors***: The UI clearly informs the user when error has occurred by providing clear and concise error messages The UI notifies the user what went wrong. The error messages provided have enough information and in a language that the user can understand, to notify the user why they are receiving the message. The UI offers a way to fix the error. The user is provided with placeholders that guide them on what to enter at each point when filling the forms.
10. ***Help and documentation***: There is no definitive additional documentation given to assist the user in case they face any challenge they cannot solve depending on their previously acquired knowledge alone.

The first report below entails expert E2's detailed findings upon concluding the heuristic evaluation of the ***speech-input*** user interface.

Some findings of the heuristic evaluation were similar to those of the text-input UI evaluation because the speech-input UI was only upgraded to support speech interaction and not fully revamped from the original text-input UI.

1. ***Visibility of system status***: The UI adheres to the visibility heuristic. The state of the system is conveyed to the users in a manner that is easy to comprehend. The system keeps the user informed on what is going on through appropriate feedback within a reasonable time (discounting any network connection issues) to empower them to make decisions. The amount of information conveyed is not too much to distract the user and is just enough for the user to act on. The UI therefore achieves open and continuous communication with the user. A

more specific instance of visibility regards the microphone icon/button for speech input availed alongside each field. The button conveys a clear and concise message as to how the user is to interact with the UI. In addition, the icon changes color to indicate that the user's input is being received and reverts back to the original color after some time to indicate that the UI is no longer receiving input.

2. ***Match between the system and the real world***: The system speaks the user's language; with words, phrases and concepts familiar to the user rather than system oriented terms. The UI follows real-world conventions and the events constituting [the chosen] operations and subsequent information appears in a logical order. The system uses direct plain language in availing options. An instance of mapping is evident via simple information such as the microphone icon beside the corresponding text field conveys the message that the user is supposed to click on it to enter the speech input.
3. ***User control and freedom***: For the chosen tasks, the system UI provides clearly marked emergency exit for when the user performs an unwanted operation. The UI supports user undo and redo operations in line with this heuristic. This is exhibited by the undo (back arrow) button availed on the screen at each step of the tasks to allow the user to leave an unwanted state in the event that they select a wrong option as well as the redo (front arrow) button provided if the user would like to revert to the state they left, for instance. Though these are availed by the browser, they still enable the UI to provide the user with control. Additionally, the user is allowed to delete any input that may have been garbled by background noise or incoherent speech as well as the option to use text input for correction.
4. ***Consistency and standards***: The UI is predictable and learn-able. The user need not wonder about conflicting outcomes for the same set of actions. Tapping on an option on the menu sends you to a sub-menu or opens up the interface to perform the operation you chose to do. This is maintained throughout the UI. This is in line with the expectations for internal consistency. Accessing the UI on iPad or an iPhone to perform the chosen operations requires the user to perform the same operations as on the PC. This is in line with the expectations for external consistency.
5. ***Error prevention***: The application provides adequate support for error prevention. For instance, the patients' interface has a voice input button for each field to reduce garbage input that results from an agglomerated input button for users who are not well-versed with the system. The color change discussed under "Visibility of system status" also guards against garbage input since the user knows when the interface is receiving input and when it is not. Additionally, the user is provided the opportunity to view the speech-to-text input received on the UI in case there are corrections that need to be done.
6. ***Recognition rather than recall***: The success of the operations entailing using the web application is clearly more dependent on the user recognising the label on the menu provided. The tasks involved (entering and retrieving patient details) require short and straightforward procedures, thus reducing the users' cognitive load.
7. ***Flexibility and efficiency of use***: The web application provides adequate support for the user to perform simple but crucial operations such as copying and pasting patient information (for instance, addresses that would otherwise long and erroneous) to the UI which can be edited further. These operations are the most error prone when one is creating a patient record. Provision for such tasks improves user experience by providing flexibility and sharpening efficiency during use.
8. ***Aesthetics and minimalism***: The UI design is focused on the essentials. The information provided on the UI is focused on the essential and is enough to guide the user without needing additional explanation but not too much to crowd the interface or overwhelm the user. The labels are clear and concise.
9. ***Help Users Recognize, Diagnose and Recover from Errors***: The UI clearly informs the user when error has occurred by providing clear and concise error messages The UI notifies the user what went wrong. The error messages provided have enough information and in a language that the user can understand, to notify the user why they are receiving the message.
10. ***Help and documentation***: There is no definitive additional documentation given to assist the user in case they face any challenge they cannot solve depending on their previously acquired knowledge alone.

B. User Based Evaluation

On the five-point Likert scale alluded to when discussing the evaluation methods used in the preceding section, the key is as follows: Scores [1 - 'Strongly Disagree', 2 - 'Disagree', 3 - 'Neutral', 4 - 'Agree', 5 - 'Strongly Agree'].

The findings of the user-based approach are shown in the tables below.

The first table outlines the Likert Scale scores awarded by the users for each of the questions/assertions provided from the System Usability Scale (SUS) in [7], upon using the text-input based user interface.

	TS A	**TS B**	**TS C**
Q(a)	2	1	1
Q(b)	4	4	5
Q(c)	2	2	1
Q(d)	4	5	4
Q(e)	3	1	2
Q(f)	5	5	4
Q(g)	2	1	2
Q(h)	5	4	5
Q(i)	3	2	1
Q(j)	5	4	5

TABLA III
Findings from user-based evaluation of text-input based UI

The second table outlines the Likert Scale scores awarded by the users for each of the questions/assertions provided from the System Usability Scale (SUS) in [7], upon using the speech-input based user interface.

	TS A	**TS B**	**TS C**
Q(a)	5	3	3
Q(b)	2	3	4
Q(c)	4	5	3
Q(d)	1	2	3
Q(e)	2	1	1
Q(f)	4	4	5
Q(g)	5	3	3
Q(h)	1	2	4
Q(i)	5	4	2
Q(j)	1	2	4

TABLA IV
Findings from user-based evaluation of speech-input based UI

VII. Discussion

In this section, we will comment on the findings of the evaluation of the text and speech input based interfaces covered in this study. We will discuss the findings by grouping them based on the evaluation technique; either user-based or expert evaluation.

A. Expert-based evaluation

A.1 Text-input UI

All heuristics were adhered to except the last one on help and documentation, which admittedly, might be perceived as redundant given the small range of tasks/operations that are carried out on the user interface under study.

Owing to the nature of the UI and the target environment of its use, evaluation of some of the heuristics couldn't actually be stretched to the limit. For instance, the limited amount of activities in the domain render the results for "User control and freedom" rather superfluous.

To sum up, expert based evaluation ends up being a bit of overkill on some heuristics and too general for others even though its results are more reliable.

A.2 Speech-input UI

Similar to the text-input UI, all heuristics were adhered to except help and documentation which in this case would be of some use.

However, even though the results of this heuristic evaluation present the picture of a well-designed user interface, this mode of expert evaluation does not take into account the deployment environment.

As explained in the findings of the user-based evaluation, a speech-input UI is not suitable for a highly volatile environment such as a health center. The sentiment is that if there is a text-input option, most users would rather go with that despite the speech-input capability being up to the universally agreed upon design standards.

B. User-based evaluation

B.1 Text-input UI

Table I in the preceding section summarizes the findings of the user-based evaluation of the text-input based web application.

The general sentiment from the test subjects (albeit a very small pool) is that the text-input UI is very user-friendly and lightweight in terms of the cognitive load required to use it.

The test subjects further mentioned that the UI is/was learn-able, flexible and robust. Proceeding to account this to the end state that the user hopes to achieve not being complex i.e., the range of tasks that the user has to perform on the UI is very limited and the tasks are very rudimentary so the margin for error on the part of the user is very slim.

This assessment is indeed in line with our expectations. Most of the target users have probably interacted with hundreds if not thousands of text-input based UIs their lives and as such, measuring usability factors such as learnability may be heavily influenced by prior knowledge of UIs that require text input to interact with forms.

On the other hand, the other usability principles (flexibility and robustness) were measured satisfactorily. It was interesting to see the test subjects not even think twice about using copy-and-paste technique, for instance, to get their (usually long) address information from another application to the web application.

It would also be prudent to mention that the Test subjects did not exhibit any negative emotions while interacting with the UI.

B.2 Speech-input UI

Table II in the preceding section summarizes the findings of the user-based evaluation of the speech-input based web application.

The results we observed are indeed in line with the test subjects' expertise level in dealing with speech-input user interfaces. The user interface is not bad per se, but rather the environment in which it has been deployed results in very poor user experience.

This is very evident from the scores awarded by the test subjects. It can be seen that the question/statements dealing with or touching on the design of the UI, consistency, how the functions are presented to the user, etc, are positive across the board. The test subjects, albeit subconsciously, recognise that the factors leading to poor user experience, in this case at least, stem from outside the user and the interface themselves.

The UI does provide the tasks needed to accomplish the task in a clear and simple manner/language that the average user can comprehend but the user's accent or background noise hinder effective and enjoyable interaction.

In addition, we also observed that simply the fact that there is an alternative mode of interaction (text) had a huge psychological impact on the user's commitment to interacting via speech.

VIII. Conclusion

In this report, we have extensively covered the deployment of a speech-input UI/web application in a health care environment as well as the evaluation of its usability. In this section we will summarise the information presented thus far.

A. Text-input UI vs. Speech-input UI

Our conclusion regarding the comparison between the two interfaces covered in this study is that however futuristic and modern speech interfaces may appear or be, some environments are simply not suited for speech-input UIs. The target domain, for instance, is an example of an extremely volatile set up in terms of the environment (background noise) interfering with the input or the users' dialects and speech impairments being a hindrance.

A speech-input UI is more suited for a quiet and calm environment. Preferably one in which time is not of the essence while carrying out the tasks required on the UI. A health facility is not one such environment. It is important the UI deployed in such a case provide flawless user experience so as not to bring about unnecessary inconvenience.

It is also important to note that it is possible for the design heuristics to be followed to an acceptable degree but still end up with poor user experience; as we have seen when discussing the findings in the preceding section.

To this effect, we can confidently say that the original text-input UI is better than the speech-input UI.

B. Expert vs. User-based Evaluation

This study was as much about evaluating the suitability of having a speech-input based user interface on a web application for health workers as it was about exploring various methods/techniques of evaluation of user interfaces.

The upside of expert-based evaluation is that the results of the evaluation are based on recognised standards. However, looking at the UI through the lens of guidelines, standards and heuristics removes the aspect of the UI being a living, breathing entity. Moreover, guidelines and heuristics reduce the UI to a checklist for the designers/programmers to adhere to instead of empathizing with the users of the target environment, especially an environment as sensitive as a health facility.

Referencias

[1] *Advantages of Voice User Interfaces*, https://www.speechly.com/blog/advantages-of-voice-user-interfaces/

[2] *Speech Input*, https://wunder.io/wunderpedia/accessibility/digital-assistive-technologies/speech-input/

[3] *The State of Speech in HCI: Trends, Themes and Challenges*, Leigh Clark[1], Phillip Doyle[1], Diego Garaialde[1], Emer Gilmartin[2], Stephan Schlogl[3], Jens Edlund[4], Matthew Aylett[5], Joao Cabral[6], Cosmin Munteanu[7], Benjamin Cowan[1] [1]*School of Information and Communication Studies, University College Dublin, Ireland,* [2]*Speech Communication Laboratory, Trinity College Dublin, Ireland,* [3]*MCI Management Center, Innsbruck, Austria,* [4]*Department of Speech, Music and Hearing, KTH Royal Institute of Technology, Sweden,* [5]*School of Informatics, University of Edinburgh, United Kingdom,* [6]*School of Computer Science and Statistics, Trinity College Dublin, Ireland,* [7]*Institute of Communication, Culture, Information and Technology, University of Toronto, Mississauga, Canada*

[4] *Speech recognition in the humanâcomputer interface*, Carl M. Rebman Jr, Milam W. Aiken, Casey G. Cegielski

[5] *The role of voice input for human-machine communication*, Philip R. Cohen, Sharon L. Oviatt *"Human-Machine Communication by Voice,." colloquium organized by Lawrence R. Rabiner, National Academy of Sciences, The Arnold and Mabel Beckman Center in Irving, CA, February 8-9, 1993.*

[6] *The limits of speech recognition*, Ben Shneiderman *Communications of the ACM, Volume 43, Number 9 (2000), Pages 63-65*

[7] *System Usability Scale*, https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

[8] *10 Usability Heuristics for User Interface Design*, https://www.nngroup.com/articles/ten-usability-heuristics/

[9] *Web Speech API*, https://wicg.github.io/speech-api/