Adapted Symptom Screening Application: COVID-19 Symptom Screening for People with

Intellectual and Developmental Disabilities

Olivia Shope

Department of Occupational Therapy, Duquesne Univeristy

1) Problem Statement/Research Question and Background

The COVID-19 pandemic has brought on significant challenges for people around the world but has been particularly difficult for those with disabilities. Daily symptom screenings are now a regular occurrence but are not always the most accessible. On Duquesne University's campus, we have an inclusive education program, and one of the faculty members involved reached out for help regarding a student's difficulties with using the symptom screening. The symptom screening questionnaire required by the university requires was not a good fit for a particular student with an intellectual/developmental disability (IDD) who struggles particularly with literacy and numeracy.

At the beginning of the project, the student was being called each morning by one of the student aids to go through the questions and record his temperature, but there were several days that the aid was unable to make that call due to time constraints or prior engagements. It was determined that the student needed a way to complete the form independently. H needed to have the instructions read aloud and phrased in a more comprehensible way. An application that could be downloaded to his phone was decided to be the best option.

2) Methods/Approach/Solutions Considered

To create this application, direct input from the student and his aid was needed. Weekly meetings with them to test and revise the application and make changes to the functions were scheduled. The most important need to be addressed was that the questions listed in the symptom screening needed to be spoken aloud by the application. It was also concluded that the questions needed to be presented one at a time with distinct buttons indicating 'yes' and 'no'. There also needed to be a feature that allowed the student to input his temperature each day.

Once the application had been prototyped with the questions from the University's screening instrument, it became apparent that some of the symptoms were not described in a way that the client was able to conceptualize. One of the weekly meetings was dedicated to going through each question and having the student identify what was being asked. The questions were then rephrased in language that the student commonly uses and in simpler terminology.

3) Description of Final Approach and Design

After working with the client and his aid in determining what would be best for them in screening for the symptoms and accessing the data, the final design was developed. The final design included a picture or cartoon portraying the symptom being asked about, the question displayed in text in common language, and the question was read aloud when the client clicked on the picture. The yes and no buttons are clearly displayed, with no in red and yes in green. Only one question has a desired input of 'yes,' which is the prompt about loss of taste and smell. This was easier for the student to understand by phrasing the question as if he was still able to smell and taste his food.

Once an answer is selected, the application moves on to the next question and goes through each one by one. Once all the questions have been answered, it moves to the next screen which is the temperature input. It was decided that the best approach to take was to have the student take a picture of his thermometer after taking his temperature. The second screen automatically prompts the student by telling him aloud to take his temperature and then take a picture. There is another button on the screen that allows the user to view the data stored in the application. This displays the date, answers given, and the picture of the thermometer from that day.

Once the application design was complete, the student was educated on how to use all the features, and the student aid was shown how to access the data. A feature was added later where the application records when the app has been used in an online database. This allowed for an easy way to check if the student was using the application consistently. The answers and pictures are only displayed and stored within the application on the student's device to protect privacy.

4) Outcome

Feedback from the student and his student aids has been very positive. The student reports that it is easy to use and has been using it most days of the week and it is part of his morning schedule. The student aid has been able to view the data and easily organize it to keep track without calling the student every day to use it. The faculty member that originally requested the application also reports that the student was using it every day during the fall semester and at the beginning of spring semester until the student received his vaccination.

5) Cost

The application was developed on the free program Thunkable because it was compatible with iOS, but in order to download the application to the student's device, the professional version of the program is needed. The professional version costs $20 a month and is the only expense for the developers.

6) Significance

COVID-19 pandemic has required changes in many aspects of life and the introduction of several new requirements. This has been especially challenging for people with disabilities. In this case, a symptom screening meant to protect the safety and health of students on a college campus was not accessible to the students on campus with IDDs due to its format and

presentation. An application was customized to fit the needs of a specific client, but the code and format is easily changed to fit the needs of others.
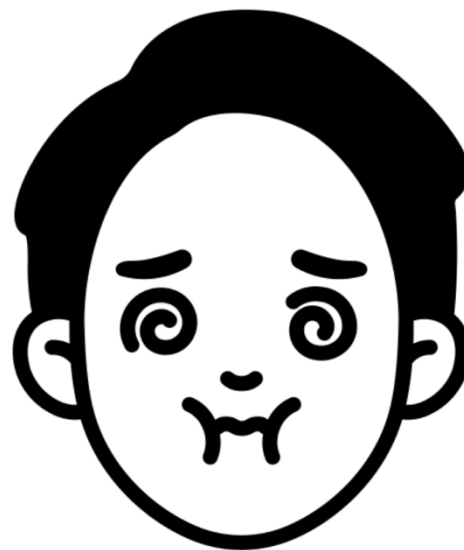
7) Acknowledgements

8) Product Photos

3/21/2021	>
Feel sick? No	>
Sore throat? No	>
Headache? No	>
Coughing/trouble breathing? No	>
Body hurt? No	>
Taste and smell? Yes	>
Stomach hurts? No	>
Runny nose? No	>

Previous Restart Next

ed by Victoruler
'oun Project

Feel sick?

Yes   No

initialize app variable pic...
initialize app variable que...
initialize app variable spe...
initialize app variable lst...
initialize app variable intQuestionIndex to 1

initialize app variable lstTodaysResponses to (empty list)
initialize stored variable lstResponseDB

when stored variable lstResponseDB initializes or changes
do if stored variable lstResponseDB == null
    do set stored variable lstResponseDB to (empty list)

when btnQuestion . Long Click
do set Alert1 . Message to "Skip to view data?"
    call Alert1 . Show
    with output
    then do if wasConfirmed
        wasConfirmed
        navigate to scrnData

to incrementAppIndex and return
do if app variable intQuestionIndex == length of app variable questionList - 1
    do navigate to scrnThermometer
    else return app variable intQuestionIndex + 1

to displayQuestion
set btnQuestion . Background Picture to (in list app variable pictureList get # app variable intQuestionIndex)
set lblQuestion . Text to (in list app variable questionList get # app variable intQuestionIndex)

when scrnQuestions . Opens
do in list app variable lstTodaysResponses insert at last as (join current month "/" current day of the month "/" current year)
    set app variable intQuestionIndex to 2
    displayQuestion

when btnYes . Click
do in list app variable lstTodaysResponses insert at last as (join (in list app variable questionList get # app variable intQuestionIndex) "Yes")
    set app variable intQuestionIndex to incrementAppIndex
    displayQuestion

when btnNo . Click
do in list app variable lstTodaysResponses insert at last as (join (in list app variable questionList get # app variable intQuestionIndex) "No")
    set app variable intQuestionIndex to incrementAppIndex
    displayQuestion

initialize app variable intResponseIndex to 1

when scrnData . Opens
do set app variable intResponseIndex to length of stored variable lstResponseDB
    show record

when btnRestart . Click
do navigate to scrnQuestions

when btnRestart . Long Click
do set Alert1 . Message to "Are you sure you want to erase all your data?"
    call Alert1 . Show
    with output
    then do if wasConfirmed
        wasConfirmed
        set stored variable lstResponseDB to (empty list)
        navigate to scrnQuestions

to show record
set List_Viewer1 . text items to (in list stored variable lstResponseDB get # app variable intResponseIndex)
set imgThermometer . Picture to (in list (in list stored variable lstResponseDB get # app variable intResponseIndex) get last)

when Previous . Click
do if app variable intResponseIndex >= 1
    do change app variable intResponseIndex by -1
        show record

when Next . Click
do if app variable intResponseIndex <= length of stored variable lstResponseDB
    do change app variable intResponseIndex by 1
        show record

when scrnThermometer . Opens
do call Text_To_Speech1 . Speak
    text "Please take your temperature, then hit "Take a p..."
    set lblInstructions . Text to "Take your temperature"

when btnViewData . Click
do navigate to scrnData

when btnTakePicture . Click
do call Camera1 . take photo
    with outputs Photo
                  DidUserCancel
                  Error
    then do in list app variable lstTodaysResponses insert at last as Photo
        call Share1 . ShareImage
        image Photo
        with output error
        then do call Text_To_Speech1 . Speak
            text "Thank you, your are done"
            in list stored variable lstResponseDB insert at last as app variable lstTodaysResponses
            call Realtime_DB1 . Save
            key (join current year "-" current month "-" current day of the month "-" current hour "-" current minute)
            value true
            with output error
            then do navigate to scrnData