Gesture Based UI Project 2017

PURPOSE OF APPLICATION

The idea of the application was a simple number genie app for UWP that could be controlled using voice commands by Cortana

Number Genie is a number guessing game where you must guess the computers number which is between 1 and 100. The app be controlled in 2 ways: using the GUI to type in your guess and clicking one of two buttons, one to submit your guess and one to choose a new random number.

The other way is through Microsoft Cortana. A voice recognition app that would hear in your guess and check it against the computers number.

The main point of the Project was too see how to implement Cortana into an otherwise simple app as well to test its voice recognition and functionality.

DISCLAIMER/ERRORS

The app doesn't work. Or rather, the voice commands don't work as they should (You can use the text box and buttons in the app to guess and cycle target numbers, but since that's not the point of the project, I don't know if that's worth anything. It doesn't crash and it runs normally, so there's that).

Both voice commands supplied have separate issues:

Start app command: Only works when the app is already open. If you use it while the app is closed Cortana will recognize the command and try to open it but will only open a null frame.

Guess number Command: Seems to restart the application. I think this is because when I call this this command in the code I navigate to "mainpage.xaml" which has hardcoded data. It might process the command then navigate to the main page where the hardcoded data overwrites the new data (new guess). I tried moving all the data to variables in another class but this didn't work.

Also, there is a number in the top right (default value -1) that is actually the number you have to guess for. It is just there for debugging purposes and to make sure that the rng works

GESTURES IDENTIFIED AS APPROPRIATE FOR THIS APPLICATION

For the number genie app, there are 3 main controls: Starting the app, Guessing the number and choosing a new random number.

Therefore, there should be 3 voice commands. One for starting the app ("Start Number genie"), one for guessing a number ("Number Genie, try number 44") and one for selecting a new random number ("Number Genie, select a new number").

Through experimentation with the voice commands I found I could combine two of the commands into as they served a similar function. The new random number command simply brought the app back to default state with new random number, like the app starting command. Coupled with the apps low memory use I could simply restart the app by once again calling the app starting command which would generate a new number.

HARDWARE USED IN CREATING THE APPLICATION

The hardware I used for the project was Microsoft Cortana. The other two major voice recognition apps available are Apple Siri and Google Assistant. I chose Cortana as I have a Windows 10 machine and was easily able to set it up to work with. There is no official Windows 10 Siri app and googling on how to use Siri on Windows 10 simply shows you how to rename Cortana to Siri. Google Assistant, unlike Siri, can be installed on a Windows 10 machine. However, you cannot yet develop apps for it beyond shortcuts to preexisting apps through Google actions making it severely limited compared to Cortana

ARCHITECTURE FOR THE SOLUTION

The app is built in C# using Visual Studio 2015

CONCLUSIONS & RECOMMENDATIONS

If I did this project from the beginning I would have used the mvvm model with one view for the app.

I also would have created it as a background service into Cortana, similar to asking for a story or recipe as opposed to a standalone app.

 Cortana is also a powerful tool already but struggled to understand my accent and words would be misheard. However, this was a rare occurrence and could be fixed by simply making the commands more specific. e.g. Cortana struggled when I tried to say "Try 30" along with other numbers. It also had an issue with focus as it wasn't sure what wanted to do with "try 30" especially if the app wasn't in focus. This was fixed by adding "Number genie" to the start of the command and adding an option "number" keyword. i.e. instead of saying "try number 30" you now say "Number Genie try 30", or "Number Genie try number 30". This made it easier for Cortana to understand.