Voicepad Mobile Application

M. Kamal Manisha
UG Student
Saveetha School of engineering, Saveetha University

Abstract--- This paper aims at creating a mobile application using speech recognition. A software application is developed to meet the mobile application requirements. This mobile application is named as VOICEPAD. This could be achieved using voice recognition API. The software used to develop this mobile application is Visual Express studio 2012, Windows Phone App 8 Software Development Kit (SDK), Blend Visual Studios 2012, Microsoft Paint. The coding will be done in C#. This paper describes about how the speech recognition is done using voice recognition API, how the recognized speech will be converted to text, where the text will get displayed and how the displayed text can be utilized. This paper also focuses at the working of the Windows Phone App 8 Software Development Kit (SDK), Blend Visual Studios 2012.

Keyword: - Digital diary, Voice search, Voice recorder, Google voice local search

I. INTRODUCTION

The mobile application VOICEPAD is created. As the name indicates, it reduces the typing effort of user. The speech of the end user will be recognized by application and it will be converted to text. The resultant will be displayed to the user in a multi-lined textbox. The user could share the displayed text to social networking site (Facebook), and put it up as their status or send message to their friends on Facebook.

The technology around the world had improved so much and the mobile application development is latest arena.

Countries like Korea and Japan are using voice recognition in every walk life right from search an element on internet to sending a text message and paper aims at providing the experience to Indian end users with the help of Nokia.

This mobile application is created using language C# in Windows Phone 8 Software Development Kit (SDK). The colors are given to the mobile application using Blend Visual Studios 2012 version. The mobile application is integrated and generated in Visual Express Studio 2012 version. The generated mobile App is tested in 512 MB of device using an Emulator. The developed mobile will work for Windows Phone Operating System 8 and its next higher versions. The final version of mobile application is tested in developer's environment in Nokia Windows Phone and is locked for developer.

II. DESCRIPTION

First stage of development of mobile application is to Design the User-Interface design in the Windows Phone 8 Software Development Kit (SDK's) in the designer module.

Then the required coding is developed by creating the Software through stage by stage development using C# code in XAML page. The Speech Recognition API is integrated in the mobile application with necessary header files.

The developed mobile application is run using Emulator 512 MB device. The errors are rectified and the application is saved and again tested and the process is repeated till the requirements are meted out.

Then mobile application is forwarded to Blend Visual Studios 2012 version and the colors are given to the app and images are inserted if necessary.

Fig.1: Welcome Page shows us how the app looks like as the app starts up by clicking on VOICEPAD. This page contains an "ADD NEW" button which navigates to speak now page on pressing it.

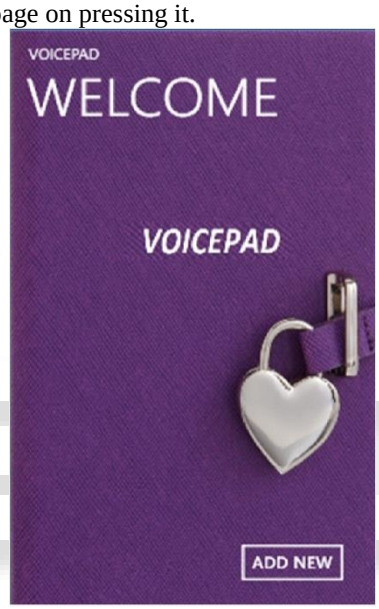

Fig. 1: Welcome Page

Fig.2: Speak Now Page contains "VOICE ON" and "SHARE" buttons which navigates to speech recognition page and Facebook login page respectively. Fig.2 also includes a text box in which converted text is displayed.

Fig. 2: Speak Now Page

Fig.3: Speech Recognition shows how the Speech Recognition API works i.e. recognizing the speech and converting it to text.

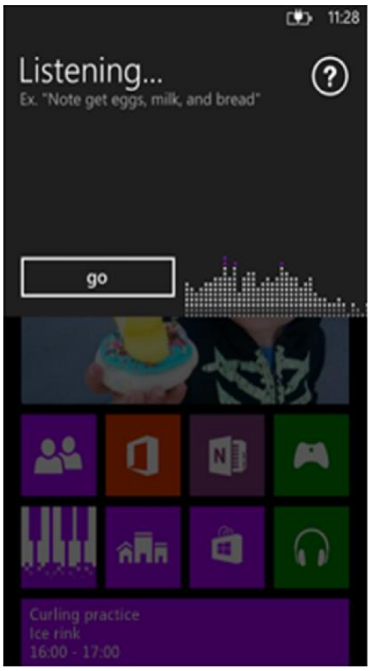

Fig. 3: Speech recognition is shown

Fig.4 shows how the text is displayed after the speech recognition is done. This displayed text will be shared to Facebook on pressing the "SHARE" button.

Fig. 4: Text Is Displayed After Speech Recognition Is Shown

The logos are created using Microsoft Paint and saved on to the App. Later, the application is sent to Nokia trainers for consideration and if even any unsorted errors will be rectified. Then the application is published on site for reviews under the given Domain with the provided details. Soon after the app will appear on the store and it can downloaded by user.

Fig. 5: Logo

A. To install Nokia SDK, a system should support

- 64bit host operating system
- At least 8GB of RAM for the host machine so that you can assign 4GB to the Virtual Machine.
- 30-40GB of free space.

III. How to use

A. Starting VOICEPAD

Click on the VOICEPAD. The application should start up and appear presently.

B. Actions

- Open a Speak Now page by pressing the "ADD NEW" button.
- Start speaking on pressing the "VOICE ON" button
  - It starts listening to the speech and stops as presently as we tend to stop speaking and starts recognizing the speech.
  - The recognized speech will be displayed within the text box in the Speak Now page.
- Share the displayed text by pressing the "SHARE" button to update it as status or to send a message.
- This "SHARE" button will automatically connect you to Facebook.

IV. Conclusion

This paper is successfully accomplished and submitted to NOKIA and it is available on APP store of WINDOWS PHONE 8. The requirements mentioned in the paper like speech recognition, speech to text conversion and text sharing are achieved successfully. And also reducing the typing effort is achieved successfully. No application is said to be completed successfully until it meets all the requirements of the end users. So, based on the comments given on VOICEPAD application further development will be done in the next version of this application.

References

[1] http://developer.windowsphone.com/en-us/downloadsdk
[2] http://www.w3schools.com
[3] http://en.wikipedia.org/wiki/C_Sharp_%28programming_language%29
[4] http://msdn.microsoft.com/en-us/library/z1zx9t92.aspx
[5] http://msdn.microsoft.com/library/prelim/csref/vcoriCReference.htm

[6] http://msdn.microsoft.com/vstudio/nextgen/technology/clangref.exe
[7] http://msdn.microsoft.com/vstudio/nextgen/technology/csharpintro.asp
[8] http://csharp.net-informations.com

