

Sifi-Save it Forget it Implementation

Dhiraj Wakodikar
Department of CSE
JIT

Sanjana Sahoo
Department of CSE
JIT

Nisha Balani
Department of CSE
JIT

Abstract :- Nowadays, every android phone has the feature of storing remainders of various events. However, providing the same feature with a voice input is very rarely available. SIFI-Save It Forget It is one such android application which enables storing of remainders related to various events such as birthdays, anniversaries, etc. using voice input. It takes as input the name of the event for which the remainder is to be generated as well as the date of the event, stores it in the calendar of the mobile phone & generates a pop-up message on the date of the event reminding the user about the event.

I. LITERATURE SURVEY

The overall goal of this project was to firstly study and understand the existing applications with the feature of storing the remainders, then identify the limitations and contribute in the same topic with greater benefits. The main advantage of our android application over the existing ones is the simple and easy to use GUI. This has enabled complete understanding and convenience on the user's side. Also, the burden of typing each event manually has been overcome as we have provided an amazing feature of voice input. Moreover, all existing applications give a reminder regarding the event on the event's date but our proposed system has extended this facility and added over the feature of automatically sending greetings or wishes to the respective event holder on user's prompt. Thus, this system with great added features will serve beneficial to the people.

II. PROPOSED SYSTEM

We propose a system which will make storing remainders an easy task.

The main features of the proposed system are as follows:

1. User Friendly:

The android app will be user friendly because the retrieval and storing of data is fast and data is maintained efficiently.

2. Greetings/Messages can be easily sent:

Default messages may be easily sent to the person for whom the event has been stored.

3. Alerts can be easily generated:

Alerts are generated easily by the app to give reminder to the user about the particular event.

4. Easy conversion of voice to text:

This app will easily recognize user voice and connect that voice into text.

5. Save time:

This app will save enough typing time & searching for events on respective days.

III. WORKING

Sifi-Save it forget it mobile application is divided into three main modules:

- Voice to text
- Text to save
- Save to send

A. Voice to text:

The user is required to dictate the message/event and his/her voice will get recognized by the application. The module translates spoken words into text by capturing the speaker's words and comparing them to preloaded words. Words that closely match those recognized by the software are converted to text and stored in the mobile phone's calendar.

B. Text to save:

This module will take converted voice to text and perform operation on that text. The inputted text is split according to the format i.e. the string will be broken down into three main units-subject, object and predicate. Nouns or Pronouns are considered to be in the subject part where as events such as birthday, anniversary, and meeting are fall under the category of objects and lastly time and date are considered as predicates. For e.g.: if we say ("I am having a meeting at 10 am"). From this, the pronoun gets separated i.e. "I" and the "meeting" will go into the object where as the "10 am" will be the predicate and the remaining left words are skipped by the application. The split text will then get saved into the calendar.

C. Save to send:

In this module, the saved entries are sent according to the user defined event. This module has two options i.e. whether to send a message or generate an alert. If an event is a birthday or an anniversary then the application will take default message from the database and ask whether to send a message or not. If user wishes to send the message then the sending process will continue and if in case user doesn't want to send a message because of low balance then the user can click on cancel button. Now the next scenario is of generating an alert. If the stored event is a

meeting then only the application will generate a pop-up message or a reminder. The complete working is explained in the following Data Flow Diagram:

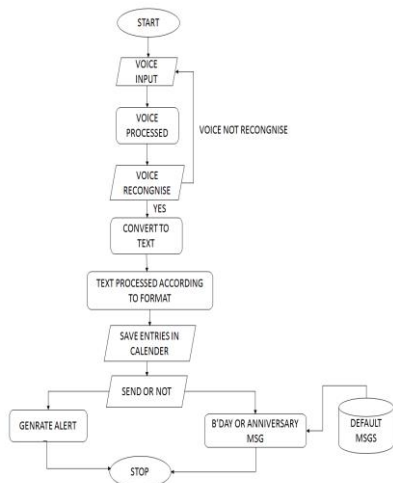


Fig-1 DFD describing the working of the system.

EXPLANATION OF DATA FLOW DIAGRAM:

The application begins by taking voice input from the user in a proper and valid format. At this point the voice get processed to match with the pre-defined natural language processing syntax and semantics. If the application validates spoken words of the user to be the correct then the flow will transfer to the next process otherwise the user has to speak again in the correct ascent. The next step is to convert the inputted voice into the text form. This step doesn't deal with any conditions, only conversion of voice to text is taking place in this phase. Then the converted text message gets saved in the calendar on the specified date. This step requires processing of converted text into the specified format. The processed text gets saved in the form of data. Now when the specified date arrives, the application will generate a pop-up message accordingly. If the event is a birthday /anniversary then the application will take default message i.e. "Happy Birthday/Anniversary" which are already stored in the database and asks the user on that date if he/she wishes to send a message to the desired person. If the event is a meeting then the application only creates an alert on that time which the user had specified earlier. In this way the flow comes to an end in a systematic manner.

RESULTS

The snapshots of the system are as follows:

1. Homepage:



Fig-2 Homepage.

The above snapshot depicts the first page of the application which the user gets to see on opening the application.

2. Main menu:

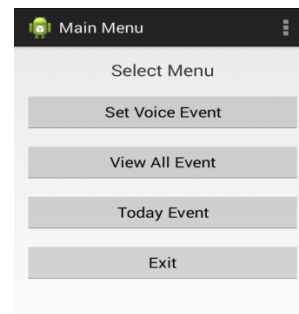


Fig:3 Main menu.

The above snapshot depicts the various options which the user can choose from and proceed further.

3. Set voice event:

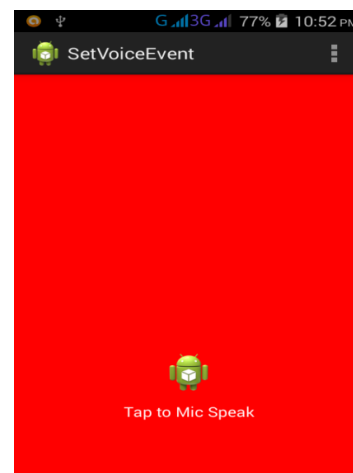


Fig-4 Tap mike to speak.

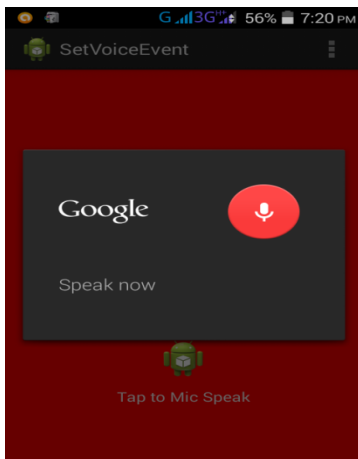


Fig-5 Give voice input.

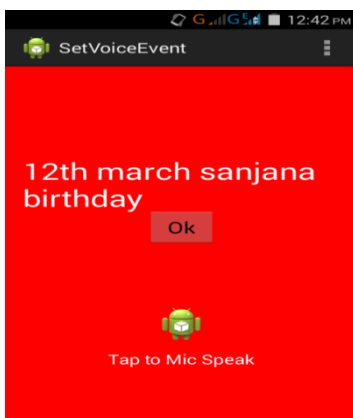


Fig-6 Voice converted into text.

The above snapshots depict how voice input is to be given to the application and how it looks after the input has been accepted.

4. EVENTS GETTING STORED IN THE CALENDAR:

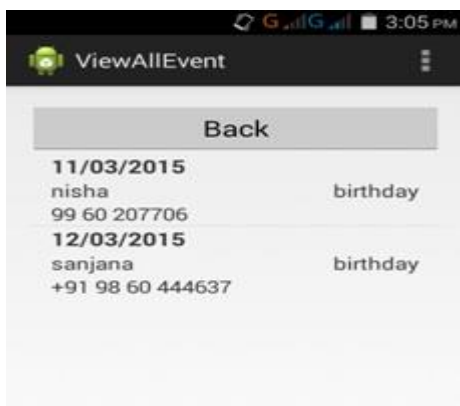


Fig-7 Events stored in calendar.

The above snapshot depicts how events will be stored in the calendar.

5. CONFIRMING FROM USER TO SEND MESSAGE OR NOT:

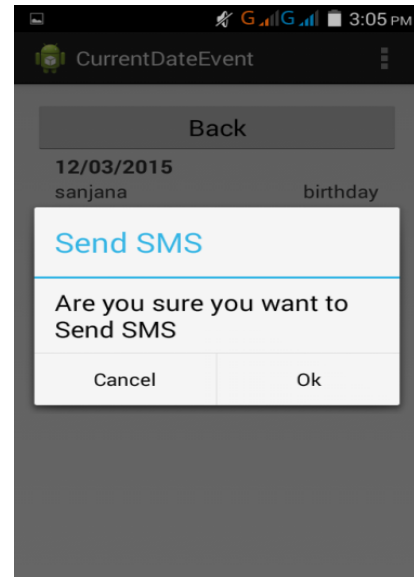


Fig-8 Confirmation of sending message.

The above snapshot depicts the confirmation asked to the user whether to send the greeting message or not. User can either accept or discard.

6. Greetings sent:

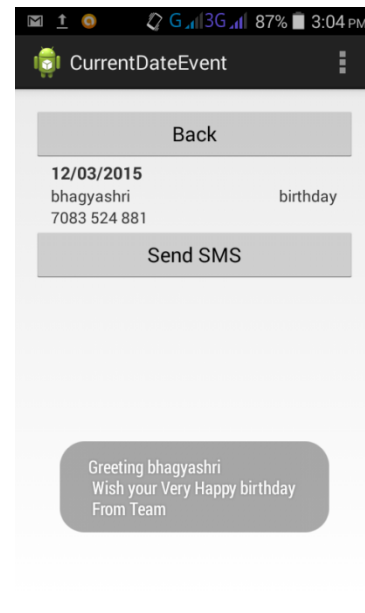


Fig-9 Current events.

The above snapshot depicts the page that will appear after greeting message has been successfully sent.

IV. CONCLUSION

This app overcomes the drawbacks of already existing reminder apps. Our app presents an overall reminder which help us to remember every important event. This app eliminates the necessity of remembering any event. Also this app save our timing because of the auto-generated message. Moreover, this app can be extended in the future to send free messages, send images, & generate alarms.

V. REFERENCES

- [1] M.Derouault and I.Mac11rialdo "Natural language modeling for Phoneme to text transcription ", IEEE Trans. Pattern anal. Machine Intell
- [2] www.androidhive.info/2014/07/android-speech-to-text-tutorial/
- [3] research.google.com/pubs/NaturalLanguageProcessing
- [4] www.thegeminigeek.com/how-speech-recognition-works/
- [5] Book Learning android second edition by Marko Gargenta, Masumi nakamura
- [6] www.Android.developer.com
- [7] Andreas Stolcke, , Barry Chen, Horacio Franco, VenkataRamana Rao Gadde, Martin Graciarena, , Mei-Yuh Hwang, Katrin Kirchhoff, , ArindamMandal, Nelson Morgan, , Xin Lei, Tim Ng, Mari Ostendorf, Kemal Sönmez, AnandVenkataraman, DimitraVergyri, and Qifeng Zhu, "Recent Innovations in Speech-to-Text Transcription at Sri-icsi-uw" *IEEE Transactions On Audio, Speech, And Language Processing*, vol. 14, no. 5, september 2006, pp 1729-1744
- [8] RyuichiNisimura, Jumpei Miyake, Hideki Kawahara and Toshio Irino, "Speech-To-Text Input Method For Web System Using Javascript", *IEEE SLT 2008* pp 209-212
- [9] A. Hagen, D. A. Connors, B. L. Pellom, The Analysis and Design of Architecture Systemsfor Speech Recognition on Modern Handheld-Computing Devices.
- [10] http://www.tutorialspoint.com/android/android_sending_sms.htm
- [11] Beginning Android™ Applicationdevelopment- Wei Meng Lee
- [12] [Android Design Patterns: Interaction Design Solutions for Developers-Greg Nudelman](#)