Paper No. T4-2.2, pp. 1-3

16th INTERNATIONAL SYMPOSIUM on POWER ELECTRONICS - Ee 2011

NOVI SAD, REPUBLIC OF SERBIA, October 26th - 28th, 2011

SOFTWARE REALISATION OF A SELF RECHARGEABLE WHEELCHAIR MODEL

Mladen Kovačev, Đuro Mijić, Radovan Čelić, Milorad Vukadinović, Vladimir Rajs, Vladimir Milosavljević

University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia

Abstract: *The aim of this project is to use smartphone as command, monitoring and emergency case device to make it easier for people with special needs to use their wheelchairs. Smartphone with Android operating system is connected to the main unit at wheelchair via bluetooth. We developed an Android application with simple user interface and various options. There are options for monitoring the battery level, driving wheelchair by touchscreen or hardware buttons, driving wheelchair by voice commands which can be usefull for people with all limbs disabled. Also there is a button for sending GPS coordinates through SMS message to specified phone number in case of emergency. The application can be installed to any mobile phone with Android operating system.*

Key Words: ***voice commands, smartphone, Android OS, SMS, GPS, bluetooth***

1. INTRODUCTION

Today, there is an expansion of smartphones, as well as different kinds of operating systems that run on these phones. Smartphones are becoming indispensable to man today and they become a mainstay for both business and private life. We used this fact and decided to use a smartphone to help people in need.

Different technologies represented in smartphones have led us to think about helping people in wheelchairs. Android operating system was chosen as the basic platform as one of the most common operating systems that smartphones use, to bring together into a meaningful whole technologies that can help people in wheelchairs. The developed application is very intuitive and easy to use and can be installed on any smartphone running on Android.

2. THE MAIN IDEA

The main idea is to connect smartphone via Bluetooth to the maincontroller unit, which controlls wheelchair, and get required information, controll wheelchair, or send an emergency SMS which contains longitude, latitude and call for help. The user interface must be simple with big buttons to minimize looking at the phone while driving. To minimize looking at the phone even more while using the application, the idea is to use speech synthesis and speech recognition. This also can be useful to people with all limbs disabled.

3. THE REALISATION

The application is developed on the Android platform using Java programming language, and the application layout is defined with an XML layout file. The application can be used with Android 2.1 and above. TTS (Text-to-speech) library is used for speech synthesis. This library allows to add speech to the application by giving a text string to TTS object which will take care of converting that string to text and speaking it to the user. For speech recognition, Speech input is integrated in application. Android SDK makes it easy to integrate speech input directly into the application. There is one application, which is pre-installed on many Android devices, and that application responds to RecognizerIntent by displaying the „Speak now“ dialog and streaming audio to Google’s server. This requires an internet connection to send data to Google’s server, and then to receive a list of possible text translations back.

The communication between smartphone and the maincontroller unit on the wheelchair is established via Bluetooth, and there is a small comunication protocol. This protocol is very simple and consists of the commands in the form of short messages. After sending the commands to the maincontroller unit, and after the maincontroller unit executes the command, it generates the answer in form of a small message. When the phone receives that small message, it must be parsed to get and show required data. For example: to get the wheelchair battery status, the phone sends „SBAT“ to the maincontroller unit. After measuring the battery level, the maincontroller unit generates a message, like „E064“ (for battery level 64%). Also, there are commands that doesn’t expect answer from maincontroller unit. This are

driving commands. Driving commands consist of four characters. The first character is the letter ‘D’ and indicates that the following character(second) determines the direction of moving ‘1’-forward, ‘2’-backward. The third character is the letter ‘T’ and indicates that the following character(fourth) determines the direction of rotation ‘1’-left, ‘2’-right. In this version there is no support for combining direction and rotation in one command, so when sending command for changing direction fourth character is always ‘0’, and when sending command for changing rotation second character is always ‘0’. Second and fourth character can be ‘3’, and this indicating maincontroller unit to stop movement or rotation, respectively. Command „D0T0“ will stop both, movement and rotation. For example: when maincontroller unit receive „D1T0“ command it will move wheelchair forward until it receive „D3T0“ or „D0T0“ command, or when maincontroler unit receive „D0T2“ command it will rotate wheelchair to the right until it receive „D0T3“ or „D0T0“ command. Pushing driving control will cause sending pre-generated commands to maincontroler unit, but when voice commands are enabled, commands are generated in the moment of receiving data from server, and recognizing appropriate string that determinate one of the commands.

GPS technology is used in the application for getting the current longitude and latitude in the moment of pushing the emergency button. The LocationManager class is used to obtain periodic updates of the device’s geographical location, and the LocationListener is used for receiving notifications from the LocationManager when the location is changed.

To send an SMS message in case of emergency, the application uses SMSManager class. The application generates an sms message text after receiving the longitude and latitude, and than sends that message to the mobile number that is chosen by the user.

4. THE APPLICATION

4.1. The main menu screen

The main menu consists of three large buttons: a button to activate on screen driving controls, a button to activate voice commands, and a button to get battery level. In the bottom of the screen there are small buttons for various options.

Fig. 1. *Main menu look*

4.2. On screen driving controls

A dialog with very simple and intuitive graphical user interface. The buttons on this dialog are very big to provide safer driving and minimalize looking at the phone.

Fig. 2. *On screen driving controlls*

4.3. Settings dialog

There are only two settings: the emergency mobile phone number and the wheelchair speed.

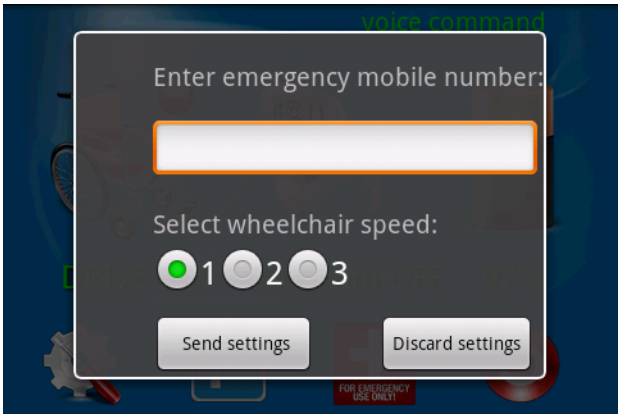

Fig. 3. *Settings dialog*

4.4. Speech input

This dialog is a result of responding of pre-installed application to RecognizerIntent. The small microphone in the middle of the dialog shows the sound level when speaking.

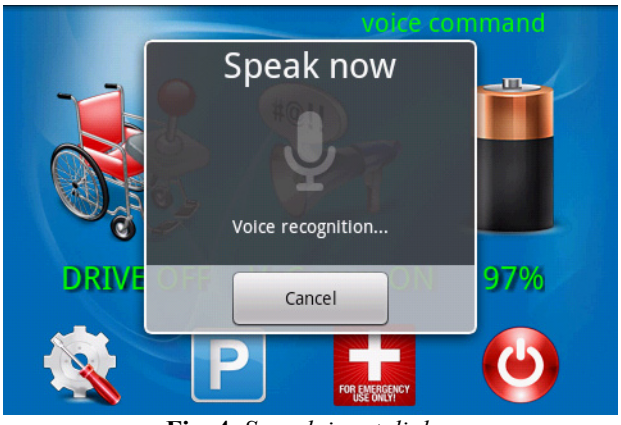

Fig. 4. *Speech input dialog*

5. TESTING AND CONCLUSION

The application is tested along with the small model of wheelchair in form of a toy car. The bluetooth is fast and reliable enough, and along with a small comunication protocol implemented in application makes the comunication between the smartphone and the maincontroller unit on wheelchair very fast and interactive. Getting the information about current geographical location in the moment of pushing the emergency button, and generating and sending the emergency message is done in about 3 to 4 seconds. The voice recognition is a bit slow, because the recorded audio must be sent to the server, and then the application is waiting for a response, but it is good enough for proof of concept. There are several libraries that can be used to recognize the speech without internet connection, but all of them are work in progress. The speech synthesis is good, and it is very useful because after sending the request for some data, the user doesn't need to look at the phone and he can concentrate on driving.

6. REFERENCES

[1] Reto Meier, "*Profesional Android 2 Application Development*", Wrox Programmer to Programmer

[2] http://developer.android.com – The Developer's Guide

[3] Reto Meier, "*Profesional Android Application Development*", Wrox Programmer to Programmer

[4] Marko Gargenta, "*Learning Android*"