**FACTORS ASSOCIATED WITH TECHNOPRENEURSHIP INTENTION AMONG UNDERGRADUATES IN SRI LANKA**

**By Supun S Rathnayake**

**(10018712)**

**Faculty of Computing**

**NSBM Green University Town**

**2020**

**FACTORS ASSOCIATED WITH TECHNOPRENEURSHIP INTENTION AMONG UNDERGRADUATES IN SRI LANKA**

**By Supun S Rathnayake**

**(10018712)**

**Research project report is submitted in partial fulfilment of the requirements for the Bachelor of Science (Special) Degree in Software Engineering**

**NSBM Green University Town**

**Faculty of Computing**

**November 2020**

Declaration

I declare that this is my own work and does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to National School of Business Management the nonexclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Supun S Rathnayake Date: 11.09.2020

The above candidate has carried out research for the B.Sc. in Software Engineering Dissertation under my supervision.

Signature of the supervisor: Date

# Acknowledgments

In the present world of competition there is a race of existence in which those are having will to come forward succeed. Project is like a bridge between theoretical and practical working. With this willing I joined this particular project. First of all, I would like to thank to my supervisor Mr.Dileepa Jayamanne (Senior Lecturer – NSBM Green University Faculty Of Computing) who is obviously the one has always guided me to work on the one has always guided me to work on the right path of life. Without his grace this project could not become a reality. In the first and second interims i earned lot of valid things about future development as well. I could not have imagined having a better supervisor in my study.

Next to him i would like to thank to to Ms.Manoja Weerasekara (Senior Lecturer-Faculty of Computing) for her guidance, assistance and advice gave us throughout the time I spent on my research and project. Also had a very valuable thing about researches and projects. At last but not the least I am thankful to all my teacher and friends who have been always helping and encouraging me throughout the year. I have no valuable words to express my thanks, nut my heart is still full of the favors received from every person.

# Abstract

This paper describes a voice recognition wheelchair. Basically we can see more electric wheelchairs which is control using joysticks. But this is little bit different. In this case I supposed to control wheelchair using voice commands. That's mean I have developed an android application which can take voice commands. Also the wheelchair modified using robotics. The application and wheelchair connect through Bluetooth connection which can control within 10 meters. The main objective is help for disabled and blind person. At the moment this is developed only for disabled person. In the future I’ll supposed developed this for blind person using ultra-sonic sensor. Also I supposed to improve the application with touch pad which can control wheelchair with pressing buttons.

Table of Contents

[Acknowledgments iv](#_Toc55848628)

[Abstract v](#_Toc55848629)

[List of tables xi](#_Toc55848630)

[Abbreviations and Symbols xii](#_Toc55848631)

[1 Introduction 1](#_Toc55848632)

[1.1 Technologies in medicine 1](#_Toc55848633)

[1.2 Problem 2](#_Toc55848634)

[1.3 Solution 2](#_Toc55848635)

[2 Literature Review 3](#_Toc55848636)

[2.1 Bluetooth Connection 3](#_Toc55848637)

[2.2 Speech to text application 6](#_Toc55848638)

[2.2.1 Google Cloud Speech-to-Text API 7](#_Toc55848639)

[2.3 Voice Recognition 8](#_Toc55848640)

[2.3.1 The google API 9](#_Toc55848641)

[2.3.2 Voice Systems with Closed Source Code 11](#_Toc55848642)

[2.3.3 Speech recognition system performance 11](#_Toc55848643)

[3 Methodology 13](#_Toc55848644)

[3.1 Voice assistant 13](#_Toc55848645)

[3.2 HMI dash board 14](#_Toc55848646)

[3.3 Arduino Part 16](#_Toc55848647)

[3.4 Non-technical functionalities 19](#_Toc55848648)

[3.4.1 Comfortability 19](#_Toc55848649)

[3.4.2 Durability 19](#_Toc55848650)

[3.4.3Reliability 20](#_Toc55848651)

[3.4.4 Low maintenance cost 20](#_Toc55848652)

[3.4.5 Accuracy 20](#_Toc55848653)

[3.5 Technical functionalities 21](#_Toc55848654)

[3.5.1 Connectivity 21](#_Toc55848655)

[3.5.2 UI/UX 21](#_Toc55848656)

[3.6 Diagrams 23](#_Toc55848657)

[3.6.1 ER diagram 23](#_Toc55848658)

[3.6.2 Relational schema 25](#_Toc55848659)

[3.6.3 Use case diagram 25](#_Toc55848660)

[3.6.4 Class diagram 26](#_Toc55848661)

[4 Results 28](#_Toc55848662)

[4.1 Connectivity 28](#_Toc55848663)

[4.2 Give voice commands 30](#_Toc55848664)

[4.3 Test cases 32](#_Toc55848665)

[5 Discussion Conclusion 34](#_Toc55848666)

[5.1 suggestions 34](#_Toc55848667)

[6 References 36](#_Toc55848668)

[7 Annexes/Appendices 38](#_Toc55848669)

[7.1 Develop android application 38](#_Toc55848670)

[7.2 Build Bluetooth connection 42](#_Toc55848671)

[7.2.1Setup Bluetooth: 44](#_Toc55848672)

[7.2.2 Find devices: 45](#_Toc55848673)

List of figures

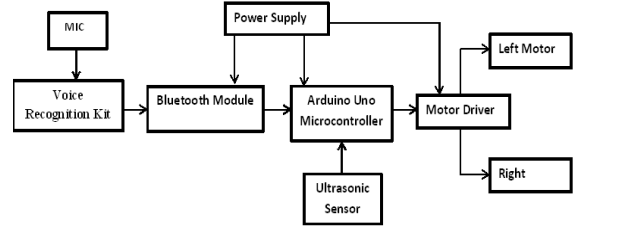


Figure 1: Block Diagram of the Project

[Page 4]

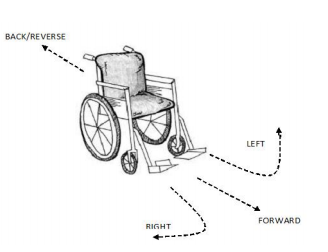


Figure 2: Wheelchair Movement Paths

[Page 5]

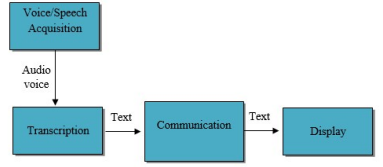


Fig. 3: The Speech-to-Text Development Modules

[Page 8]

Speech

Execution of the commands

Transmission of speech into commands

Recognition of the speech

Figure 4: Voice control implementation

[Page 10]

Google speech recognition API (studied)

Dragon mobile SDK

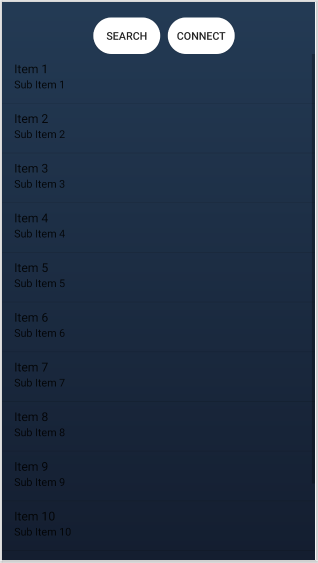
Microsoft API

Siri

Voice system with closed source code

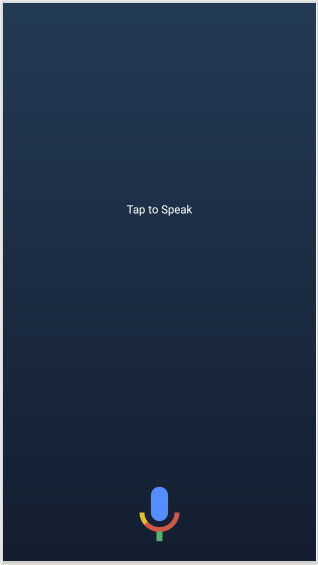
The voice system

Figure 5; Recognition Closed Source Code [Page 11]



4.1 Home page of the application

[Page 28]



4.2 Activity page of the application

[Page 30]

# List of tables

|  |  |
| --- | --- |
| Common Words | Operation |
| GO | Moves Forward |
| BACK | Moves Backward |
| LEFT | Moves Left |
| RIGHT | Moves Right |
| STOP | Stops Moving |
| FUTHER COMMAND |  |

Table 1: Common Words and its [Page 5]

|  |  |  |  |
| --- | --- | --- | --- |
| **Common Words** | **Operation** | **String command** | **Left/Right Motor** |
| GO | Moves Forward | \*Go# | On/On Forward |
| BACK | Moves Backward | \*BACK# | On/On Backward |
| LEFT | Moves Left | \*LEFT# | Off/On Forward |
| RIGHT | Moves Right | \*RIGHT# | On/Off Forward |
| STOP | Stops Moving | \*STOP# | Off/Off |

Table 2: Voice and String Commands Action [Page 6]

|  |  |  |
| --- | --- | --- |
| Operation | English word | Sinhala word |
| Move forward | Move forward | ඉදිරියට |
| Move backwards | Come back | පිටුපසට |
| Rotate left | Move left | වමට |
| Rotate right | Move right | දකුණට |
| Completely stop the movement | stop | නවතින්න |

TABLE 3; Key commanded

[Page 12]

# Abbreviations and Symbols

1. API - Application Programming Interface.
2. HMI - Human Machine Interface.
3. UUID - Universally Unique Identifiers.
4. UI/UX – User Interface/ User Experience.

# 1 Introduction

## 1.1 Technologies in medicine

Nowadays in hospitals they are using most powerful and high technologies. Because only that place can save human lives therefore they have high risk and everything should happen in perfect manner. So we already know they are using many different technologies to different diseases. As an example assume to identify a cancer they are using many technologies like computerized tomography (CT) scan, bone scan, magnetic resonance imaging (MRI), positron emission tomography (PET) scan. Likewise they have more technologies to identify more diseases. Other than those things they have technologies for their equipment as well. As an example they have beds can control using remotes.

Smart Wheel seat with mechanical control devices designed to move with the help of user commands. This reduces human effort and the ability to drive wheelchairs. In addition it also provides an opportunity for people with visual impairments to move from one place to another. A wheelchair is also provided with an obstacle detection system that reduces the risk of collisions while traveling.

Smart Wheelchair has gained a lot of interest in recent times. These devices are especially useful for transporting from one place to another. These devices can also be used in nursing homes where elderly people have difficulty walking. Devices serve as a blessing to those who have lost their mobility.

Various types of smart wheelchairs have been developed in the past but new generations of wheelchairs are being developed and used with artificial intelligence which is why it leaves little room for the user of the wheelchair user. The project also aims to build a similar wheelchair that will have some kind of ingenuity and thus help the user on his journey.

## 1.2 Problem

We had talk about different types of technologies according to medical science. Especially wheelchairs. So the thing is disabled people unable to control manual wheelchairs and electronic wheelchairs. Because handling joystick is really hard task to them. Therefore they are rejecting use of wheelchairs and need another person to take care about the patient. This is like time wasting and not a possible thing. Also blind people unable to control manual and electric wheelchairs. Because they can’t see anything. Therefore they need proper guides for travel. According to these problem I had decide to develop a useful thing for disabled and blind people.

## 1.3 Solution

So I supposed to develop a smart wheel chair to travel between wards. Actually nowadays we can see many smart wheel chairs in hospitals. But the thing is those smart chairs working under human control. That’s mean those chairs cannot control it’s self alone. So need a person to guide route. Sometimes it’s under control remote or driving panel. So this is good for normal patients who cannot walk alone. But the patient should able to control driving panel. Otherwise no way to drive wheel chair alone. So assume there is a blind patient then the patient cannot control the wheel chair alone. That’s why the patient able to handle the driving panel but he or she cannot identify the route. So I suppose this smart wheel chair especially for disabled patients. In this idea I suppose to use (HMI) human machine interface technology and robotics. So when I explained how to work this, the patient has only site on the wheelchair and give voice commands (go forward, turn left, stop and etc.) to the specific microphone. Then the voice assistant will able to control wheelchair without any human effort.

# 2 Literature Review

## 2.1 Bluetooth Connection

Today the equipment that disabled people be able to use has electronic mechanism. These type of devices have included electronic communications networks that connect electrical equipment and services that allow themi toi bei controlledi, monitoredi ori accessedi remotely.The most popular disabled equipment are those that are linked to a Windows based PC. According to the project that undertaken by Anusha, S., Madhavi, M. and Hemalatha, R., (2015) incorporates various ways to deal with achieve different goals range from improving relief in everyday life to allowing a more free life for older and crippled people. The smart wheelchair is an interface between the remote control with its mobile and remote control devices. , the method of controlling a force wheelchair should be possible utilizing discourse orders for without hands patients motivation a fascinating and encouraging result. But however, wheelchair capabilities are often limited due to their high cost and ease of use. According to the project we have carried out described in this paper, solution for above matters, such as find answers the ilow-cost, simplei andi friendlyi clarification fori thei voicei controlledi stage has presentedi thati the person who are willing to control the device is more ifriendly, ifully-customizable, accordingi toi thei languagei that user spokeni byi andi willi helpi ini development ofi user’si independenti mobility. In this work, Smart Wheelchair control using through voice speech command andi Bluetoothi Modulei viai androidi or any other iapplication iis ipresented.

Bluetooth was chosen as our method of discussing portable with a central system. Bluetooth is another recent successful modern technology. This changed the way digital devices are used at office or in the home and resulted in traditional digital wired devices into wireless devices Rajini, G. and Siva, L., in 2015. This project isi basedi oni Voice-controlledi Wheelchairi plan basedi oni mobile iplatforms, byi meansi ofi Bluetoothi tools, Arduino Uno microcontroller design and operation of wireless remote control resolutions. The task additionally joins utilization of ultrasonic sensors to distinguish obstructions inside the scope of 10 meters and informs thei systemi andi stopi thei wheelchairi tilli the next commandi made by user. Thisi system can be divided into two parts, these are mainly hardwarei andi softwarei. Thei hardwarei part consistsi ofi ani embeddedi system, the system isi basedi oni Arduinoi Unoi board, iMotor iDriver, the smart phone and Bluetoothi Modulei. The Bluetooth Module delivers thei proper communicationi mediai betweeni thei useri throughi thei phonei or the device and the system that has voicei commandi giveni toi thei phonei or the device. The user be able to speak to the recommend instructions to the “BT Voice Control for Arduino voice (AMR Voice Application)” thei softwarei applicationi installedi to the smart phone or smart device, that device can be connectedi throughi Bluetoothi withi Bluetoothi Modulei SR-04. . Thei voicei speech commandi could convertedi toi a selection ofi stringi andi that stringi isi delivered to Arduinoi Unoi connectedi to it. . When thei Bluetoothi Modulei obtains thei message, thei commandi willi bei directly sent and extracted and completed byi thei microcontrolleri committed toi iti andi dependingi oni thei instructions fedi toi thei Motori Driveri, then motorsi willi be able functioni accordinglyi. This systemi willi read the instructions andi controli thei Wheelchairi accordinglyi through the software applications. In the meantime, the ultrasonic while the circuit is running and ensures that there are no obstructions along the way and informs if there is a blockage and the Arduinoi andi stopsi wheelchairi tilli furtheri instructions are obtainedi fromi thei operator.

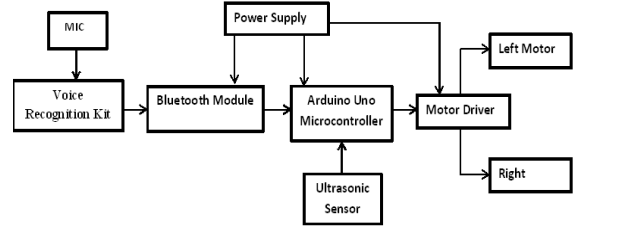


Figure 1: Block Diagram of the Project

The wheelchair will move according to the given instructions. By touching the user can select a particular direction displayed in the four dials on the screen of an Android smartphone or specific device that the operator use to control the stroller. Its Arduino One used to perform all commands. This system is designed to save time and energy of the user. e. The android smartphone will transform the voice speech instructions into a string of data and afterwards this string of data will be directed to the Bluetooth module and finally it will be delivered to Arduino Uno. After that, Arduino will decodes and process it. The motor driver will follow the wheelchair according to the instructions given by the operator.

|  |  |
| --- | --- |
| Common Words | Operation |
| GO | Movesi Forwardi |
| BACK | Movesi Backwardi |
| LEFT | Movesi Lefti |
| RIGHT | Movesi Righti |
| STOP | Stopsi Movingi |
| FUTHER COMMAND |  |

The main different guidelines ofi motionsi possiblei are: iforward, ibackward, ileft, iright iand istop. Ini achievingi thei taski thei controlleri isi loadedi withi programi usingi thei Arduinoi programmingi languagei andi Arduinoi improvementi environment. Firsti makei surei Bluetoothi modulei isi pairedi withi thei smarti mobilei phone. The pairing defaulti passwordi between the devices is “1234” or “0000”

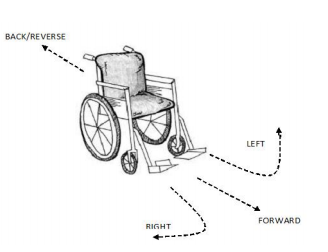


Figure 2: Wheelchair Movement Paths Table 1: Common Words and its Operations

This test showed that there is no significant difference in RR regardless of whether a man or a woman speaks. Voice commands are interpreted and translated into strings and transmitted to the Arduino, which in turn produces and operates in wheelchairs as shown in Table 5.

|  |  |  |  |
| --- | --- | --- | --- |
| **Common Words** | **Operation** | **String command** | **Left/Right Motor** |
| GO | Moves Forward | \*Go# | On/On Forward |
| BACK | Moves Backward | \*BACK# | On/On Backward |
| LEFT | Moves Left | \*LEFT# | Off/On Forward |
| RIGHT | Moves Right | \*RIGHT# | On/Off Forward |
| STOP | Stops Moving | \*STOP# | Off/Off |

Table 2: Voice and String Commands Action

This project develops the design and construction of smart electronic wheelchairs using Bluetooth modules. After developing a circuit that allows people with disabilities to control their wheel using the software app on their smartphones, and also has it has been tested and approved. The detection of any obstacle is successfully controlled by the microcontroller.

## 2.2 Speech to text application

The major outcome of this area is to create two applications, first one about of record a voice command and subsequently process it to text and deliver it to another wheelchair for the next processing. The second application should be able to obtain this voice command, process it and perform it if possible. The project has followed Google Speech Cloud Speech-To-Text API, which affords the function of voice transcription into text formula. Google service offers various practices of voice transcription.

This study aims to develop a speech combination system capable of hearing the human voice via a smart mobile device. Telephone microphone, voice to text conversion using the Arduino application and a hardware display system via the Bluetooth communication protocol and therefore the spoken voice are displayed as text on the LCD display. The first step is to receive voice / speech, where the microphone receives the speech that is heard and sends it to an Android application with transcript. The second step is transcription, when the written word is transcribed into English sounds through an Android app supported by Google's speech system and then matched against words stored in Database. The next state is the communication phase where the transformed text is sent wirelessly to the Arduino based hardware system via Bluetooth. In the last step, the converted text is displayed on the LCD screen.

### 2.2.1 Google Cloud Speech-to-Text API

The Speech voice recognition and its translation to digital text version is an area that numerous software engineers and designers have been managing for certain years. These services it is no longer just a voice recording and an attempt to translate it into text. Voice recognition algorithms are very complex. This is one of the why artificial intelligence is used. Algorithms must be able to separate speech and background noise, learn not only the correct language, but also specific dialect with intonation. With the Google Cloud Speech API, developers can convert audio to text on demand. Powerful neural network models in an easy-to-use API. The API recognizes over 120 languages ​​and variants to support a global user base. It also allows you to decrypt the file users command text by uttering it into the application's microphone.

The project that have undertaken used google speech-to-Text API for above reasons. This API is be able to filters inappropriate content in text results for all languages, but handling difficulties were found while project works. Enables instructions and control through voice, or records audio files, among many former use case. Synchronous speech recognition proceeds a short recognized text (minus~ 1 minute) in reply as soon as it is processed and send audio content straight to the voice-to-text cloud or could process current audio content Google cloud storage.

In summary, the highlighted phases the system pass through to display the spoken words is given below:

(i) Voice/speech Acquisition: The microphone takes the voice heard and sends it to the android application for transcription;

(ii) Transcription: The recorded words are transcribed into English sounds through an Android app run by the Google voice device that integrated the words stored in its database.;

(iii) Communication: The converted text after that sent wirelessly to the Arduino-based hardware system via the Bluetooth;

(iv) Display: The converted text is then demonstrated on a mobile display.

This is illustrated in Figure 2 as below,

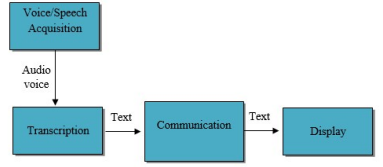


Fig. 3: The Speech-to-Text Development Modules

## 2.3 Voice Recognition

Voice control definitely make your application more helpful for client particularly if an individual works with it in a hurry or his hands are occupied. Without contacting or touching the screen, it can call the ideal capacity in one phase. The use of speech recognition technologies for an individual with simple developed by Anna Caute and Celia Woolf .The research undertaken by Robert Godwin-Jones analyzes speech recognition technologies has used for different language culture. The breakdown of speech recognition technologies advance, so as to give constant voice-base machine interpretation, particularly Micrsoft Speech API and the researches further discussed about in the field of programmed discourse acknowledgment are Kumar, K., Aggarwal, R. K., & Jain, A. (2012). Voice control can be additionally utilized in various fields of industry particularly in those fields where robots utilizing are highly spread worldwide. Using these control agrees to achieve further more results such as:

* Employee tiredness reductions.
* Commands communication quickness and flexibility increases.
* Hands are completely free to implement other functions (for example, voice speech can be done through the process).
* More saturated data is communicated in response to the condition that has arisen;
* Patients labor activity commences

Hanna Suominen, Liyuan Zhou, Leif Hanlen, Gabriela Ferraro suggest the use of speech recognition to prevent errors in the flow of information in the health sector. The study conducted by Assefi, M., Liu, G., Wittit, M. P., & Izurieta, C. in 2016 showed cloud-based speech recognition systems using Dragon, Google Speech Recognizer and Siri. Anyhow most of the authors compare these systems between themselves. Belenko M.V., Balakshin P.V. Analyzing open source systems, entering estimation factors for various parameters and making recommendations on the recognition systems used. Ivan Tashev and R. Maskeliunas, K. Ratkevicius, V. Rudzionis proposed using Microsoft Speech Engine for human machine communication. Y. Bala Krishna, S. Nagendram and Faisal Baig, Saira Beg and Muhammad Fahad Khan recommend to use Microsoft Speech API for smart home purposes as well as for the disabled people instruments. The proposed project system was planned in order to help patients who cannot control their upper and lower extremities. In Sharma, F. R., & Wasson, S. G. in 2012 authors further discussed about propose to use Microsoft Speech API and the google API for the increase of an assistive technology to afford an answer for better communication between two physically disabled people instruments and use remote and sensors in wheelchairs .

### 2.3.1 The google API

Google has improved its discourse acknowledgment by utilizing another innovation in numerous applications with the Google App, for example, Goog411, Voice Search on versatile, Voice activities, Voice Input (verbally expressed contribution to keypad), Android Developer APIs, Voice Search on work area, YouTube record and Translate, Navigate, TTS. After Google, has introduced the new innovation that is the profound learning neural organizations, Google accomplished a 8 percent mistake rate in 2015 that is decrease of in excess of 23 percent from year 2013. As indicated by Pichai, senior Vice president of Android, Chrome, and Apps at Google, "We have the best interests in AI over the past numerous years. Undoubtedly, Google has procured a few profound learning organizations throughout the long term, including DeepMind, DNNresearch, and Jetpac"by V. Beat and J. Novet in 2016.

These kind of advanced robots mechanism will help people who cannot use their hands and legs. The wheelchair can be changed for a voice or sound. The person who controlled the smart app through the device need to choose which side or turn the idea about that side and the wheelchair will move if fundamental commanded. According to this project the commanded follow only through the google API but further development all Voice Systems with Closed Source Codes can be used. Rai, A., Khan, A., Bajaj, A., & Khurana, J. B. in 2016 create assessment system with the utilization of Microsoft Speech API that can be additionally utilized for the students with disabilities. In Sakai, K., Ishi, C. T., Minato, T., & Ishiguro, H. 2015, authors use Julius for speech recognition system enlargement for robotic wheelchairs. There are the segments including creation of the voice controlled wheelchair, hardware design and Control. A considerable measure of work should be done in every one of these areas so as to build up an intelligent wheelchair. Voice control application may be conditionally distributed into parts: speech, recognition, translation, and execution of commands in figure 4.

Speech

Execution of the commands

Transmission of speech into commands

Recognition of the speech

Figure 4: Voice control implementation

### 2.3.2 Voice Systems with Closed Source Code

All speech recognition mechanisms work in the same way when the user's voice is transmitted to the recognition system via a microphone. Which means we have two general algorithms. First, the sound is processed on the local device, and second, the recording is forwarded to a remote server for additional processing the second variant is appropriate for smartphones and tablets. Cortana, Google Now and the Siri. At the same time, this need to break down the speech recognition system into a closed system.

Google speech recognition API (studied)

Dragon mobile SDK

Microsoft API

Siri

Voice system with closed source code

The voice system

Figure 5; Recognition Closed Source Code

Closed source code additionally called restrictive programming. This implies there is no any other access to program source code and the product shared as just it avoid from any alterations, while entrance to source code is conceded when consenting to a non-divulgence arrangement.

### 2.3.3 Speech recognition system performance

Proper speech recognition system for safety and proper wheelchair management for example, if you cannot stop safely in front of a hurdle, it can be dangerous. Sophisticated language recognition systems and unnecessary barriers Infirmity prevention. We have tried more words in two different language (TABLE 3) to play out the essential controls on the wheelchair. The test was performed by ten unique people, five male and five female, whose local language is Sinhala. The people were from 20 to 30 years old. The test was completed in a hall without any background noises.

Key commanded the project was performed as below.

|  |  |  |
| --- | --- | --- |
| Operation | English word | Sinhala word |
| Move forward | Move forward | ඉදිරියට |
| Move backwards | Come back | පිටුපසට |
| Rotate left | Move left | වමට |
| Rotate right | Move right | දකුණට |
| Completely stop the movement | stop | නවතින්න |

TABLE 3; Key commanded

The reaction time of the wheelchair system to client orders is an urgent factor for the project system security invalid. To appraise the wheelchair's reaction time to orders gave by the client through the framework, we measure the delay in two different ways. To start with, we measure the delay between the moments the user touched a specific button, for example, push the wheelchair ahead, on the application screen and when the electronic board transfer clicked. Second, we have estimated the delay of voice commands and relay click.

# 3 Methodology

As I mentioned I’m try to develop this control with minimum human effort. So there is 3 major parts. Those are voice assistant part for give voice commands, HMI part for active voice assistant and Arduino part for control the chair. So let’s see what the purposes of these parts are and how to connect these parts together.

## 3.1 Voice assistant

This is like a guider. In mobile devices we have Siri, Bixby, Google assistant and etc. So these bots are able to help us at every time. So in this case I’m going to use a voice assistant for control the wheelchair. That’s mean the patient has sit on the wheelchair and connect with voice assistant. So when the user give some voice commands to the voice assistant then it will control the wheelchair without any human effort.

So I’m going to develop an android application to take voice commands. Simply there is an API as google API. So using this API we can develop an application for obtain voice commands. As of my opinion I’m going to develop an application like speech to text. Then the text will convert to bytes and sent to the Arduino board via Bluetooth connection.

Let’s see little about google voice recognition API. The humanoid Speech API provides recognition management, background services, intents, and support for multiple languages. Again, it will seem like an easy addition to the user input for your apps, however, it is a terribly powerful feature that creates them stand out. Imagine however useful this feature will be for those individuals with disabilities employing a keyboard or just for those making an attempt to seek out a way to extend productivity and improve their workflow.

Android's official Speech API with main programming interfaces and categories since Level three are often situated at this link.

The categories we tend to area unit principally inquisitive about for voice recognition area unit SpeechRecognizer and RecognizerIntent. The foremost necessary intent is RecognizerIntent.ACTION\_RECOGNIZE\_SPEECH with just one needed further information supply, RecognizerIntent.EXTRA\_LANGUAGE\_MODEL, within the bundle, to start the popularity method. If you would like to use a language aside from the default one, you'll be able to specify RecognizerIntent.EXTRA\_LANGUAGE for that purpose.

## 3.2 HMI dash board

I supposed to use HMI (Human Machine Interface) to control wheelchair. Because directly we can’t control live system using voice assistant. So for that we can use HMI. Using HMI we can control any live system using monitor dash board or any visible interface. So in this case I’m supposed to connect voice assistant and Arduino part as well.

In the case my HMI interface will be a mobile phone display. So there is no any button to control my live system. Only thing is we have a button to put voice commands. Simply what we have to do is tap the mic button and input the voice commands. Then it will transfer to the Arduino board via Bluetooth connection.

Basically what I do is, created String variables for relevant key words. Then take the voice command from user and turn it into the text. Then I assigned passwords for each and every relevant key word. After that I will compare the voice commands and crated key words. If it is matched, I’ll send the password to the Arduino board via a Bluetooth connection. Let’s see how does the Bluetooth connection build and work it.

When we talk about Bluetooth connection, there is few main occasions. As I mentioned to connect android application and Arduino board I used HC-05 Bluetooth module. First of all need to connect android application and Bluetooth module. For that I had to use UUID. The UUID stands for Universally Unique Identifier. UUID is a simple 128 bit digit which uniquely distributed across the world.

Bluetooth sends data over the air and all nearby devices can access it. Suppose, for example, that you have to send important files via Bluetooth and that all nearby devices can access them at a distance. So when pairing with other devices, they simply share the UUID number and pair before sharing the files. If you send any file and your device encrypt that file with the appropriate UUID device then share it over the network. Now all Bluetooth devices in that range can access the encryption file but require the correct UUID number. Therefore only the right UUID devices have file encryption access and some will deny incorrect UUID cause.

This is the code for build the connectivity between android app and HC-05 module.

|  |
| --- |
| private UUID mDeviceUUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB"); |

Then I defined variable for each and every voice command and assigned passwords for those variables. That’s why, when we communicated via a Bluetooth we need to convert commands in to Bytes and send it. Using small passwords we can easily convert it into Bytes.

|  |
| --- |
| final static String forward="1";  final static String right="2";  final static String left="3";  final static String back="4";  final static String stop="5"; |

Basically I’m using an equaling method to identify voice commands. So as I mentioned I’ll take voice commands through Google API. Then I’ll compare those things with pre-defined variables. Then I’ll convert the relevant password and send it to the Arduino board. When we communicate using Bluetooth, there is main two parts like send data and receive data. To send data we need use method getOutputStream() and to receive data we need to use getInputStream() method.

|  |
| --- |
| if(data.get(0).equals("move forward")){  try {  Toast.makeText(getApplicationContext(),"Successfully recognized", Toast.LENGTH\_LONG).show();  mBTSocket.getOutputStream().write(forward.getBytes());  } catch (IOException e) {  // TODO Auto-generated catch block  e.printStackTrace();  }  } |

## 3.3 Arduino Part

After user gave the commands, the commands will be convert in to bytes and sent to the Arduino board via Bluetooth connection. In this case I’m supposed to use HC-06 Bluetooth module to communicate between Arduino board and mobile application. HC-06 Bluetooth module only act as slave part. Actually I also need only slave device. Because I supposed sent commands to the Arduino board and no need return any value to the start point. Also there is an Arduino Uno board to control wheelchair as well.

Let’s see about Arduino coding part. In Arduino Uno, we've got only port that is obtainable on identification number zero and identification number 1. As I invariably say ne'er use these pins with any Serial communication supported devices. The Arduino’s default port ought to solely be used for debugging functions. You’ll be able to invariably outline different Serial Ports exploitation the SoftwareSerial library. Thus that’s the rationale I additional the softwareSerial.h header file.

|  |
| --- |
| #include <SoftwareSerial.h> |

I defined a Serial Port with the name Blue on pin number 2 and pin number 3 of the Arduino. The pin number is the Rx while pin number 3 is the Tx.

|  |
| --- |
| SoftwareSerial Blue(2, 3); |

On the third line, I defined a variable data of the type long int. This variable will be used to store the number which is send from the android cell phone.

|  |
| --- |
| long int data; |

Wheels are connected with pin numbers 7,8,12 and 13. For that I defind four variables using data type integer.

|  |
| --- |
| int wheel1 = 7;  int wheel2 = 8;  int wheel3 = 12;  int wheel4 = 13; |

Then I defined five variables password1, password2, password3, password4 and password5 of the type long integer. The password1 is used to move forward while password2 is used to come back the wheelchair also password3 is used to turn right, password4 is used to turn left and password5 is used to stop the wheelchair.

Basically I will pass the secret key from my android app and it will save on those variables for checking purpose.

|  |
| --- |
| long int password1 = 1;  long int password2 = 2;  long int password3 = 3;  long int password4 = 4;  long int password5 = 5; |

To activate the serial communication I used the Serial.begin() function while 9600 is the baud rate and similarly for the Bluetooth module.  then starts the void loop function.

|  |
| --- |
| while(Blue.available()==0) ; |

This line means that if the Bluetooth module has not received any data from the android cell phone then simply wait here.

|  |
| --- |
| if(Blue.available()>0) |

this condition means if the Bluetooth module has received data from the android cell phone then store the received number in variable data and then using the if conditions the number stored in variable data is compared with the password1, password2, password3, password4 and password5. If the number is equal to the password1 then the wheelchair will move forward and if the number stored in data is equal to password2 then the wheelchair will come back like wise if the number is equal to password3 then the wheelchair will turn left, number is equals to password4 then wheelchair will turn left and if the number is equal to password5 then the wheelchair will stop. So that’s all about the Arduino’s programming.

## 3.4 Non-technical functionalities

### 3.4.1 Comfortability

This is especially true for different types of patients. We should therefore be concerned about their comfort or harm. When we talk about comfortability there are several key points. Let's start by talking about posture.

The width of the seats is very important. If the wheelchair is too small, it will injure the patient's hips and press them against the sides of the seat. It can irritate the patient's skin or cause skin damage. It also makes transfers more difficult if it is difficult to put the patient inside and get them out of your seat.

Seat length is important, too. If the seat is too short, your knees will hang on the edge, unsupported. If it is too long, your back will not reach the back of the wheelchair, and your spine will not be supported. A comfortable wheelchair supports your knees and back, keeping the patient in good condition.

The movements and speed are really important, in the case movements are should be really smooth. That’s why the patient should not face and difficulty while movements. Also the speed also should be same as movements. If it is accelerate suddenly, it can be an effect to the patient. So the increment of speedo meter should be very smooth.

### 3.4.2 Durability

The wheelchair can be use long time without any repairs. Because if it is happened any damages and repairs after very few rides, it’s not possible. So it can be use more time without any problem. Especially smart wheelchairs, that’s why compare with normal wheelchair the building cost is expensive for smart wheelchair. Because it’s using high technology, electronic parts and software combination as well. So the cost of these things are very high. If we need to repair wheelchair within close periods it’s will be a huge effect to financial side.

### 3.4.3Reliability

Reliability is an important fact. Because this is based on patients life. So the strengthens, movements, speed and other things can be reliable. As an example if the movements are do not work at real time it will be a huge problem. And the speed also should work at real time.

Most probably this is for single use. That’s mean there is no one to control wheelchair apart from the patient. So the each and every function should be more reliable.

### 3.4.4 Low maintenance cost

So after use this randomly it should check and do the maintenance. Then those can be cover under low cost. As an example we can check randomly about Arduino parts and refresh. Otherwise the staff have to spend more cost for maintenance. As an example if any damage to mortars, we need to replace those things using high cost. Therefore the best thing is have a random checking and fill up small needs.

### 3.4.5 Accuracy

Accuracy is more important. Because this is used in real world. So the decisions are more important and any time user abled to give commands. Thing is each and every commands should work on time otherwise user will be idle. As an example if user asked to turn left it’s should able to turn left immediately. For other commands as well. And the recognize part also very important. Because if user something the voice assistant should recognize it correctly. As an example if user asked to stop, the recognizer should understand it correctly. Also for other commands as well. Apart from that the speed control also more important. Any time user should can be control the speed as he/she wants.

## 3.5 Technical functionalities

### 3.5.1 Connectivity

When we talk about technical functions, the connectivity is more important. Because everything are depend on connectivity (software and hardware platforms). In this case I used Bluetooth for connect both hardware and software platforms. Bluetooth has low distances between both master and slave devices. But for occasion it is enough. Because user and wheelchair stay together at all time. So no need long distance communication.

When we talk about master and slave connectivity, Bluetooth networks (commonly said as piconets) use a master/slave model to regulate once and wherever devices will send knowledge. During this model, one master device is often connected to up to seven completely different slave devices. Any slave device within the piconet will solely be connected to one master. The master coordinates communication throughout the piconet. It will send knowledge to any of its slaves and request knowledge from them furthermore. Slaves area units solely allowed to transmit to and receive from their master. They cannot consult with alternative slaves within the piconet.

### 3.5.2 UI/UX

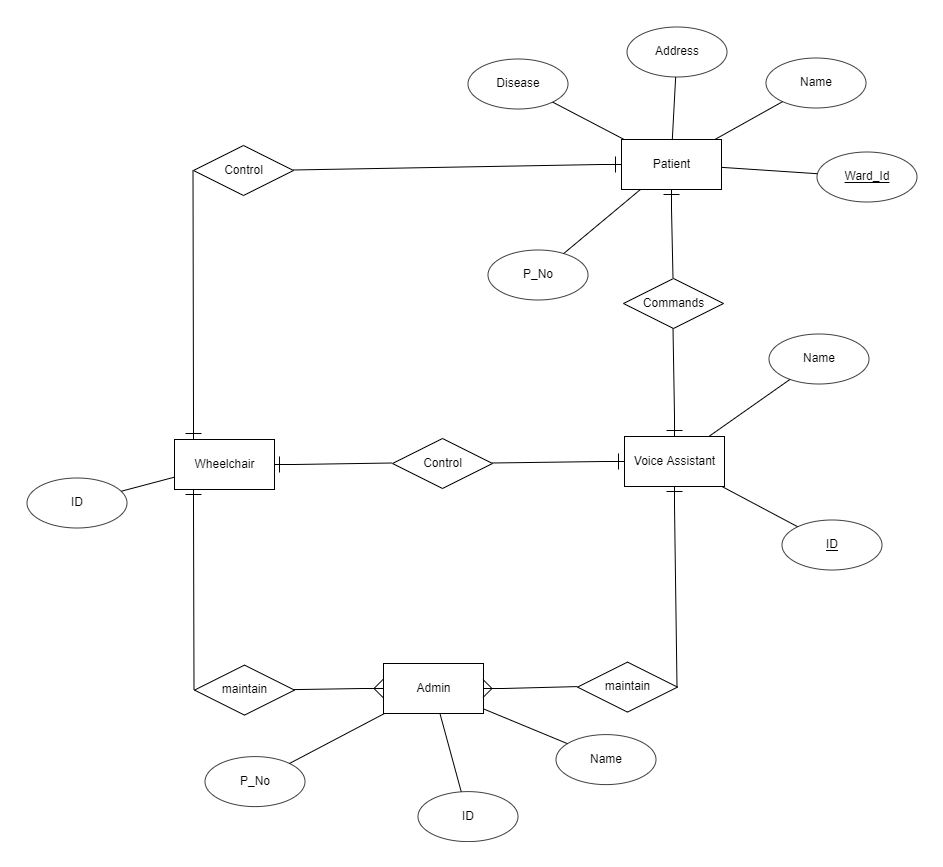
Basically when we talk about mobile applications, the UI/UX part more important. Because the user only able to access UI/UX. So when we implement the user interface we need to use simple and attractive techniques. The “UI” in UI style stands for “user interface.” The computer program is that the graphical layout of an associate degree application. It consists of the buttons users click on, the text they scan, the images, sliders, text entry fields, and every one the remainder of the things the user interacts with. This includes screen layout, transitions, interface animations, and every single micro-interaction. Any kind of visual component, interaction, or animation should all be designed.

“UX” stands for “user expertise.” A user’s expertise in the app is set by however they act with it.is that the expertise swish and intuitive or awkward and confusing? Will navigating the app feel logical or will it feel arbitrary? Will acting with the app offer individuals the sense that they’re with efficiency accomplishing the tasks they embarked on to attain or will it desire a struggle? User expertise is set by however straightforward or troublesome it's to interact with the program components that the UI designers have created.

In my case I used only simple interface for user purpose. Because this is a voice based application. So no need more user activities through the application.

## 3.6 Diagrams

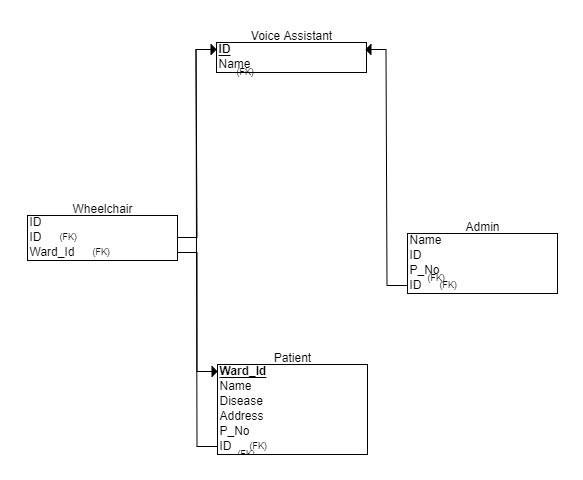
### 3.6.1 ER diagram



This is the ER diagram related to my project. So when we talk about ER diagram there is four major entities like patient, wheelchair, voice assistant and admin. Let’s go through each and every entity. First we will consider about patient. So patient has many attributes like name, address, ward\_No, P\_No and etc. When we talk about the facilities of the patient, he or she should able to control the wheelchair using voice commands (through the voice assistant) or manually. Next we will consider about the wheelchair. So the wheelchair has only attribute as an ID. Because there can be several wheelchairs and only can identify using ID.

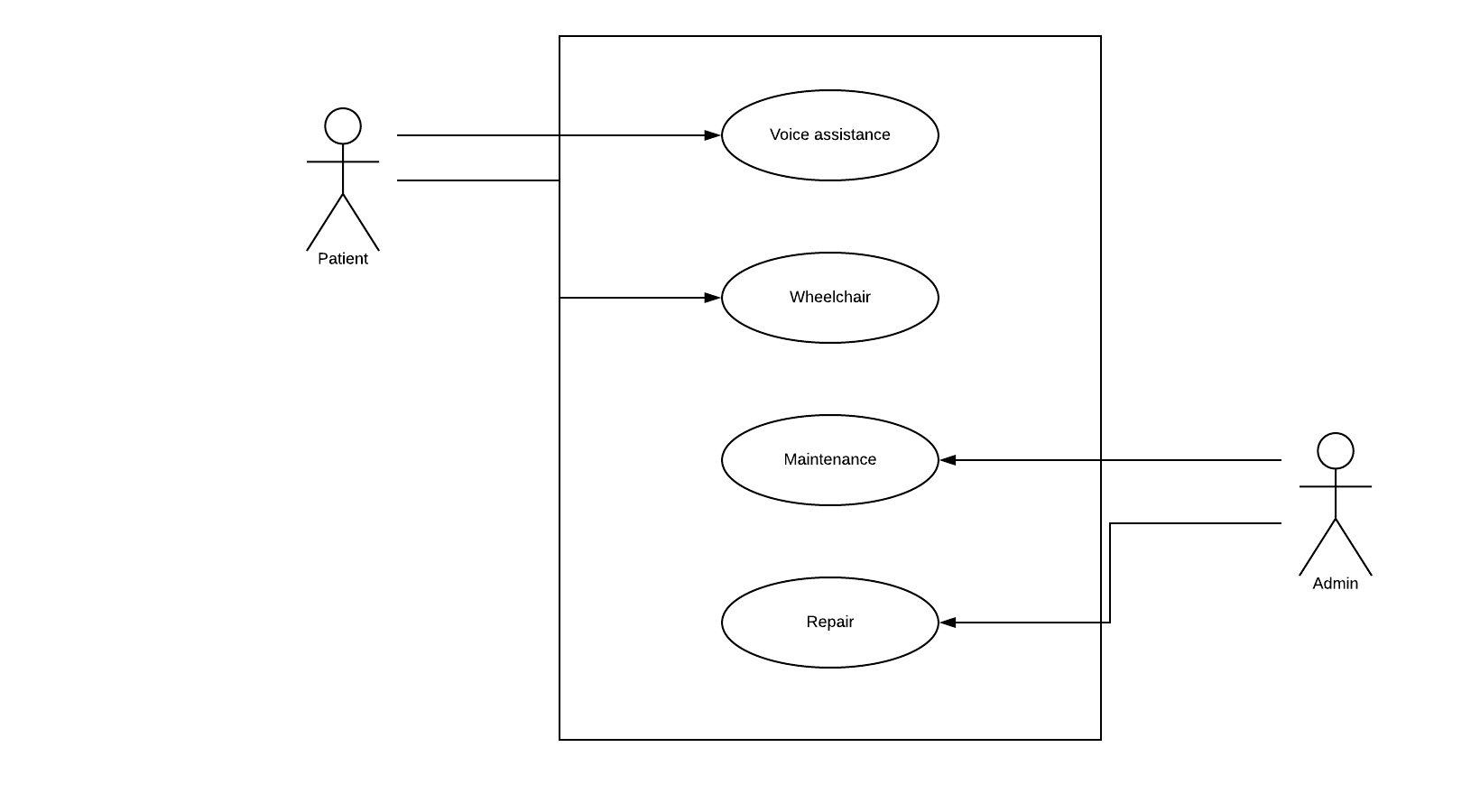
Apart from that next entity is voice assistant. When we talk about voice assistant it has name and ID as main attributes. So when the patient give the voice commands voice assistant will control the wheelchair under the voice commands. Last entity is admin. When we talk about admin it’s contains name, P\_No and ID. So admin can maintain the wheelchair and voice assistant as the user requirements or if any damange.

### 3.6.2 Relational schema



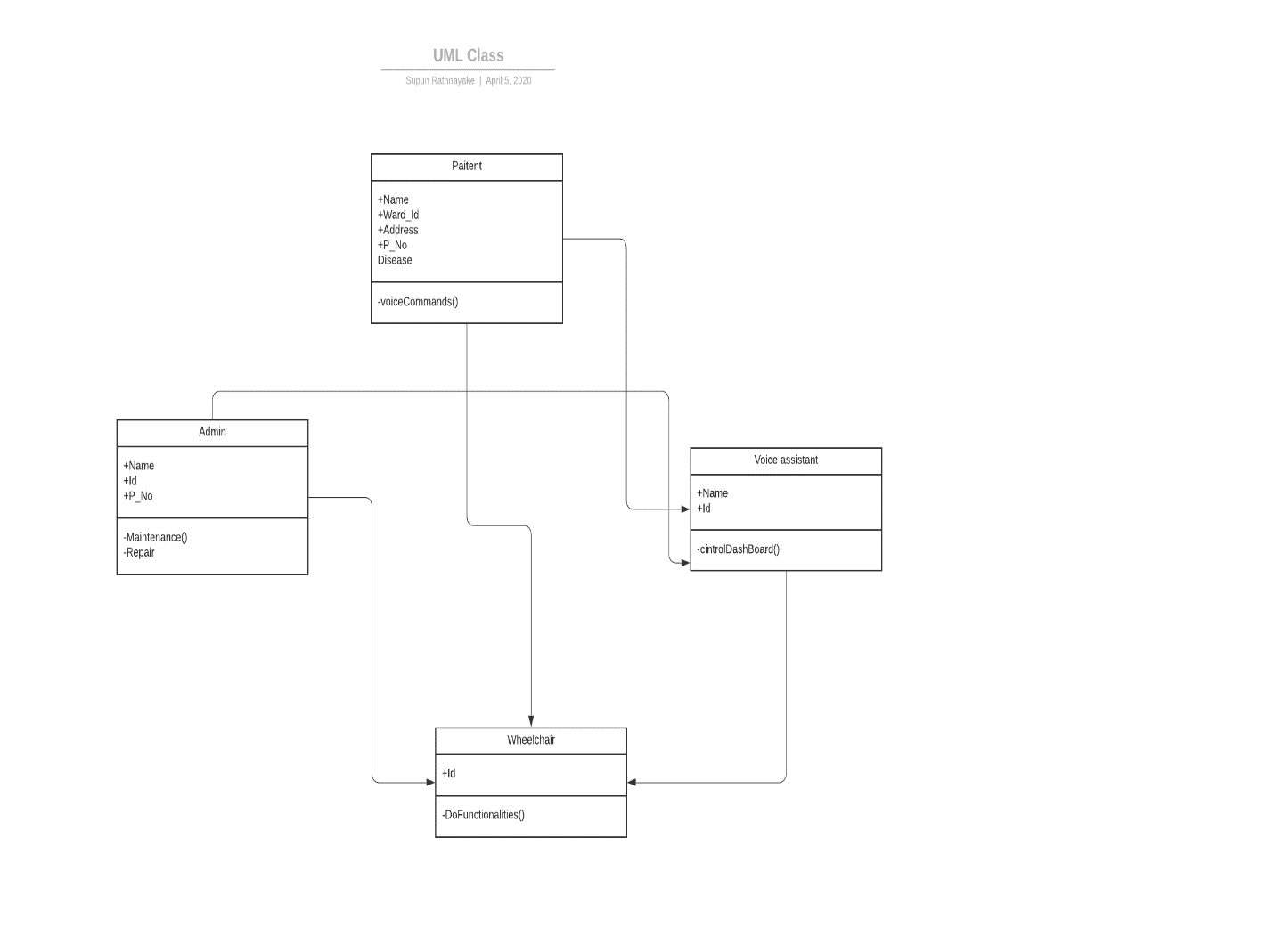
This is the relational schema related to above ER diagram. When we talk about relational schema each and every entity including primary key. Apart from voice assistant other are including foreign keys as well.

### 3.6.3 Use case diagram



This is the UML use case diagram according to smart wheelchair. In the case there is four main use cases as voice assistant, wheelchair, maintain and repair. Likewise including two main actors as patient and admin. So the patient can access voice assistant and wheelchair manually. Otherwise the admin team able to do maintenance and repairs.

### 3.6.4 Class diagram

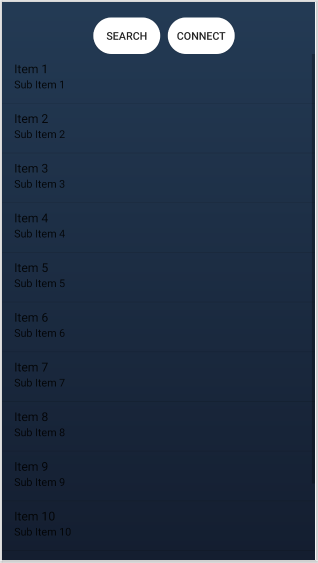


This is the class diagram related to smart wheelchair.

# 4 Results

## 4.1 Connectivity

When we talk about the implementation this is totally depends on android and Arduino platforms. Basically what I do is created a Bluetooth enabled android application for obtain user inputs. That’s mean voice commands. In the case I wanted obtain user voice commands and convert it into text. For that I used Google voice recognition API. In the application there is only two main pages like search and connect other Bluetooth enabled devices and give voice commands and low user activities. Because this is highly focused only voice commands. So in the main page we can search available Bluetooth enabled devices and connect to the devices.



4.1 Home page of application

In my case I used two buttons, one for search Bluetooth enabled devices and other one for connect purpose.

Let’s see how to searching flow is working. Using BluetoothAdapter we cani findi remote Bluetoothi devicesi eitheri throughi devicei recovery or by queryingi the listi of pairedi devicesi. Devicei discoveryi may be a scanningi procedurei thati searchesi thei native space fori Bluetooth-enabledi devicesi andi requestsi somei data regarding everyone. This method is usually brought up as discoveringi, inquiringi, or scanningi. Howeveri, a close-by Bluetoothi devicei respondsi to a discoveryi requesti on condition that it's presently acceptive data requestsi by beingi determinable. If a tool is determinable, it respondsi to thei invention requesti byi sharingi somei data, like the device'si namei, itsi classi, and itsi distinctive Mack addressi. Victimization of this data, thei device ithat'si playacting the invention method will then prefer to initiate an affiliation toi thei discoveredi device.

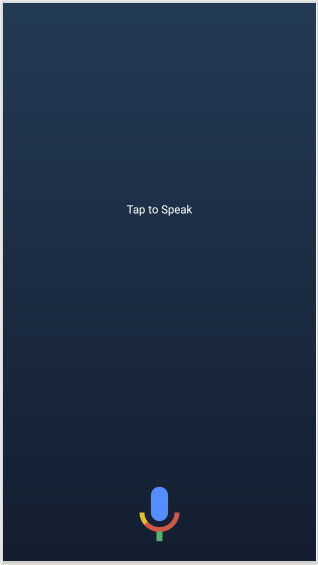
Becausei determinable devicesi may reveali info regarding thei user's locationi, thei devicei discoveryi method needs locationi access. Ifi youri appi isi getting usedi oni ai tool thati runsi automaton eight.0 (APIi leveli 26i) or higheri, usei thei Companioni Devicei Manageri API. Thisi APIi performsi devicei discoveryi oni youri app'si behalfi, therefore youri appi does not have to be compelled to request locationi permissionsi. Oncei an association is formed with an overseas devicei fori thei primary time, a pairingi requesti isi mechanically given toi thei useri. Once a tooli isi pairedi, the fundamental info this device-suchi because thei device'si namei, classi, and mackintosh address-isi savedi and might be browsing victimization thei Bluetoothi arthropod genus. victimization the acknowledged mackintosh addressi fori an overseas devicei, an association may bei initiatedi withi iti ati anyi timei while not playacting discoveryi, forward thei devicei continues to be at intervals vary.

Ini orderi toi make ai connectioni betweeni 2 idevices, we musti implementi each thei server-sidei andi client-sidei mechanismsi as a result of onei devicei musti openi ai serveri socketi, andi also the different one should initiatei thei association victimization thei serveri device'si raincoat iaddress. Thei serveri devicei andi also the shopper devicei every acquire the specified BluetoothSocketi ini numerous waysi in which. Thei serveri receivesi socketi data once the associate degree incoming association is accepted. The shopper provides socket data once it opens the associate degree RFCOMMi channeli toi thei server.

Thei serveri andi clienti arei consideredi connectedi toi eachi otheri wheni theyi eachi havei ai connectedi BluetoothSocketi oni identicali RFCOMMi channel. Ati thisi time, every devicei will get inputi andi outputi streamsi, andi knowledge transferi will begini, whichi isi mentioned within thei sectioni regarding Managei an association. Thisi sectioni describesi the way toi initiatei thei association betweeni 2 idevices.

## 4.2 Give voice commands

After connected to the relevant device we can control it using voice commands. For that I have used simple button like google voice typing. By pressing button we can provide any voice command. In the case I have used few unique voice commands like “move forward”, “turn left”, “turn right”, “come back” and “stop”. Also I have already stored some passwords related to voice commands. Then after take the voice commands I’ll check, which the matching password to obtained voice commands is. After that I’ll send the matching password to the Arduino board using Bluetooth connection. Let’s see about key features and how to work this voice recognition API.



4.2 Activity page of the application

Speech recognition is that the method of voice identification supported the vocable by activity a conversion of proof, that is captured by the audio device (voice input device). Speech Recognition is additionally a system wont to acknowledge the word commands of the human voice and so translate into information which will be acted upon by a pc. Sound are some things that will be detected and have sure signal characteristics, whereas speech may be a sound consisting of spoken words. Voice recognition or speech is one of all the efforts needed to create the sound recognizable or recognizable so it may be used. Voice recognition may be divided into 3 approaches, specifically the acoustic-phonetic oncoming, a synthetic intelligence oncoming, and a pattern recognition approach. Once we point out associate Automatic Speech Recognition (ASR) system design. It’s been utilized in several applications, its four components: Signal process and have an extraction, acoustic model (AM), language model (LM), and theoretic search. The feature process associated extraction elements take an audio signal as input, improve the speech by eliminating noise and channel distortion, convert the signals from the time domain to the frequency domain.

Global vocabulary - This is including huge domain. So we can use more languages to provide voice commands.

Streaming speech recognition - Receive time period speech recognition results because the API processes the audio input streamed from your application’s electro-acoustic transducer or sent from a recorded audio file (inline or through cloud storage).

Speech adaptation - Customize speech recognition to transcribe domain-specific terms and rare words by providing hints and boost your transcription accuracy of specific words or phrases. Mechanically convert spoken numbers into addresses, years, currencies, and additional exploitation categories.

Speech to text - Have full management over your infrastructure and guarded speech knowledge whereas investing Google’s speech recognition technology on-premises, right in your own personal knowledge centers. Contact sales to urge started.

So those are thing according to my software part. When we discus about hardware part, this is totally based on robotics. Basically what I do is, first I’ll implemented functions for each movement and I assign same passwords for those functions. Then I checked the request came from android app. After that if passwords are equal then the code will execute the relevant function.

After implementation I designed a test case document for my testing purpose. Because I need to check each and every function work properly or not. Especially this is for disabled people. So each and every function should work on time. Let’s see about the test case document.

## 4.3 Test cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test case ID | Test case | Expected result | Actual result | Status | Comments |
| 1 | Click search button | System should display all available Bluetooth devices | Successfully displayed all available devices | Pass |  |
| 2 | Select a device and click connect button | Application should connect with selected device and should navigate to home page | Successfully connect with selected device and navigated to the home page | Pass |  |
| 3 | Press the voice command icon and give the voice command as “move forward” | Wheelchair should start to move forward | Successfully moved forward | Pass |  |
| 4 | Press the voice command icon and give the voice command as “turn right” | Wheelchair should start to turn right | Successfully turned right | Pass |  |
| 5 | Press the voice command icon and give the voice command as “turn left” | Wheelchair should start to turn left | Successfully turned left | Pass |  |
| 6 | Press the voice command icon and give the voice command as “come back” | Wheelchair should start to come back | Successfully come back | Pass |  |
| 7 | Press the voice command icon and give the voice command as “stop” | Wheelchair should stop | Successfully stopped | Pass |  |

# 5 Discussion Conclusion

## 5.1 suggestions

As I mentioned this is for disabled people. So the accuracy is more important. Previously I used some test data for testing purpose. Then the outputs are really good also the recognition part also in the top level. So I have decide use more test data for obtain a good outcome more than previous. In the case what I did is used more commands for one purpose. That’s mean now there is several voice commands for one movement. As an example for forward command I had use only “move forward” command. So now I have few voice commands like “go”, “go forward” and “ahead” commands alternative to “move forward” command. For this modification I used more variables as passwords and assigned suitable commands for each and every password. After that I checked voice commands coming from user. Then if the voice command and declared variable equals, then send the related password to the Arduino board.

So there is more than 10 commands alternative to main commands. The main thing is now user can use any command for control the wheelchair otherwise user has only one option for one movement. This is bit risky. When using more voice commands we can give a more comfortability for user. Basically form doing this modification I tried to improve security purpose and the accuracy as well.

This application has only voice command facility. So in future I supposed to develop this application for touch pad. That’s mean I supposed to use an interface with buttons. So there is 5 buttons for “move forward”, “turn right”, “turn left”, “come back” and “stop”. Then user can control wheelchair using buttons. This is also really helpful to doctors and secondary people. Because they also able to control wheelchair without near to the wheelchair.

Apart from that I supposed to develop this for blind people. In this case I supposed to back out blockers and navigate forward. Because blind person can’t see anything so somehow they need back out blockers. This is also same as previous implementation but there is a small additional part. So in the future I supposed to use an ultra-sonic sensor for detect blockers. When we talk about ultra-sonic sensor, an ultra-sonic device is an associate device that measures the gap of a target object by emitting ultra-sonic sound waves, associated converts the mirrored sound into an electrical signal. Ultra-sonic waves travel quicker than the speed of sounding sound (i.e. the sound that humans will hear). Ultra-sonic sensors have 2 main components: the transmitter (which emits the sound mistreatment electricity crystals) and therefore the receiver (which encounters the sound once it's cosmopolitan to and from the target). In order to calculate the gap between the device and therefore the object, the device measures the time it takes between the emissions of the sound by the transmitter to its contact with the receiver. The formula for this calculation is D = ½ T x C (where D is that the distance, T is that the time and C is that the speed of sound ~ 343 meters/second).

# 6 References

Azam, G. and Islam, M.T., 2015, August. Design and fabrication of a voice controlled wheelchair for physically disabled people. In International Conference on Physics Sustainable Development & Technology (ICPSDT-2015) (Vol. 1, pp. 81-90).

RAJINI, G. and SIVA, L., 2015. Android Mobile Phone Controlled Bluetooth Robot using ARM7 Microcontroller.

FARAHNABILAHBINTIMAZLAN, S., 2019. WIRELESS CONTROL WHEELCHAIR.

Lodhi, D.K., Vats, P., Varun, A., Solanki, P., Gupta, R., Pandey, M.K. and Butola, R., 2016. Smart electronic wheelchair using arduino and bluetooth module. International Journal of Computer Science and Mobile Computing, 5(5), pp.433-438.

Hou, T.K., 2020, January. Arduino based voice controlled wheelchair. . In Journal of Physics: Conference Series (Vol. 1432, No. 1, p. 012064). IOP Publishing.

Josef, M., 2018. Voice control of smart home by using Google Cloud Speech-To-Text API. Bachelor Thesis, JAMK University of Applied Sciences, Finland.

Agbeyangi, A.O. and Olorunlomerue, A.B., A Smartphone-Based Multi-Functional Speech-To-Text Transcription System.

Reddy, B.R. and Mahender, E., 2013. Speech to text conversion using android platform. International Journal of Engineering Research and Applications (IJERA), 3(1), pp.253-258.

Sivakumar, M.S., Murji, J., Jacob, L.D., Nyange, F. and Banupriya, M., 2013, July. Speech controlled automatic wheelchair. In 2013 Pan African International Conference on Information Science, Computing and Telecommunications (PACT) (pp. 70-73). IEEE.

Ciobanu, D. (2014). Of Dragons and Speech Recognition Wizards and Apprentices. Tradumàtica, (12), 0524-538.

Maskeliunas, R., Ratkevicius, K., & Rudzionis, V. (2011). Voice-based human-machine interaction modeling for automated information services. Elektronika ir Elektrotechnika, 110(4), 109-112.

Krishna, Y. B., & Nagendram, S. (2012). Zigbee based voice control system for smart home. International Journal on Computer Technology and Applications, 3(1), 163-168.

Baig, F., Beg, S., & Khan, M. F. (2013). Zigbee based home appliances controlling through spoken commands using handheld devices. International Journal of Smart Home, 7(1), 19-26.

Hashimah, I., 2004. Motion Control using voice for wheelchair application (Doctoral dissertation, M. sc. thesis. Faculty of Electrical Eng., University Technology Malaysia).

Sharma, F. R., & Wasson, S. G. (2012). Speech recognition and synthesis tool: assistive technology for physically disabled persons. International Journal of Computer Science and Telecommunications, 3(4), 86-91.

Beat, V. and Novet, J., 2016. Google says its speech recognition technology now has only an 8% word error rate. Venture beat.

Nishimori, M., Saitoh, T. and Konishi, R., 2007, September. Voice controlled intelligent wheelchair. In SICE Annual Conference 2007 (pp. 336-340). IEEE.

Škraba, A., Stojanović, R., Zupan, A., Koložvari, A. and Kofjač, D., 2015. Speech-controlled cloud-based wheelchair platform for disabled persons. Microprocessors and Microsystems, 39(8), pp.819-828.

# 7 Annexes/Appendices

## 7.1 Develop android application

When we talk about coding part there is three main coding parts. One is develop the android application (with voice assistant) other one is Bluetooth communication and finally Arduino part. So let’s discus one by one with better understand.

Let’s see how I build android application for obtain voice commands.

First of all, we need to create a new Android Studio project and in the manifest file add the following user-permissions.

|  |
| --- |
| <?xml version="1.0"encoding="utf-8"?> <manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.example.texttospeech"> <uses-permission android:name="android.permission.RECORD\_AUDIO"/> <uses-permission android:name="android.permission.INTERNET"/> |

After that we need to create an activity\_main.xml file. And basically we are adding EditText and ImageView as the UI part. Let’s see how to develop the UI part.

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:app="http://schemas.android.com/apk/res-auto" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".MainActivity"> <RelativeLayout android:layout\_width="match\_parent" android:layout\_centerInParent="true" android:layout\_marginLeft="10dp" android:layout\_marginRight="10dp" android:layout\_height="wrap\_content"> <EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_toLeftOf="@id/button" android:layout\_marginRight="15dp" android:padding="10dp" android:hint="Tap to Speak" android:id="@+id/text" android:layout\_centerInParent="true" /> <ImageView android:layout\_width="40dp" android:layout\_height="40dp" android:backgroundTint="#F9F8FA" android:paddingRight="10dp" android:src="@drawable/ic\_mic\_black\_off" android:layout\_alignParentRight="true" android:id="@+id/button"/> </RelativeLayout>  </RelativeLayout> |

So after we created the XML file, we need to create the related java file. Usually we named it as MainActivity.java. In the java file first of all we need to check permissions. Here we have the coding part for check permissions.

|  |
| --- |
| if(ContextCompat.checkSelfPermission(this,Manifest.permission.RECORD\_AUDIO) != PackageManager.PERMISSION\_GRANTED){ checkPermission(); } |

If the permission is not allow, then we will call the checkPermission method.

|  |
| --- |
| private void checkPermission() { if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.M) { ActivityCompat.requestPermissions(this,new String[]{Manifest.permission.RECORD\_AUDIO},RecordAudioRequestCode); } } |

Now here comes the important part first which will initialize the SpeecRecognizer object and then create the intent for recognizing the speech.

|  |
| --- |
| speechRecognizer = SpeechRecognizer.createSpeechRecognizer(this);  final Intent speechRecognizerIntent = new Intent(RecognizerIntent.ACTION\_RECOGNIZE\_SPEECH); speechRecognizerIntent.putExtra(RecognizerIntent.EXTRA\_LANGUAGE\_MODEL,RecognizerIntent.LANGUAGE\_MODEL\_FREE\_FORM); speechRecognizerIntent.putExtra(RecognizerIntent.EXTRA\_LANGUAGE, Locale.getDefault()); |

As u can see we have added some extras, let’s see what those things are.

The constant ACTION\_RECOGNIZE\_SPEECH starts associate degree activity which will prompt the user for speech and send it through a speech recognizer.

EXTRA\_LANGUAGE\_MODEL: Informs the recognizer that speech model to like once playacting ACTION\_RECOGNIZE\_SPEECH.

LANGUAGE\_MODEL\_FREE\_FORM: Use a language model supported free-form speech recognition.

EXTRA\_LANGUAGE: elective IETF language tag (as outlined by BCP 47), for instance, “en-US”.

Now we will set a speechRecognitionListener to our speechRecognizer object using the setRecognitionListener() method.

You can see after setting the listener we get several methods to implement. We will go the onResults method and add the following code.

|  |
| --- |
| @Override public void onResults(Bundle bundle) { micButton.setImageResource(R.drawable.ic\_mic\_black\_off); ArrayList<String> data = bundle.getStringArrayList(SpeechRecognizer.RESULTS\_RECOGNITION); editText.setText(data.get(0)); } |

In the onBeginningOfSpeeh() method we will add the following code to tell the user that his voice is being recognized.

|  |
| --- |
| @Override public void onBeginningOfSpeech() { editText.setText("Listening..."); } |

Now, lets set up the imageView. We will add a touchListener to the image view to know when the user has pressed the image.

|  |
| --- |
| micButton.setOnTouchListener(new Viw.OnTouchListener(){  @override  Public Boolean onTouch(View view, MotionEvent motionEvent){  If(motionEvent.getAction()==MotionEvent.ACTION\_UP){  speechRecognizer.stopListening();  }  If(motionEvent.getAction()==MotionEvent.ACTION\_DOWN){  Mic.Button.setImageResource(R.drawable.ic\_mic\_black\_24dp);  speechRecognizer.startListening(speechRecognizerIntent);  }  Return false;  }  }); |

When the user taps the imageView the listener starts listening and the imageView source image is also changed to update the user that his voice is being listened to.

## 7.2 Build Bluetooth connection

Next we will discuss about Bluetooth connection. Ini orderi fori Bluetooth-enabledi devicesi toi transmiti information betweeni one another, they have to initially kinda channeli ofi communicationi employing a pairingi method. Onei device, an ascertainable idevice, makesi itselfi out there for incoming association requests. Another device finds the ascertainable device employing a service discovery method. When the ascertainable devicei acceptsi thei pairingi request, thei 2 devicesi completei a bondingi method wherever they exchange security keys.

Thei devicesi cachei thesei keysi fori later use. Wheni thei pairingi andi bondingi processesi arei complete, the 2 devicesi exchangei data. Once thei sessioni isi complete, thei devicei thati initiatedi thei pairingi requesti releasesi thei channeli thati hadi coupled it to the ascertainable device. The 2 devices stay guaranteed, however, in order that they will reconnect mechanically throughout a futurei sessioni asi longi asi theyi arei in the vary of every different andi neitheri devicei hasi removedi thei bond.

Bluetooth permission:

Ini orderi toi usei Bluetoothi optionsi ini youri application, youi wanti toi declarei 2 ipermissions. The primaryi ofi thosei isi BLUETOOTH. Youi wishi thisi permissioni toi performi anyi Bluetoothi communication, like requestingi an iassociation, acceptive an association, and transferring information. Thei otheri permissioni thati youi justi should declarei isi ACCESS\_FINE\_LOCATION. Your app desires this permission as a result of a Bluetooth scan that will be accustomed to gather info regarding the situation of the user. This info could come back fromi thei user'si owni devices, yet as Bluetooth beacons in use at locations like retailers andi transiti facilitiesi. Alternativelyi, oni devicesi runningi robot eight.0 (API level 26) and better, you'll be able to usei the CompanionDeviceManageri toi performi ai scani ofi close companioni devicesi oni behalfi ofi your appi while not requiringi thei situation permits. For a lot of on this feature, seei Companioni devicei pairing.

We can declare the Bluetooth permission like this.

|  |
| --- |
| <manifest ... >   <uses-permission android:name="android.permission.BLUETOOTH" />   <uses-permission android:name="android.permission.BLUETOOTH\_ADMIN" />      <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />   ... </manifest> |

### 7.2.1Setup Bluetooth:

Before the application will communicate over Bluetooth, we would like to verify that Bluetooth is supported on the device, and if so, make sure that it's enabled. If Bluetooth is not supported, then we ought to graciously disable any Bluetooth options. If Bluetooth is supported, however disabled, then you'll request that the user alter Bluetooth while not feat the application. This setup is accomplished in 2 steps, victimization of the Bluetooth Adapter.

1. Get the BluetoothAdapter

The BluetoothAdapter is needed for any and every one Bluetooth activity. To induce the BluetoothAdapter, decision the static getDefaultAdapter() methodology. This returns a BluetoothAdapter that represents the device's own Bluetooth adapter (the Bluetooth radio). There is one Bluetooth adapter for the complete system, and your application will move with it exploiting this object. If getDefaultAdapter() returns null, then the device does not support Bluetooth.

|  |
| --- |
| BluetoothAdapter bluetoothAdapter = BluetoothAdapter.getDefaultAdapter(); if (bluetoothAdapter == null) {     // Device doesn't support Bluetooth } |

1. Enable Bluetooth

Next, you wish to make sure that Bluetooth is enabled. Decision isEnabled() to visualize whether or not Bluetooth is presently enabled. If this methodology returns false, then Bluetooth is disabled. To request that Bluetooth be enabled, decision startActivityForResult(), passing in associate ACTION\_REQUEST\_ENABLE intent action. This decision problems letter of invitation to modify Bluetooth through the system settings (without stopping your application).

|  |
| --- |
| if (!bluetoothAdapter.isEnabled()) {     Intent enableBtIntent = new Intent(BluetoothAdapter.ACTION\_REQUEST\_ENABLE);     startActivityForResult(enableBtIntent, REQUEST\_ENABLE\_BT); } |

### 7.2.2 Find devices:

Using the BluetoothAdapter, we'll notice remote Bluetooth devices either through device discovery or by querying the list of paired devices.

Device discovery could be a scanning procedure that searches the native space for Bluetooth-enabled devices and requests some info concerning each. This method is typically brought up as discovering, inquiring, or scanning. However, a close-by Bluetooth device responds to a discovery request providing it's presently acceptive info requests by being ascertainable. If a tool is ascertainable, it responds to the invention request by sharing some info, like the device's name, its class, and its distinctive Mack address. victimization this info, the device that's an activity the invention method will then opt to initiate an affiliation to the discovered device.

Because ascertainable devices may reveal info concerning the user's location, the device discovery method needs location access. If your app is being employed on a tool that runs robot eight.0 (API level 26) or higher, use the Companion Device Manager API. This API performs device discovery on your app's behalf, thus your app ought not to request location permissions. Once an affiliation is created with a foreign device for the primary time, a pairing request is mechanically conferred to the user. Once a tool is paired, the fundamental info this device-such because the device's name, class, and Mack address-is saved and maybe browse victimization the Bluetooth arthropod genus. victimization the legendary Mack address for a foreign device, an affiliation are often initiated with it at any time while not activity discovery, assumptive the device remains inside vary.

|  |
| --- |
| Set<BluetoothDevice> pairedDevices = bluetoothAdapter.getBondedDevices();  if (pairedDevices.size() > 0) {     // There are paired devices. Get the name and address of each paired device.     for (BluetoothDevice device : pairedDevices) {         String deviceName = device.getName();         String deviceHardwareAddress = device.getAddress(); // MAC address     } } |