SPEECH EMOTION RECOGNITION FOR CHILD SAFETY USING AIOT

A PROJECT REPORT

Submitted by

JENNIFER K (412719205004)

VIJAYALAKSHMI V (412719205017)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

TAGORE ENGINEERING COLLEGE

RATHINAMANGALAM, CHENNAI 600 127

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

ANNA UNIVERSITY : CHENNAI 600 025

BONAFIDE CERTIFICATE

Certificate that this report **"SPEECH EMOTION RECOGNITION FOR CHILD SAFETY USING AIOT"** is the bonafide work of **"JENNIFER K (412719205004), VIJAYALAKSHMI V (412721920517)"** who carried out the project work under my supervision.

SIGNATURE
Dr. R. BEULAH JAYAKUMARI., M.E., Ph.D.,
Associate Professor
HEAD OF THE DEPARTMENT
Department of
Information Technology,
Tagore Engineering College,
Rathinamangalam,
Chennai-600 127.

SIGNATURE
Mr.S.MURUGESAN.,M.E., (Ph.D).,
Assistant Professor
SUPERVISOR
Department of
Information Technology,
Tagore Engineering College,
Rathinamangalam,
Chennai-600 127.

Submitted for the project Viva Voice examination held on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We take this prospect to extend our earnest and wholehearted thanks to our Chairperson **Dr. M. MALA, M.A., M.Phil., Ph.D.,** Tagore Engineering College, Rathinamangalam, Chennai for providing all necessary infrastructure and other facilities to complete this project successfully.

We comprehend our heartfelt thanks to **Dr. R. RAMESH, M.E., Ph.D., FIE, MIETE, MISTE.,** Principal, Tagore Engineering College for his constant support and encouragement during the tenure of this project.

We express our genuine gratitude and appreciation to our Head of the Department **Dr. R. BEULAH JAYAKUMARI, M.E., Ph.D.,** for her valuable suggestions and inspiration to complete the project successfully.

We extend our sincere gratitude to our Project Supervisor **Mr.S.MURUGESAN, M.E., (Ph.D).,** for extending all the necessary help to complete the project work successfully.

We thank our Project Coordinator **Mr. S. MURUGESAN, M.E., (Ph.D).,** Associate professor helped us to coordinate throughout the project work.

We also express our sincere thanks to **All Teaching Faculty and Non-Teaching Staff** who have rendered help during various stages of our project work.

ABSTRACT

In a Modernized Technology world, there is always an urgent need to enhance the safety of people especially little children. The prime inspiration of this paper talks about the idea of safety and tracking device for child. Thepurpose of the device is to help parents locate their children with ease. The system aims to enhance child safety by utilizing AIoT technologies in real-time speech emotion recognition with GPS and GSM to detect the child's live location and emotional state in danger situations. The system utilizes a speech emotion recognition module to analyse the child's speech in real-time and sends an alert message to the parent or guardian through the GSM module in case of a danger situation. The GPS module helps in tracking the child's live location in case of an emergency. The system offers numerous advantages, such as enhanced child safety, real-time monitoring, and minimal data storage requirements. Future enhancements to the system may include the use of advanced AI algorithms to improve emotion recognition accuracy and the integration of additional sensors to provide more comprehensive monitoring of the child's safety. Overall, the proposed system offers a promising solution to improve child safety and provides an effective example of the potential benefits of AIoT technologies.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF ABBREVIATION

ABBREVIATION	STANDS FOR
AIOT	Artificial Intelligence of Things
AI	Artificial Intelligence
IoT	Internet of Things
IDE	Integrated Development Environment
SER	Speech Emotion Recognition
GSM	Global System for Mobile Communication
USB	Universal Serial Bus
GPS	Global Positioning System
SIM	Subscriber Identity Module
SMS	Short Message Service
GUI	Graphical User Interface
GPRS	General Packet Radio Service
COM	Communication Port
GPR	Ground Penetrating Radar

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

AIoT stands for Artificial Intelligence of Things, which refers to the integration of artificial intelligence (AI) technology and the Internet of Things (IoT) devices. The combination of these two technologies can create smart systems that can learn, adapt, and make decisions based on data collected from various IoT sensors. AIoT systems can be used in various applications such as smart homes, smart cities, industrial automation, healthcare, transportation, and agriculture. For example, in a smart home, AIoT can be used to automatically adjust the temperature, lighting, and other settings based on the user's preferences and habits. In industrial automation, AIoT can be used to optimize production processes and reduce downtime by predicting equipment failures. AIoT devices typically include sensors, processors, and communication modules, which enable them to collect and process data, communicate with other devices, and make decisions based on the data. AI algorithms, such as machine learning and deep learning, are used to analyze the data and make predictions or recommendations. The benefits of AIoT include increased efficiency, improved accuracy, enhanced automation, and reduced cost. However, there are also concerns about data privacy and security, as AIoT systems collect and process large amounts of sensitive data. Therefore, it is important to implement proper security measures to protect the data and ensure the privacy of users. AIoT systems have the potential to revolutionize the way we live and work. In agriculture, AIoT can be used to monitor soil moisture levels, temperature, and other environmental factors, and use that data to optimize crop yields and reduce water usage.

In healthcare, AIoT can be used to monitor patients remotely and provide personalized treatment recommendations based on their health data. In transportation, AIoT can be used to improve traffic flow, reduce congestion, and enhance safety by predicting and preventing accidents. One of the key benefits of AIoT is increased efficiency. By using AI algorithms to analyze data collected by IoT sensors, AIoT systems can optimize processes and reduce waste. For example, in a smart factory, AIoT can monitor production processes and predict when equipment is likely to fail, allowing for preventative maintenance to be performed before any downtime occurs. This can lead to increased productivity and reduced costs. Another benefit of AIoT is improved accuracy. By using AI algorithms to analyze data, AIoT systems can make more accurate predictions and recommendations. For example, in a smart home, AIoT can learn a user's preferences and adjust settings accordingly, providing a more personalized experience. In healthcare, AIoT can analyze patient data to identify patterns and make more accurate diagnoses and treatment recommendations. AIoT also enables enhanced automation. By integrating AI and IoT technologies, processes can be automated more effectively. For example, in a smart city, AIoT can be used to control traffic lights and adjust traffic flow in real-time, reducing congestion and improving safety. In industrial automation, AIoT can be used to control machinery and optimize production processes. Finally, AIoT can reduce costs. By optimizing processes and reducing waste, AIoT can help organizations save money. For example, in agriculture, AIoT can reduce water usage and optimize crop yields, leading to cost savings. In healthcare, AIoT can reduce hospital readmissions and improve patient outcomes, leading to cost savings for healthcare providers. While AIoT has many benefits, there are also concerns about data privacy and security. AIoT systems collect and process large amounts of sensitive data, and it is important to implement proper security measures to protect that data and ensure the privacy of users. As AIoT continues to evolve, it is important to address these concerns and ensure that the benefits of AIoT are realized in a safe and responsible way.

1.2 OBJECTIVE

The objective of combining speech emotion recognition with a GPS (Global Positioning System) module could be to better understand the emotional state of individuals in specific locations. By collecting data on both the emotions expressed in speech and the geographic location of the individual, researchers or organizations could gain insights into how different environments or situations affect people's emotional states. The combination of speech emotion recognition and GPS could also be used in personalized navigation systems. By analyzing the emotional state of the user, the system could provide more appropriate or helpful directions, such as avoiding routes or locations that may cause anxiety or stress. Overall, the objective of combining speech emotion recognition with GPS is to better understand the complex interplay between emotional states and the environment, and to develop more effective strategies for improving emotional well-being and navigation. Our aim of this work is to track and secure the child anywherein the place, over a command through SMS to communicate between device and parent with the help of GSM module wired to Arduino NANO Board. It is used track a child in every situation and sends information to their parents with ease. The overall system consists many electronic components to give an alert message the parents.

CHAPTER 2
LITERATURE SURVEY

Title: Speech Emotion Recognition using Deep Neural Networks with GPS-based Location Information

Author: H. Choi and H. Lee

Year: 2021

Description: This system that works on security solution using smart devices based on IOT. In this paper the system intends to a wireless technique in the form of embedded device namely Raspberry Pi for women that will serve the purpose of alerts and way of communicating with secure channels and it captures the image using R-pi camera. There are many android applications for women safety but theyas not as much as efficient. So, to solve this issue of women safety we developed a prototype which is easy to use and which is efficient to provide help to that victim. so, when the victim presses kits button, our application will capture the photo, collect user's information to send notification to registered phone numbers withlink of captured image. This saves the time and that victim get help without loss of time. Also, in the case of Children security the system proposes a speed monitoring and location tracking facilities using GPS, GPRS, GSM. The system consists of bus unit. The bus unit which is used to detect the path of Bus by using GPS. Weather the bus is travelling on its day to day route and also it monitors the over speeding of bus. For the mechanism of vehicle tracking Haversine and Trilateration algorithm are used. According to that the by using GSM alert messages will be send to their parents and vehicle owner. The system has been developed on web-based data driven application and android application has provides the useful information.

Title: Spatial-Temporal Analysis of Emotion in Speech Using GPS Location Data"

Author: K. Lerman, J. Murdock, and A. Pavan

Year: 2021

Description: The authors collected data from social media platforms that included user-generated content in the form of text and speech, along with GPS location information. They used a machine learning approach to automatically detect emotions in speech and then analyzed the data to determine if there were any spatial or temporal patterns in the expression of emotions. The results of the study suggest that certain emotional states are more likely to be expressed in specific geographic locations. For example, the authors found that negative emotions, such as sadness and anger, were more likely to be expressed in areas with high levels of urbanization and population density, while positive emotions, such as happiness, were more likely to be expressed in areas with natural scenery and open spaces. The study also found that emotions expressed in speech were influenced by temporal factors such as time of day and day of the week. For example, positive emotions were more likely to be expressed during weekends, while negative emotions were more likely to be expressed during weekdays. Overall, this study provides insights into the relationship between emotions and geographic location, which could have implications for understanding how environmental factors affect emotional states and for designing more emotionally engaging marketing or advertising campaigns. The study also demonstrates the potential of using machine learning techniques and GPS data to analyze emotions expressed in speech.

Title: Speech Emotion Recognition with Multimodal Information Combining Audio and GPS Data

Author: J. Gu, Y. Liu, and L.

Year: 2022

Description: The authors propose a deep learning model that combines both types of data to achieve higher accuracy than models that use only one type of data. The proposed model uses a convolutional neural network (CNN) to extract features from audio data and a recurrent neural network (RNN) to process GPS location data. The features from both networks are then combined and fed into a fully connected neural network for final classification. The authors conducted experiments to evaluate the performance of the proposed model using a dataset of speech recordings with corresponding GPS location information. The experimental results show that the proposed model achieves higher accuracy compared to models that use only audio or GPS data. The study also demonstrates the potential of using multimodal data to improve speech emotion recognition accuracy. The authors suggest that the use of additional types of data, such as visual or physiological data, could further improve emotion recognition accuracy. Overall, this study highlights the importance of considering multiple sources of information in emotion recognition and shows the potential of combining audio and GPS data to improve the accuracy of speech emotion recognition. The findings of this study could have applications in areas such as virtual assistants, affective computing, and emotional marketing.

Title: Smart Security solution for women and children safety based on GPS using IOT

Author: Asmita Pawar, Pratiksha Sagare, Tejal Sasane and Kiran Shinde

Year: 2020

Description: This system that works on security solution using smart devices based on IOT. In this paper the system intends to a wireless technique in the form of embedded device namely Raspberry Pi for women that will serve the purpose of alerts and way of communicating with secure channels and it captures the image using R-pi camera. There are many android applications for women safety but theyas not as much as efficient. So, to solve this issue of women safety we developed a prototype which is easy to use and which is efficient to provide help to that victim. so, when the victim presses kits button, our application will capture the photo, collect user's information to send notification to registered phone numbers withlink of captured image. This saves the time and that victim get help without loss of time. Also, in the case of Children security the system proposes a speed monitoring and location tracking facilities using GPS, GPRS, GSM. The system consists of bus unit. The bus unit which is used to detect the path of Bus by using GPS. Weather the bus is travelling on its day to day route and also it monitors the over speeding of bus. For the mechanism of vehicle tracking Haversine and Trilateration algorithm are used. According to that the by using GSM alert messages will be send to their parents and vehicle owner. The system has been developed on web-based data driven application and android application has provides the useful information.

Title: Smart Security Solution for Women based on Internet of Things (IOT)

Author: G C Harikiran, Karthik Menasinkai, Suhas Shirol

Year: 2020

Description: Today in the current global scenario, the prime question in every girl's mind, considering the ever-rising increase of issues on women harassment in recent past is mostly about her safety and security. The only thought hauntingevery girl is when they will be able to move freely on the streets even in odd hours without worrying about their security. This paper suggests a new perspective to usetechnology for women safety. "848Indian Women Are Harassed, Raped, Killed Everyday!!" That's a way beyond HUGE number! We propose an idea which changes the way everyone thinks about women safety. A day when media broadcasts more of women's achievements rather than harassment, it's a featachieved! Since we (humans) can't respond aptly in critical situations, the need for a device which automatically senses and rescues the victim is the venture of our idea in this paper. We propose to have a device which is the integration of multipledevices, hardware comprises of a wearable "Smart band" which continuouslycommunicates with Smart phone that has access to the internet. The application is programmed and loaded with all the required data which includes Human behaviour and reactions to different situations like anger, fear and anxiety. This generates a signal which is transmitted to the smart phone. The software or application has access to GPS and Messaging services which is pre-programmed insuch a way that whenever it receives emergency signal, it can send help request along with the location co-ordinates to the nearest Police station, relatives and the people in the near radius who have application.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

In previous days, if a child had been theft or in danger it is difficult for the parents to identify the location and personal status. They need the help of the people surrounding the child and sometimes the people surrounding the child may do harm to him. The parents need to search all the places. For that need there are many wearable's in the market which help to track the daily activity of childrenand also help to find the child using Wi-Fi and Bluetooth services present on the device. The speech and emotion recognization system should recorded the audio from a conversation is first stored in cloud storage, and then processed offline using algorithms to detect emotions in the speech. The emotion detection results are then stored in the cloud along with the original audio data. However, a disadvantage of this system is that it is not designed for real-time emotion detection, and the processing time can be relatively long, depending on the amount of speech data being analyzed. Additionally, the reliance on cloud storage may raise privacy and security concerns for some users, as their speech data is stored offsite and potentially accessible to unauthorized parties.

3.1.1 DISADVANTAGES

1. **Delayed response:** The non-real-time system may not be able to respond to the situation in real-time, which may cause a delay in sending an alert or notification to the caregivers or parents.
2. **Security risks:** Storing data in the cloud can pose security risks, as the data can be vulnerable to cyber-attacks or unauthorized access.
3. **Dependence on internet connectivity:** The system depends on internet connectivity, and if the internet is down, the system may not function correctly, and the data may not be available in real-time.
4. **Cost:** The cost of cloud storage can be high, especially for large amounts of data. This may make the system more expensive to operate and maintain.
5. **Reliability:** The reliability of the system may be lower, as there may be technical issues or failures with the cloud storage provider, which can affect the accessibility and reliability of the data.

3.2 PROPOSED SYSTEM

The proposed system for real-time speech emotion recognition with GPS to send a message through GSM to detect the child's live location and emotional state in danger situations using the technology of IoT is designed to enhance child safety by leveraging AIoT technologies. The system consists of a microphone for capturing the child's speech signals, a GPS receiver for tracking the child's location in real-time, and a GSM module for sending SMS messages to a predefined number in case of emergency. The speech signals are pre-processed to remove noise and enhance features, and then fed into a deep neural network for emotion recognition. The output of the neural network is used to classify the child's emotional state into different categories such as happy, sad, angry, or neutral. The GPS receiver continuously tracks the child's location and sends real-time data to the system. The system uses this data to determine if the child has moved out of a predefined safe area or is in danger based on their location and emotional state. In case of danger, the system sends an SMS message to a predefined phone number that includes the child's live location and emotional state. The data is not stored in the cloud but is processed and analyzed in real-time using the technology of IoT. The system is integrated with other smart devices, such as smartphones and home automation systems, to provide customizable alerts to the parents or guardians. The alerts can be customized based on the child's emotional state and location, and can be sent as SMS messages, push notifications, or email notifications. Overall, this proposed system is designed to provide real-time monitoring of a child's emotional state and location, enhancing their safety and providing peace of mind to parents or guardians.

3.2.1 ADVANTAGES

1. Real-time tracking: The system can track the child's location and emotional state in real-time, providing immediate alerts and notifications to parents or guardians in case of any danger.

2. Cost-effective: By not storing data in the cloud, the system can save costs associated with cloud storage and maintenance. It can also reduce latency and improve the system's response time.

3. Increased privacy and security: By not storing data in the cloud, the system can ensure better privacy and security of the child's data. It reduces the risk of data breaches and unauthorized access to sensitive information.

4. Customizable alerts: The system can provide customizable alerts to parents or guardians based on their preferences, ensuring they are informed about their child's location and emotional state.

5. Easy integration: The system can easily integrate with other IoT devices, such as smartwatches or home automation systems, to provide a comprehensive solution for child safety and security.

CHAPTER 4
REQUIREMENT SPECIFICATIONS

4.1 INTRODUCTION

The requirements specification is a technical specification of requirements for the hardware products. It is the first step in the requirements analysis process it lists the requirements of a particular hardware system including functional, performance and safety requirements. The requirements also provide usage scenarios from a user and an operational perspective. The purpose of hardware requirements specification is to provide a detailed overview of the hardware project, its parameters and goals. This describes the project target and its user interface, hardware and software requirements.

4.2 HARDWARE AND SOFTWARE SPECIFICATION

4.2.1 HARDWARE REQUIREMENTS

- GSM module : 800A
- ARDUNIO NANO V3.0 DEVELOPMENT BOARD
- JUMPER WIRE: Female – Female / Male – Female
- GPS TRACKER
- USB MICROPHONE

4.2.2 SOFTWARE REQUIREMENTS

- Operating System: Windows
- Software: Arduino IDE, PyCharm
- Programming Language: Embedded C, Python

4.3 TECHNOLOGIES USED

4.3.1 SOFTWARE

4.3.1.1 Arduino IDE

Arduino IDE Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board. This software is used to upload a program into Arduino UNO. A program for Arduino may be written in any programming language for a compiler that produces binary machine code for the target processor. Atmel provides a development environment for their microcontrollers, AVR Studio and the newer Atmel Studio. The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages processing and wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, syntax highlighting and it provides simple one-click mechanismsto compile and upload programs to an Arduino board. It also contains messagearea, a text console, a toolbar with buttons for common functions and a hierarchyof operation menus.

A program written with the IDE for Arduino is called a sketch. Sketches are saved on the development computer as text files with the file extension.ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension.pde. The Arduino IDE supports the languages C and C++ using special rules of code structuring.

Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution. The Arduino IDE employs the program argued to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware. This will insert one or more include statements at the top of the sketch and compile the library with your sketch. Because libraries areuploaded to the board with your sketch, they increase the amount of space it takes up. If a sketch no longer needs a library, simply delete its #include statements fromthe top of your code.

4.3.1.2 PYCHARM

PyCharm is a popular integrated development environment (IDE) used for Python programming. It is developed by JetBrains and offers a range of features designed to make the development process more efficient and streamlined. PyCharm provides tools for code analysis, debugging, testing, and version control, all within a user-friendly interface. It also offers code completion, syntax highlighting, and a range of code navigation tools. PyCharm supports a variety of frameworks, including Django, Flask, and Pyramid, and allows for the creation of virtual environments for project-specific dependencies. Overall, PyCharm is a powerful tool for Python development, favored by both beginners and experienced developers alike.

Some of the key features of PyCharm are:

- Code Completion: PyCharm provides code completion feature which helps developers to write code faster and with fewer errors. It suggests code completion options based on the context and language rules.

- Code Analysis: PyCharm has a powerful code analysis engine that helps developers to detect and fix errors, identify code smells, and improve code quality.

- Debugging: PyCharm has a built-in debugger that helps developers to find and fix bugs in their code. It supports remote debugging and allows developers to debug their code on a remote machine.

- Testing: PyCharm provides support for various testing frameworks such as PyTest, Unittest, and Django testing frameworks. It allows developers to write and run tests for their code within the IDE.

- Version Control Integration: PyCharm supports version control systems like Git, Subversion, Mercurial, and Perforce. It provides an intuitive user interface to manage code changes, commits, and merges.

- User Interface Design: PyCharm has a built-in GUI designer that allows developers to create, design, and preview graphical user interfaces (GUI) for their Python applications.

- Plugins: PyCharm supports a wide range of plugins that extend its functionality. Developers can install and use various plugins to enhance their development experience.

Overall, PyCharm is a powerful and versatile IDE for Python development that offers a wide range of features to help developers create high-quality Python applications.

4.3.2 PROGRAMMING LANGUAGE

4.3.2.1 C++

The C++ language is an object-oriented programming language & is a combination of both low-level & high-level language – a Middle-Level Language. The programming language was created, designed & developed by a Danish Computer Scientist – Bjarne Stroustrup at Bell Telephone Laboratories (now known as Nokia Bell Labs) in Murray Hill, New Jersey. As he wanted a flexible & a dynamic language which was similar to C with all its features, but with additionality of active type checking, basic inheritance, default functioning argument, classes, inlining, etc. and hence C with Classes (C++) was launched. C++ was initially known as "C with classes, " and was renamed C++ in 1983. ++ is shorthand for adding one to variety in programming; therefore C++ roughly means that "one higher than C."

HISTORY OF C++

The trace of the programming language C++ can be done back to 1979 when Bjarne Stroustrup was doing some development for this thesis for PhD. One in all the words Stroustrup had the chance to figure with, was a language referred to as Simula, that because the name implies, could be a language primarily designed for simulations.

The Simula sixty-seven language – that was the variant that Stroustrup worked with is considered the primary language to support the object-oriented programming paradigm. Stroustrup found that this paradigm was helpful for package development; but, the Simula language was way too slow for practice & practical use. Shortly subsequently, he began work on "C with Classes", because what the name implies was meant to be a superset of the C language. His main dream was to get his advanced object-oriented programming, created by him, into the C language, which was during his era was still the programming language that was widely respected for its

movability, portability & compactness, by not sacrificing the speed or the low-level practicality. His programming language enclosed inlining, basic inheritance, default function arguments, categories, and reliable sort was checking additionally to all or any the options of the C language.

The first C with categories compiler was referred to as Cfront, that got derived from a C compiler referred to as CPre. It had been a program designed to translate C with categories code to universal C. A rather attention-grabbing purpose value noting is that Cfront was written mostly in C with classes, creating it a self-hosting compiler (a compiler which will compile itself). Cfront would later be abandoned in 1993 when it became troublesome to integrate new options into it, mainly C++ exceptions. Even so, Cfront created a significant impact on the implementations of future compilers and also the operating system UNIX.

In 1983, the name of the language got modified from C with categories to C++. The ++ operator within the C language is the associate operator for incrementing a variable, which provides some insight into however Stroustrup regarded the programming language. Several new options got additional around this point, the foremost notable of that area unit virtual functions, perform overloading, references with the const keyword, and single-line comments by using the two forward slashes (which may be a feature taken from the language BCPL).

In 1985, Stroustrup's relation to the language entitled The C++ programming language was printed and got released. That very same year, C++ got enforced as an advertisement product and hence starting of it as a commercial element. The programming language wasn't formally standardized, nevertheless creating the book an essential reference. The programming language was updated once more in 1989 to incorporate protected and static members, still as an associate inheritance from many categories & classes.

In 1990, The Annotated C++ manual, which was a reference got released all over. In 1990 only, in the same & identical year, Borland's Turbo C++ compiler would also be commercially discharged as an advertisement product. Turbo C++ additional an excessiveness of other libraries which might have a substantial impact on C++'s development. Though Turbo C++'s last stable unharness was in 2006, the compiler continues to get widely used.

In 1998, the C++ customary advisory & standards committee printed the first international standard for C++ ISO/IEC 14882:1998, which might be informally called C++98. The Annotated C++ manual was aforementioned to be a significant influence within the development of the quality. The quality template library (also known as the Standard Template Library) that began its intellectual construction in 1979 got additionally enclosed in it. In 2003, the committee well-versed multiple issues that were according with their 1998 customary and revised it consequently. The modified language got dubbed as C++03.

In 2005, the same committee of C++ discharged a technical report (called as TR1) particularization varied options they were aiming to boost the newest C++ standard. The new rule was informally dubbed C++0x because it was expected to be discharged someday before the tip of the first decade. Ironically, however, the new customary wasn't fulfilled until mid-2011.

Many technical reports were discharged up till then, and a few compilers began adding experimental support for the new options and features. In mid-2011, the new C++ customary (dubbed C++11) got finished.
The Boost library project created a substantial impact on the new rule, and a few of the new modules were derived directly from the corresponding Boost libraries.

A number of the new options that got included were:

- New for loop syntax providing practicality just like foreach loops in specific different languages
- Customary threading library (which up till 2011 each C and C++ were lacking)
- Variadic templates
- Automotive vehicle (AUTO) keyword
- New instrumentation categories & classes
- New C++ time library, atomics support
- Comprehensive organization library
- Regular expression support
- Higher support for unions and array-initialization lists.

C++20 is the latest version of C++. A newer & advanced version of C++ is being released, i.e. C++23. C++ is still the third most popular programming language according to the TIOBE Index's latest study of 2019, behind Java & C, by knocking out Python. All credits for this go to the newly released C++11 version, which according to the users made it much robust, safer, easier simpler, and more expressive. Bjarne Stroustrup has created excellent work with C++. C could be a low-level programming language, and so, it does not have any classes. It does not contain several options which will create programming more well-off, however, is that the quickest language (assembly is more rapid; however, programming in construction isn't one thing you'd consider), what Bjarne Stroustrup did he additional the thing orientated half, by treating code like life objects.

What makes C++ therefore lovely is that it's the speed of C and it's additionally a high-level programming language, therefore allow us to say the most effective of each world. However on the opposite hand, C++ is challenging to be noted to newbies, and for an equivalent code that you wrote on C++, you'll be able to write in Python for the quarter of the time.

FEATURES OF C++:

The five best features of C++ are

- It is the most primarily used language in competitive programming as far as facts go. Most of the extremely rated coders typically use C++ for cryptography. You'll see it on any online websites.
- STL (standard guide library): Its an extremely time-saver for people who recognize C and still writing code for bubble type.
- Operator overloading
- Multiple inheritances. Voluminous languages don't have this facility.
- Ability to modularize code, encapsulation, and polymorphism after all.

4.3.2.2 PYTHON

Python is a high-level programming language that was first released in 1991 by Guido van Rossum. It is an open-source and interpreted language, which means that the code written in Python is not compiled but interpreted line-by-line at runtime. Python is widely used for various applications such as web development, scientific computing, data analysis, artificial intelligence, machine learning, and more. Python has gained popularity over the years due to its simplicity, readability, and flexibility. It has a simple and elegant syntax, which makes it easy to learn and write code quickly. Python has a vast standard library that provides a range of modules and functions to simplify programming tasks. Additionally, Python has a large and active community that contributes to its development and maintenance Python is an object-oriented programming language that supports multiple programming paradigms such as functional and procedural programming. It has dynamic typing, which means that variable types are determined at runtime. Python also supports automatic memory management, which simplifies memory allocation and deallocation for the programmer. Python is cross-platform, which means that it can run on various operating systems such as Windows, macOS, Linux, and more. It also has excellent integration capabilities with other languages such as C/C++, Java, and more. In summary, Python is a high-level programming language that is widely used for various applications due to its simplicity, readability, flexibility, and extensive library support.

4.3.3 HARDWARE

4.3.3.1 ARDUINO NANO

NANO Version 3 is the open source smallest Embedded Development board based on Atmega328 SMD Package Microcontroller. It is a Surface mount Breadboard Friendly board integrated with Mini USB Port. DC Power Jack is not available on this Board, so power can be given through Mini USB Cable. It automatically sense and switch to the higher potential source of power, there is no need for the power select jumper.

Specifications

Microcontroller	Atmel ATmega328 SMD Package
Operating Voltage (logic level)	5 V
Input Voltage (recommended)	7-12 V
Input Voltage (limits)	6-20 V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	40 m A
Flash Memory	32 KB (of which 2KB used by boot loader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz
Dimensions	0.70” x 1.70”

Power

The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply(pin 27). The power source is automatically selected to the highest voltage source.

Memory

The ATmega328 has 32 KB, (also with 2 KB used for the boot loader. The ATmega328 has 2 KB of SRAM and 1 KB of EEPROM.

Input and Output

Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digital Write(), and digital Read() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-upresistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. Seethe attach Interrupt() function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write() function.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pinis HIGH value, the LED is on, when the pin is LOW, it's off.

- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, isnot currently included in the Arduino language.

The Nano has 8 analog inputs, each of which provide 10 bits of resolution (i.e.1024 different values). By default they measure from ground to 5 volts, though is itpossible to change the upper end of their range using the analog Reference() function. Analog pins 6 and 7 cannot be used as digital pins. Additionally, some pins have specialized functionality:

- I2C: A4 (SDA) and A5 (SCL). Support I2C (TWI) communication using theWire library (documentation on the Wiring website).

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. Bring this line LOW to reset the microcontroller. Typically used toadd a reset button to shields which block the one on the board.

Communication

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB andthe FTDI drivers (included with the Arduino software) provide a virtual com portto software on the computer. The Arduino software includes a serial monitor whichallows simple textual data to be sent to and from the Arduino board.
The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial

communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Nano's digital pins. The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus. To use the SPI communication, please see ATmega328 datasheet.

Programming

The Arduino Nano can be programmed with the Arduino software (download). Select "Arduino Duemilanove or Nano w/ ATmega328" from the Tools > Board menu (according to the microcontroller on your board). The ATmega328 on the Arduino Nano comes preburned with a boot loader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. You can also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Nano is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the FT232RL is connected to the reset line of the ATmega328 via a 100 nano farad capacitor. When this line is asserted (taken low), the reset line drops long enoughto reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This meansthat the boot loader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Nano is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the boot loader is running on the Nano.

While it is programmed to ignore malformed data (i.e. anything besides an upload of new code) it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration orother data when it first starts, make sure that the software with which it communicates waits second after opening the connection and before sending this data.

Figure 4.1 Arduino Nano V3

4.3.3.2 GSM MODULE

The SIM800A Quad-Band GSM/GPRS Module with RS232 Interface is a complete Quad-band GSM/GPRS solution in an LGA(Land grid array) type whichcan be embedded in the customer applications. SIM800A support Quad-band 850/900/1800/1900 MHz, it can transmit Voice, SMS, and data information withlow power consumption.

With tiny size of 100 x 53 x 15 mm, it can fit into slim and compact demands of custom design. Featuring and Embedded AT, it allows total cost savings and fast time-to-market for customer applications.

The SIM800A modem has a SIM800A GSM chip and RS232 interface while enables easy connection with the computer or laptop using the USB to the Serial connector or to the microcontroller using the RS232 to TTL converter. Once you connect the SIM800A modem using the USB to RS232 connector, you need to findthe correct COM port from the Device Manager of the USB to Serial Adapter.

Then you can open Putty or any other terminal software and open a connection to that COM port at 9600 baud rate, which is the default baud rate of this modem. Once a serial connection is open through the computer or your microcontroller youcan start sending the AT commands. When you send AT commands for example "ATr" you should receive back a reply from the SIM800A modem saying "OK" orother response depending on the command sent.

Important Note:

The SIM800A requires huge power, so use a 12V DC 2 to 3 Amps adapter, any wrong adapter might cause damage also to the modem.

On the board there are Tx(D) and Rx(D) pins, near the 5V and GND pins, don't ever connect anything to them or send any command to them, they are not the TTLpins for serial communication, they are for upgrading the firmware, and if you connect them to the microcontroller or computer, the firmware will get damaged and the module will stop working.

Please read carefully the marking on the SIM card tray on how to open or close, those are made of brittle plastics and will break if you put too much pressure on them, so be very gentle and careful while removing or inserting SIM Cards.

Don't tight the antenna too much, the screw threads might get damaged by excessive or over tightening of antenna.

Please don't touch the connections or components and always put the modem on a non conductive base.

Figure 4.2 Gsm Module

FEATURE

1. Quad-band 850/900/1800/1900MHz.
2. GPRS class 2/10.
3. Control via AT commands (3GPP TS 27.007, 27.005 and SIMCOM enhancedAT command set).
4. High-Quality Product (Not hobby grade).
5. 5V interface for direct communication with MCU kit.
6. Configurable baud rate.
7. Built-in SIM Cardholder.
8. Built-in Network Status LED.
9. Inbuilt Powerful TCP/IP protocol stack for internet data transfer over G
10. Low power.

4.3.3.3 USB MICROPHONE

A USB microphone is a type of microphone that connects to a computer or other device via a USB port. Unlike traditional analog microphones, which require a separate audio interface or mixer to connect to a computer, a USB microphone has an integrated audio interface that converts the analog audio signal into a digital signal and transmits it directly through the USB connection.

Figure 4.3 USB Microphone

USB microphones are popular among podcasters, gamers, and other content creators who need a high-quality, plug-and-play microphone that is easy to use and requires minimal setup. They are also commonly used for video conferencing and other communication applications that require clear, high-fidelity audio. One advantage of USB microphones is that they are usually very portable and can be easily used with laptops or other mobile devices. They also typically have built-in headphone jacks, which allows users to monitor their audio output in real time. Overall, a USB microphone is a convenient and easy-to-use option for anyone who needs a high-quality microphone for recording audio or communicating online.

4.3.3.4 GPS TRACKER

A GPS tracker, also known as a GPS device or GPS locator, is a device that uses the Global Positioning System (GPS) to determine and track its precise location. GPS trackers are commonly used for personal and vehicle tracking, asset tracking, and even pet tracking. There are different types of GPS trackers, including standalone trackers and those that are built into devices such as smartphones. A GPS tracker typically consists of a GPS receiver that receives signals from GPS satellites and a cellular modem that transmits location data to a server or a user's device. These devices are designed specifically for tracking vehicles and are often used by businesses with fleets of vehicles to monitor their locations and usage. They can also be used by individuals to track their own vehicles or to monitor the driving habits of other family members. A GPS tracker for vehicles typically includes a GPS receiver, a cellular modem, and a SIM card for cellular data connectivity.

Figure 4.4 GPS Tracker

The location data is transmitted to a server, which can be accessed by the user through a web portal or a mobile app. GPR trackers can provide real-time location data, as well as historical location data that can be used to track the vehicle's movements over time. They can also provide alerts for events such as speeding, unauthorized use, and tampering. Overall, GPS trackers, including GPR trackers, can be useful tools for monitoring the location and usage of vehicles and other assets.

CHAPTER 5

SYSTEM ARCHITECTURE

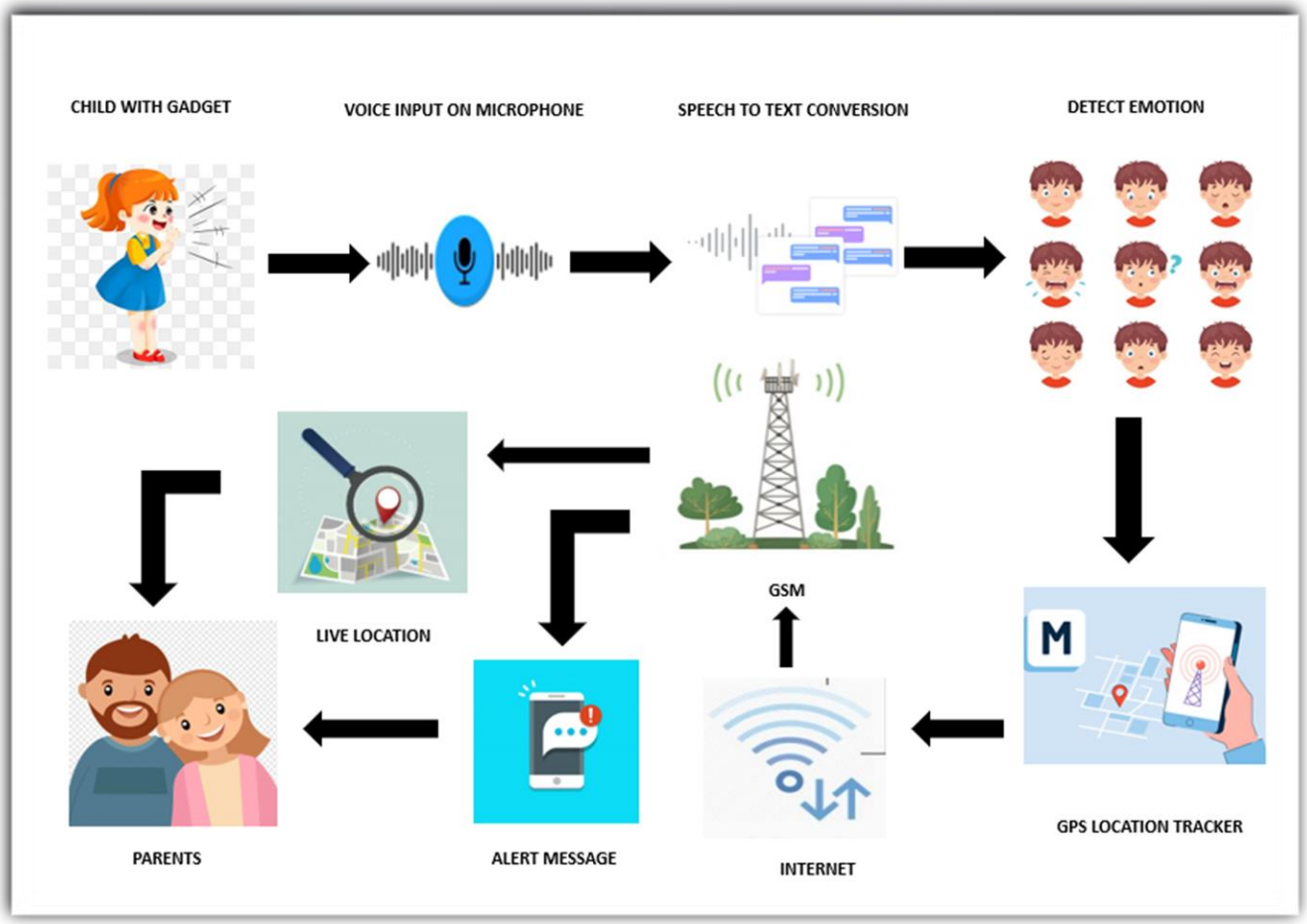


Figure 5.1 System Architecture

In this architecture, the USB Microphone is used to capture speech data, which is then pre-processed and features are extracted to train a Speech Emotion Recognition model. The model is then used to classify the speech into various emotions in real-time.

The Arduino Nano acts as the central processing unit, which receives the processed speech data, performs emotion recognition using the trained model, and sends the message using the GSM Module and GPS Tracker with live location tracking.

The GSM Module is used to establish a connection with the mobile network, which enables sending messages to the intended recipient. The GPS Tracker is used to provide location information, which is sent along with the message to provide additional context.

The Live Location Tracking feature tracks the real-time location of the device and sends it along with the message. This allows the recipient to know where the message was sent from.

Finally, the message is sent over the mobile network and received on the recipient's mobile device.

CHAPTER 6

SYSTEM DESIGN

6.1 MODULES

- Speech Emotion Recognition
- GPS for Live Location Tracking
- GSM for Sending Messages.

6.2 MODULES EXPLANATION

6.2.1 Speech Emotion Recognition

This module processes the speech signals and extracts emotional features using machine learning techniques to recognize the emotion in real-time. A Speech Emotion Recognition (SER) module is a software tool that uses machine learning algorithms to automatically recognize and classify emotions from speech signals. It can analyze various acoustic features of speech such as pitch, energy, spectral centroid, and spectral bandwidth to extract emotional characteristics and classify them into different emotion categories like happiness, sadness, anger, fear, and surprise. SER modules have numerous applications such as in developing emotional intelligence in chatbots, speech-enabled personal assistants, customer service systems, and mental health diagnosis systems. It can also be used in enhancing human-computer interaction by enabling computers to understand and respond appropriately to human emotions.

6.2.2 GPS for Live Location Tracking

A GPS module is a device that receives signals from Global Positioning System (GPS) satellites and provides information about the location, speed, and direction of an object.

It uses a GPS receiver chip to receive signals from the satellites and calculate the object's position using trilateration. The module typically consists of an antenna, a receiver chip, and a microcontroller that processes the data and outputs it in a usable format. GPS modules can be used in a wide range of applications, including navigation, tracking, and geolocation. They are commonly found in smartphones, cars, and other devices that require location-based services. Some GPS modules also include additional features such as data logging, real-time clock, and wireless connectivity.

6.2.3 GSM for Sending Messages

This module sends an emergency message to the parent's smartphone via GSM network in case of danger situations. A GSM module is a device that enables mobile communication using the Global System for Mobile Communications (GSM) standard. It allows the transmission of voice, data, and short messages between devices over cellular networks. GSM modules are widely used in applications that require wireless communication, such as tracking systems, security systems, remote monitoring, and automation systems. They typically interface with a microcontroller or a computer through serial communication protocols, such as RS-232 or USB. GSM modules come in various form factors, such as plug-in cards, surface-mount modules, or integrated modules. They support different frequency bands and network technologies, depending on the region and the service provider. GSM modules are a key component of many IoT devices that require wireless connectivity and remote control. They enable devices to communicate with each other and with the internet, and to exchange data and commands in real-time. These modules work together to provide a comprehensive child safety solution that uses real-time speech emotion recognition and GPS tracking to detect danger situations and send emergency alerts to parents.

CHAPTER 7
CONCLUSION

7.1 CONCLUSION

In conclusion, the proposed system for real-time speech emotion recognition with GPS to send a message through GSM to detect the child's live location and emotional state in danger situations using the technology of IoT is a promising solution for improving child safety. This system integrates various technologies, including speech emotion recognition, GPS, GSM, and IoT, to provide accurate and real-time information about a child's emotional state and location. With this system, parents or caregivers can respond quickly to any potential danger or emergency, reducing the risk of harm to the child. Overall, this system has the potential to make a significant positive impact on child safety and well-being.

7.2 FUTURE ENHANCEMENT

For further development can improve a device by adding features like Integration with wearable devices: The system could be integrated with wearable devices, such as smartwatches or fitness trackers, to provide continuous monitoring of the child's location and emotional state. Multi-language support: The system could be expanded to support multiple languages, which could increase its usability in diverse communities. Advanced machine learning algorithms: The system could be improved by incorporating more advanced machine learning algorithms for speech emotion recognition, such as deep learning, which could improve accuracy and reliability. Enhanced privacy features: The system could be designed with enhanced privacy features, such as end-to-end encryption, to protect the child's personal data and prevent unauthorized access.

APPENDIX

APPENDIX 1: SOURCE CODE

ARDUINO IDE – C++ PROGRAM CODE

```
#include <SoftwareSerial.h>
#include <TinyGPS++.h>
static const int RXPin = 4, TXPin = 5;
SoftwareSerial mySerial(2,3);
SoftwareSerial ss(RXPin, TXPin);
TinyGPSPlus gps;
void setup()
{
  ss.begin(9600);
  mySerial.begin(115200);   // Setting the baud rate of GSM Module
  Serial.begin(9600);
  pinMode(13,OUTPUT);
}
void loop()
{
  while(Serial.available())
  {
    char j=Serial.read();
```

```
    if(j=='1')
    {
       Serial.println("SHE NEEDS HELP");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SHE NEEDS HELP
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
           }
    }
    if(j=='2')
    {
       Serial.println("SHE IS IN DANGER");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
```

```
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SHE IS IN DANGER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='3')
    {
       Serial.println("SOMEONE HITS HER");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SOMEONE HITS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='4')
    {
```

```
      Serial.println("SOMEONE PULLS HER");
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SOMEONE PULLS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='5')
    {
      Serial.println("SOMEONE HOLDS HER");
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
```

```
            SendMessage("SOMEONE HOLDS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='6')
    {
       Serial.println("SOMEONE PUSHES HER");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SOMEONE PUSHES HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
      if(j=='7')
    {
       Serial.println("SOMEONE PINCHES HER");
```

```
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SOMEONE PINCHES HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
   }
    if(j=='8')
   {
      Serial.println("SHE IS VERY AFRAID");
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
```

```
            SendMessage("SHE IS VERY AFRAID
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='9')
    {
       Serial.println("SOMEONE KICKS HER");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SOMEONE KICKS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='10')
    {
       Serial.println("SOMEONE HOLDS HER HANDS");
```

```
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SOMEONE HOLDS HER HANDS
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='11')
    {
       Serial.println("SOMEONE GETS CLOSER TO HER");
       while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
```

```
            SendMessage("SOMEONE GETS CLOSER TO HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='12')
    {
       Serial.println("SOMEONE MISBEHAVES TO HER");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SOMEONE MISBEHAVES TO HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='13')
    {
       Serial.println("SOMEONE ATTACKS HER");
```

```
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SOMEONE ATTACKS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='14')
    {
       Serial.println("SHE IS IN DANGER");
       while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
```

```
            SendMessage("SHE IS IN DANGER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='15')
    {
       Serial.println("SHE IS VERY ANGRY");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SHE IS VERY ANGRY
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='16')
    {
       Serial.println("SOMEONE HURTS HER");
```

```
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SOMEONE HURTS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='17')
    {
       Serial.println("SHE IS VERY SAD");
       while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
```

```
            SendMessage("SHE IS VERY SAD
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='18')
    {
       Serial.println("SHE FEELS VERY TIRED");
       while (ss.available() > 0)
         if (gps.encode(ss.read()))
           displayInfo();
           if (millis() > 5000 && gps.charsProcessed() < 10)
           {
             Serial.println(F("No GPS detected: check wiring."));
             SendMessage("SHE FEELS VERY TIRED
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
    if(j=='19')
    {
       Serial.println("SHE FEELS VERY DIZZY");
```

```
      while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
            SendMessage("SHE FEELS VERY DIZZY
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
     if(j=='20')
    {
       Serial.println("SOMEONE THREATENS HER");
       while (ss.available() > 0)
        if (gps.encode(ss.read()))
          displayInfo();
          if (millis() > 5000 && gps.charsProcessed() < 10)
          {
            Serial.println(F("No GPS detected: check wiring."));
```

```
            SendMessage("SOMEONE THREATENS HER
https://www.google.com/maps/search/?api=1&query=12.862242199988533,80
.13094248192812");
          }
    }
 }
}
void displayInfo()
{
  Serial.print(F("Location: "));
  if (gps.location.isValid())
  {

Serial.print("https://www.google.com/maps/search/?api=1&query="+String(gp
s.location.lat(), 6)+","+String(gps.location.lng(), 6));
    Serial.print("\n");

SendMessage("https://www.google.com/maps/search/?api=1&query="+String(
gps.location.lat(), 6)+","+String(gps.location.lng(), 6));
    //Serial.print(gps.location.lat(), 6);
    //Serial.print(F(","));
    //Serial.print(gps.location.lng(), 6);
```

```
  }
  else
  {
    Serial.print(F("INVALID"));
  }
}
void SendMessage(String message)
{
  mySerial.println("AT+CMGF=1");
  delay(1000);
  mySerial.println("AT+CMGS=\"+919087829001\"\r");
  delay(1000);
  mySerial.println(message);
  delay(100);
  mySerial.println((char)26);
  delay(1000);
}}
```

PYCHARM – PYTHON PROGRAM CODE

```
import speech_recognition as sr
import pyttsx3


# Initialize the recognizer
r = sr.Recognizer()
import serial
import time
arduino = serial.Serial(port='COM3', baudrate=9600, timeout=.1)
def write_read(x):
    arduino.write(bytes(x, 'utf-8'))


# Function to convert text to
# speech
def SpeakText(command):
    # Initialize the engine
    engine = pyttsx3.init()
    engine.say(command)
    engine.runAndWait()
   # Loop infinitely for user to
  # speak
while (1):
```

```
# Exception handling to handle
# exceptions at the runtime

    try:
 # use the microphone as source for input.
  with sr.Microphone() as source2:

            # wait for a second to let the recognizer
            # adjust the energy threshold based on
            # the surrounding noise level
            r.adjust_for_ambient_noise(source2, duration=0.2)

            # listens for the user's input
            audio2 = r.listen(source2)

            # Using google to recognize audio
            MyText = r.recognize_google(audio2)
            MyText = MyText.lower()
            print("you say ", MyText)
            # SpeakText(MyText)
            if (MyText == "help me"):
               print("SHE NEEDS HELP")
```

```
    SpeakText("SHE NEEDS HELP")
    arduino.write(bytes('1', 'utf-8'))
if (MyText == "i am in danger"):
    print("SHE IS IN DANGER")
    SpeakText("SHE IS IN DANGER")
    arduino.write(bytes('2', 'utf-8'))
if (MyText == "don't hit me"):
    print("SOMEONE HITS HER")
    SpeakText("SOMEONE HITS HER")
    arduino.write(bytes('3', 'utf-8'))
if (MyText == "don't pull me"):
    print("SOMEONE PULLS HER")
    SpeakText("SOMEONE PULLS HER")
    arduino.write(bytes('4', 'utf-8'))
if (MyText == "leave me"):
    print("SOMEONE HOLDS HER")
    SpeakText("SOMEONE HOLDS HER")
    arduino.write(bytes('5', 'utf-8'))
if (MyText == "don't push me"):
    print("SOMEONE PUSHES HER")
    SpeakText("SOMEONE PUSHES HER")
    arduino.write(bytes('6', 'utf-8'))
```

```
if (MyText == "don't pinch me"):
    print("SOMEONE PINCHES HER")
    SpeakText("SOMEONE PINCHES HER")
    arduino.write(bytes('7', 'utf-8'))
if (MyText == "i am very afraid"):
    print("SHE IS VERY AFRAID")
    SpeakText("SHE IS VERY AFRAID")
    arduino.write(bytes('8', 'utf-8'))
if (MyText == "don't kick me"):
    print("SOMEONE KICKS HER")
    SpeakText("SOMEONE KICKS HER")
    arduino.write(bytes('9', 'utf-8'))
if (MyText == "please leave my hands"):
    print("SOMEONE HOLDS HER HANDS")
    SpeakText("SOMEONE HOLDS HER HANDS")
    arduino.write(bytes('10', 'utf-8'))
if (MyText == "don't get closer to me"):
    print("SOMEONE GETS CLOSER TO HER")
    SpeakText("SOMEONE GETS CLOSER TO HER")
    arduino.write(bytes('11', 'utf-8'))
if (MyText == "behave yourself"):
    print("SOMEONE MISBEHAVES TO HER")
```

```
    SpeakText("SOMEONE MISBEHAVES TO HER")
    arduino.write(bytes('12', 'utf-8'))
if (MyText == "don't attack me"):
    print("SOMEONE ATTACKS HER")
    SpeakText("SOMEONE ATTACKS HER")
    arduino.write(bytes('13', 'utf-8'))
if (MyText == "don't kill me"):
    print("SHE IS IN DANGER")
    SpeakText("SHE IS IN DANGER")
    arduino.write(bytes('14', 'utf-8'))
if (MyText == "get lost from here"):
    print("SHE IS VERY ANGRY")
    SpeakText("SHE IS VERY ANGRY")
    arduino.write(bytes('15', 'utf-8'))
if (MyText == "dont hurt me"):
    print("SOMEONE HURTS HER")
    SpeakText("SOMEONE HURTS HER")
    arduino.write(bytes('16', 'utf-8'))
if (MyText == "i am very sad"):
    print("SHE IS VERY SAD")
    SpeakText("SHE IS VERY SAD")
    arduino.write(bytes('17', 'utf-8'))
```

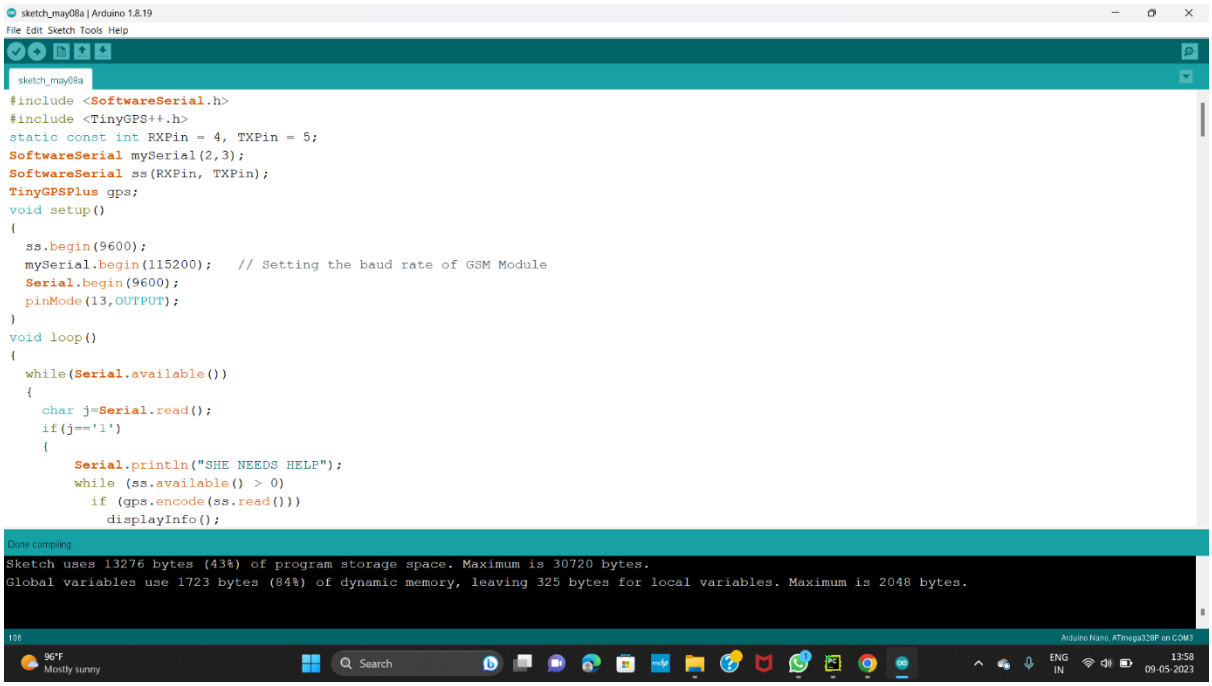
```
if (MyText == "i feel very tired"):
    print("SHE FEELS VERY TIRED")
    SpeakText("SHE FEELS VERY TIRED")
    arduino.write(bytes('18', 'utf-8'))
if (MyText == "i feel very dizzy"):
    print("SHE FEELS VERY DIZZY")
    SpeakText("SHE FEELS VERY DIZZY")
    arduino.write(bytes('19', 'utf-8'))
if (MyText == "don't threaten me"):
    print("SOMEONE THREATENS HER")
    SpeakText("SOMEONE THREATENS HER")
    arduino.write(bytes('20', 'utf-8'))

    except sr.RequestError as e:
    print("Could not request results; {0}".format(e))
    except sr.UnknownValueError:
   a = 0;
```

APPENDIX-2

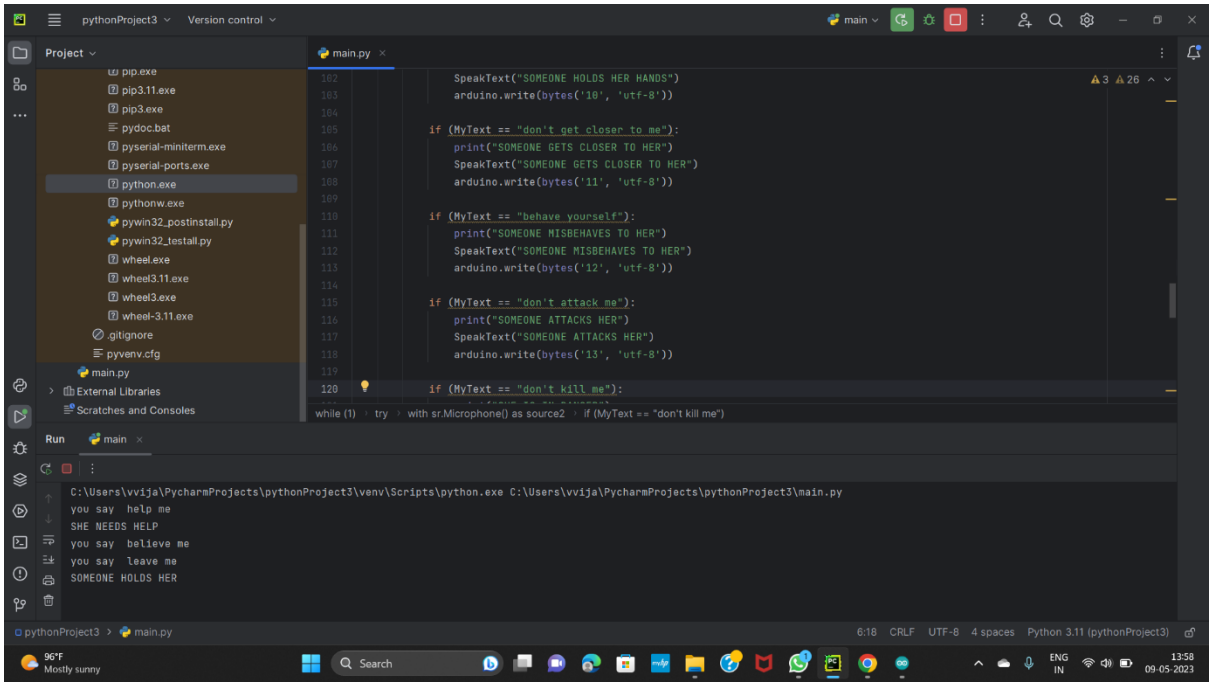
SCREEN SHOT

Hardware Setup

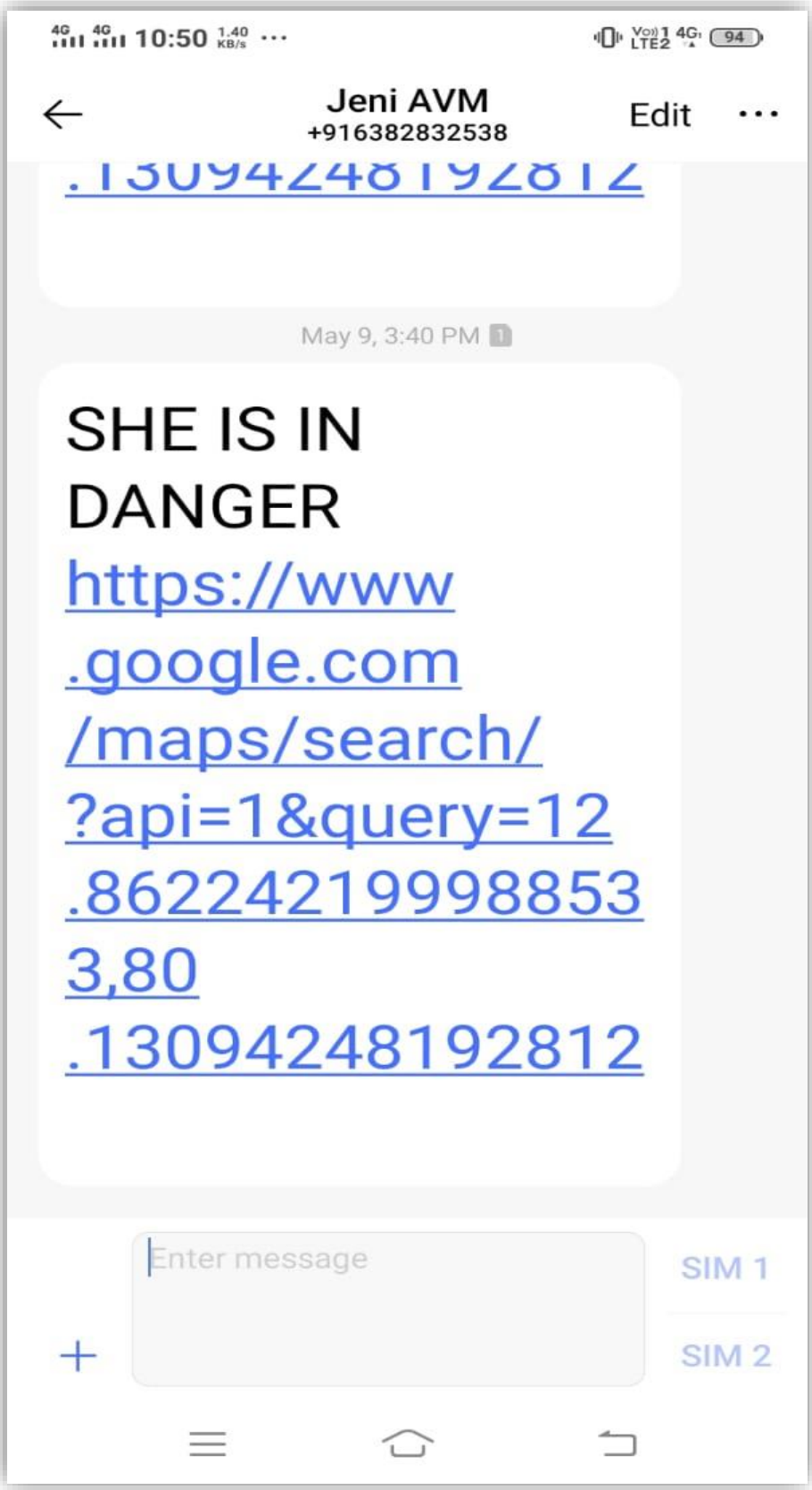
The system combines cutting-edge AIoT technologies to provide real-time monitoring of a child's emotional state and location. By leveraging speech emotion recognition, GPS tracking, and SMS messaging, it enhances child safety and delivers peace of mind to parents or guardians. The system's ability to integrate with other smart devices adds an additional layer of customization and convenience, making it a comprehensive solution for ensuring the well-being of children.



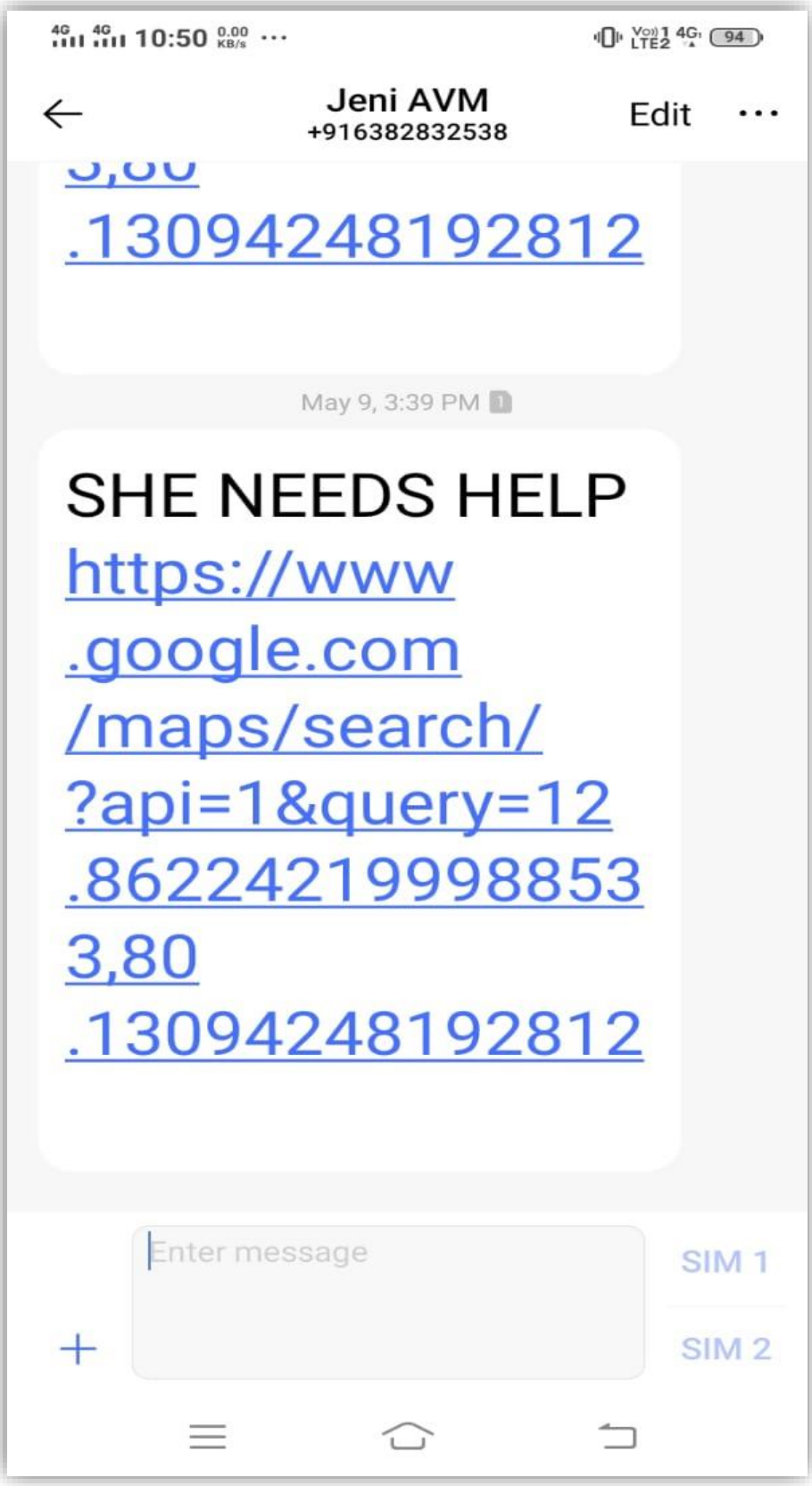
Code Uploaded On Arduino Nano



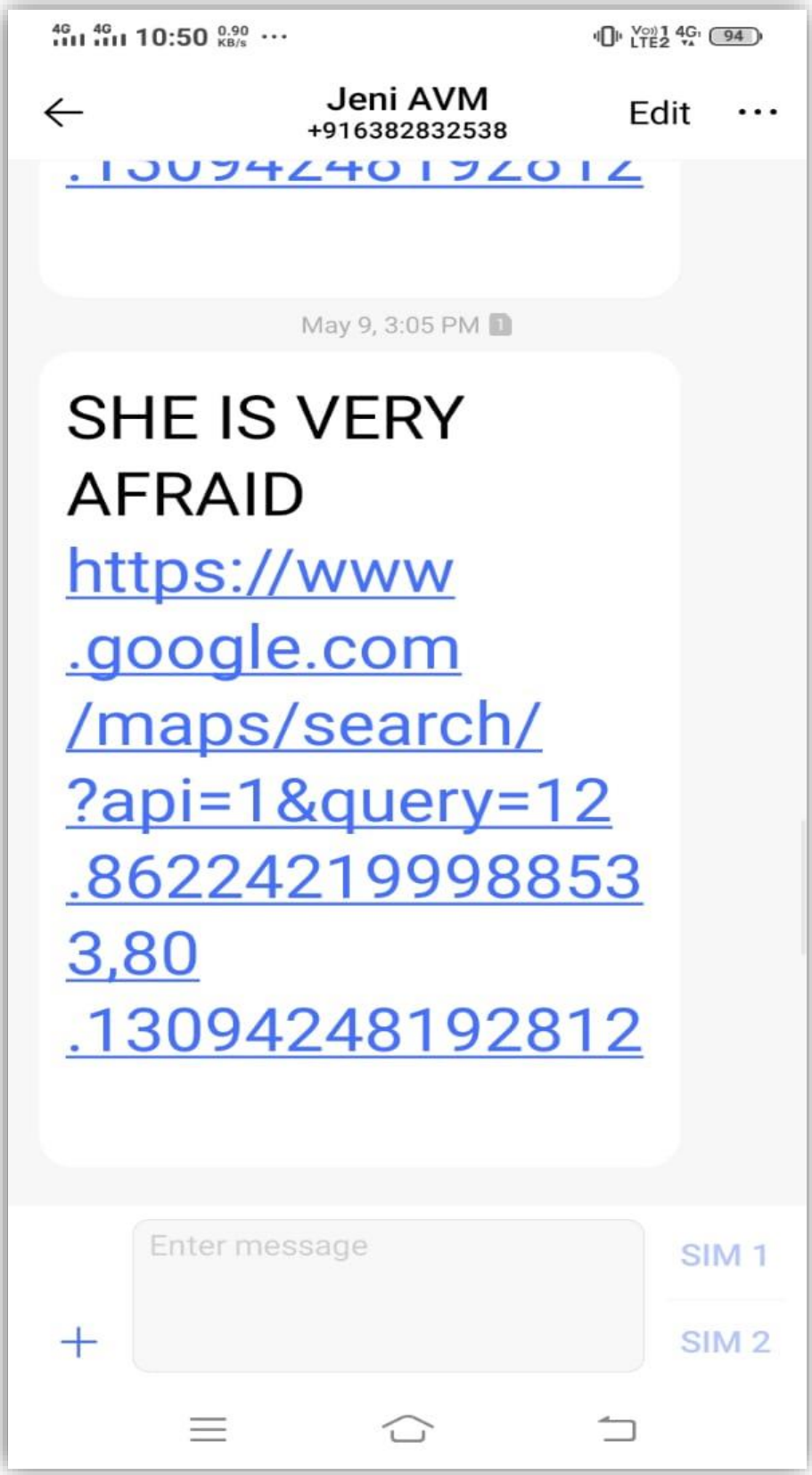
Speech Emotion Recognition Using Python Code



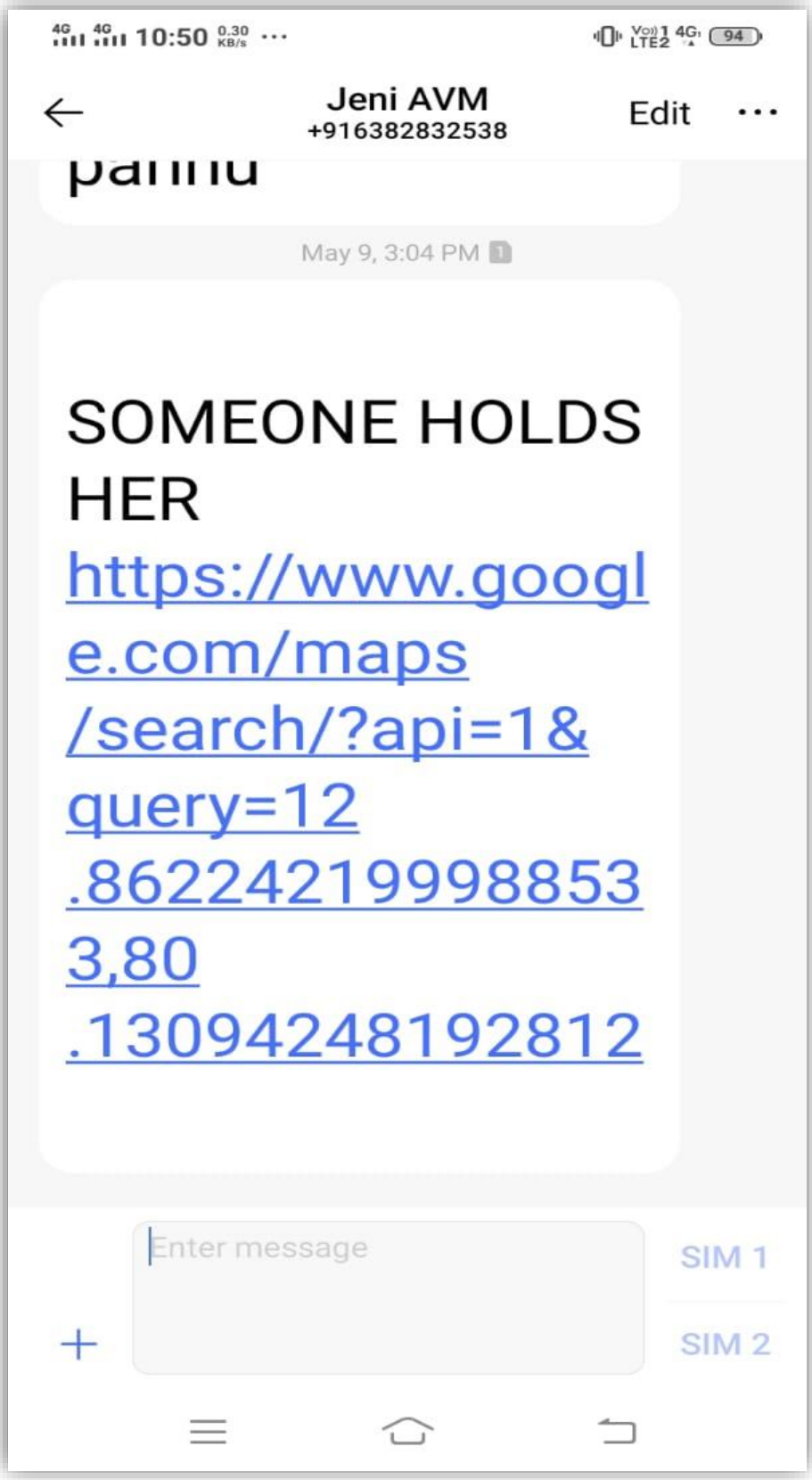
Sends Alert When She Is In Danger



Sends Alert When She Needs Help



Sends Alert When She Is Very Afraid



Sends Alert When Someone Holds Her

REFERENCES

1 Schuller, B., Batliner, A., Steidl, S., Seppi, D., & Laskowski, K. (2010). The INTERSPEECH 2010 Paralinguistic Challenge. In Interspeech.

2 Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., & Zafeiriou, S. (2016). Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In Proceedings of IEEE ICASSP.

3 Eyben, F., Scherer, K., Schuller, B., Sundberg, J., André, E., Busso, C., ... & Weninger, F. (2016). The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing. IEEE Transactions on Affective Computing, 7(2), 190-202.

4 B., Steidl, S., Batliner, A., Burkhardt, F., Devillers, L., Müller, C., & Narayanan, S. (2013). The INTERSPEECH 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism. In Interspeech.

5 Schuller, B., Batliner, A., & Burkhardt, F. (2009). A survey on paralinguistics in speech and language processing. Language and Speech,

6 Yang, Y., Gao, W., Wen, L., & Sun, J. (2017). A comparative study on speech emotion recognition using neural networks. Neurocomputing, 237, 88-99.

7 El Ayadi, M., Kamel, M. S., & Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. Pattern Recognition, 44(3), 572-587.

8 Fan, W., Wang, D., Liu, X., Li, Z., Zhang, Y., & Xu, X. (2018). Speech emotion recognition using deep neural network and extreme learning machine. Information Sciences, 423, 295-306.

9 Wang, Y., Huang, Y., Liu, H., Liu, X., & Liu, Y. (2019). Speech emotion recognition using deep learning: A review. IEEE Access, 7, 108662-108677.

10 Saha, G., Sahoo, S. K., & Ghosal, A. (2020). Speech emotion recognition using hybrid deep learning models: A comprehensive review. IEEE Access, 8, 42381-42408.

11 Mollahosseini, A., Chan, D., & Mahoor, M. H. (2017). Going deeper in facial expression recognition using deep neural networks. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision.

12 Ekman, P., & Friesen, W. V. (1978). The facial action coding system: A technique for the measurement of facial movement. Consulting Psychologists Press.

13 Chen, L., Zhou, G., Zhang, H., Zhao, Y., & Yang, M. (2019). Speech emotion recognition using transfer

14 El Ayadi, M., Kamel, M. S., & Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. Pattern Recognition, 44(3), 572-587.

15 Lee, C. M., & Narayanan, S. S. (2005). Toward detecting emotions in spoken dialogs. IEEE Transactions on Speech and Audio Processing, 13(2), 293-303.

16 Gunes, H., & Schuller, B. (2013). Categorical and dimensional affect analysis in continuous input: Current trends and future directions. Image and Vision Computing, 31(2), 120-136.

17 Han, K., Kim, M., Lee, G., & Kim, H. (2019). Emotion recognition using speech signals: A review. Applied Sciences, 9(10), 2024.

18 Neiberg, D., & Kim, H. N. (2019). Emotion recognition from speech: A review. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 3(3), 1-30.

19 Liu, C., Zhu, Q., & Wang, H. (2018). Speech emotion recognition: A survey. IEEE Transactions on Affective Computing, 9(3), 237-256.

20 Alam, J., Riccardi, G., & Hakkani-Tur, D. (2016). Audio-based emotion recognition: A survey. IEEE Transactions on Affective Computing, 7(1), 3-14.

21 Kim, H. N., & Bang, J. W. (2020). Deep learning-based speech emotion recognition: A review. ACM Transactions on Interactive Intelligent Systems (TiiS), 10(3), 1-29.

22 Kwon, O., Kim, M., & Kim, H. (2020). Multi-modal speech emotion recognition: A review. Sensors, 20(12), 3334.

23 Hu, Y., Zhang, Y., Lu, Y., & Zhang, B. (2019). Speech emotion recognition: A review of the state-of-the-art. IEEE Access, 7, 107562-107577.

24 Lee, Y., Lee, C., Yoon, S., & Kim, J. (2019). Speech emotion recognition based on deep learning: A review. Journal of Engineering, 2019(10), 1-11.