VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi - 590 018, Karnataka, India

PROJECT REPORT

on

"SOLAR POWERED SELF VOICE ASSISTIVE DEVICE FOR RURAL VILLAGE SMART MEDICARE USING FINGERPRINT TECHNOLOGY"

Submitted in partial fulfillment of the requirements for the award of the Degree

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

by

Jinith M	**USN: 1DA21EC063**
Mahesh P	**USN: 1DA21EC076**
Samarth M Katageri	**USN: 1DA21EC127**
Dashrath Rao	**USN: 1DA21EC176**

Under the Guidance of

Dr. Tanuja Patgar

Assistant Professor, Department of ECE, Dr. AIT, Bengaluru - 560056

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

MALLATHAHALLI, OUTER RING ROAD, BENGALURU – 560056

2024-25

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

Mallathahalli, Bengaluru - 560056

Department of Electronics & Communication Engineering

Certificate

Certified that the project work entitled “**SOLAR POWERED SELF VOICE ASSISITIVE DEVICE FOR RURAL VILLAGE SMART MEDICARE USING FINGERPRINT TECHNOLOGY**”, carried out by **JINITH M**, bearing USN : **1DA21EC063**, **MAHESH P**, bearing USN:**1DA21EC076**, **SAMARTH M KATAGERI** bearing USN :**1DA21EC127**, **DASHRATH RAO**, bearing USN:**1DA21EC176**, bonafide students of Dr. Ambedkar Institute of Technology, Bengaluru – 560056 in partial fulfillment for the award of Bachelor of Engineering in Electronics and Communication Engineering of the Visvesvaraya Technological University, Belagavi during the year 2024–2025. It is certified that all the corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements.

Signature of the Guide	**Signature of the HOD**	**Signature of the Principal**
(Dr. Tanuja Patgar)	**(Dr. Jambunath S. Baligar)**	**(Dr. M. N. Thippeswamy)**
(Assistant Professor)	**(Head of the Department)**	**(Principal)**
(Dept. of ECE, Dr. AIT)	**(Dept. of ECE, Dr. AIT)**	**(Dr. AIT)**

External Viva

Name of the Examiners **Signature with Date**

1.__________________ __________________

2. __________________ __________________

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY
Mallathahalli, Bengaluru - 560056

Department of Electronics & Communication Engineering

Declaration

We, **JINITH M**, bearing USN : **1DA21EC063**, **MAHESH P**, bearing USN:**1DA21EC076**, **SAMARTH M KATAGERI** bearing USN :**1DA21EC127**, **DASHRATH RAO**, bearing USN:**1DA21EC176**, hereby declare that, the project work entitled "**SOLAR POWERED SELF VOICE ASSISITIVE DEVICE FOR RURAL VILLAGE SMART MEDICARE USING FINGERPRINT TECHNOLOGY**" is independently carried out by us at Department of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology, Bengaluru-560056, under the guidance of **Dr. Tanuja Patgar,** Assistant Professor, Department of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology. The Project work is carried out in partial fulfillment of the requirement for the award of degree of Bachelor of Engineering in Electronics and Communication Engineering during the academic year 2024-2025.

Place: Bengaluru	Name & Signature of students
Date:	Jinith M
	Mahesh P
	Samarth M Katageri
	Dashrath Rao

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this project would be complete only with the mention of the people who made it possible, whose support rewarded our effort with success.

We are grateful to Dr. Ambedkar Institute of Technology for its ideals and its inspirations for having provided us with the facilities that have made this project a success.

We are grateful to our Principal **Dr. M N Thippeswamy**, Dr. Ambedkar Institute of Technology, who gave a continuous support and provided us comfortable environment to work in.

We would like to express our sincere thanks to **Dr. Jambunath S Baligar**, Professor and Head, Department of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology for his support. We pay out profound gratefulness and express our deepest gratitude to our project guide **Dr. Tanuja Patgar**, Assistant Professor, Department of Electronics and Communication Engineering for the suggestions and guidance.

We are thankful to our project coordinator **Dr. Meenakshi L Rathod**, Assistant Professor, Dept. of Electronics and communication for their advice, supervision and guidance throughout the course of the project.

It is our pleasure to acknowledge the cooperation extended by teaching staff and non- teaching staff members of Dept. of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology for the encouragement during project work. Finally, it gives immense pleasure to acknowledge the cooperation extended by family members, friends for the encouragement during this Project Work.

JINITH M,
MAHESH P,
SAMARTH M KATAGERI,
DASHRATH RAO

ABSTRACT

The Semi-Automated Medicine Vending Machine is a groundbreaking solution designed to enhance access to essential medications, particularly in underserved areas. Combining advanced hardware like Arduino MEGA and fingerprint recognition with seamless connectivity to the Adafruit IO cloud platform, this device modernizes pharmaceutical distribution. It tracks individual user profiles, monitors medicine inventories in real time, and offers manual transfer options, ensuring both security and ease of use. Functioning much like an ATM for medicine, the machine provides a user-friendly interface that makes it accessible to a wide range of users. By leveraging sensor data and cloud-based management, it delivers reliable pharmaceutical services, especially in locations with limited or no medical infrastructure, such as remote villages, highways, or disaster-prone areas.

Addressing critical healthcare challenges, the primary objective of this machine is to bridge gaps in healthcare delivery by providing immediate access to lifesaving medications during emergencies. Its design prioritizes simplicity and efficiency, offering rapid service without relying on traditional medical distribution systems. This makes it a vital resource in emergencies, where timely access to medicine can be the difference between life and death. By enabling equitable healthcare access and reliability, the Semi-Automated Medicine Vending Machine represents a transformative approach to solving healthcare inequity. It embodies a sustainable, innovative solution with the potential to improve the quality of life and well-being for vulnerable populations, reinforcing its role as a critical tool for modern healthcare delivery.

INDEX

LIST OF FIGURES

CHAPTER 1

1.1. INTRODUCTION

The **Semi-Automated Medicine Vending Machine (SMVM)** is an innovative healthcare solution aimed at addressing the challenges of accessing essential medicines in areas where traditional healthcare infrastructure is limited or non-existent. By utilizing biometric fingerprint authentication, the system ensures secure and personalized dispensing of medicines. This project is designed to provide a reliable, efficient, and scalable alternative to conventional pharmacies, making it particularly beneficial for remote locations such as rural villages, highways, and underserved regions.

Similarities to an ATM Machine

The SMVM operates similarly to an Automated Teller Machine (ATM), offering convenience and security through technology. In an ATM, users authenticate themselves with a bank card and PIN to access cash. Likewise, the SMVM verifies users' identities through fingerprint scanning to dispense medicines securely. Both systems aim to enhance accessibility while ensuring user-specific services. Just as ATMs provide financial independence in remote areas, SMVM units enable decentralized access to healthcare, making essential medicines readily available.

Challenges in Remote Areas

Many remote areas face significant healthcare challenges due to the lack of hospitals, pharmacies, and trained medical personnel. Rural villages and highways often lack proper infrastructure, forcing residents to travel long distances to access basic healthcare facilities. This delay in treatment increases health risks and often leads to preventable complications. Moreover, the logistical barriers, coupled with a scarcity of resources, make it difficult to ensure a consistent supply of essential medicines in such areas.

Addressing Challenges with SMVM

The SMVM provides a practical solution to these issues by decentralizing access to medicines. With its compact and portable design, the machine can be installed in remote locations, ensuring that essential medicines are always within reach. The system operates in two modes: **Automatic Mode**, where the machine autonomously dispenses medicine after verifying the user's fingerprint and prescription data, and **Manual Mode**, where an operator can control the dispensing process if necessary. Additionally, SMVM units are available 24/7, unlike traditional pharmacies with limited operating hours, ensuring uninterrupted access to essential medicines.

By leveraging fingerprint authentication, the SMVM ensures that medicines are dispensed only to authorized users, preventing misuse and ensuring equitable distribution. This technology-driven approach makes the system scalable, adaptable, and highly effective in addressing healthcare challenges in underserved regions.

The SMVM project bridges the gap in healthcare delivery for remote areas, much like ATMs have transformed access to financial services. By combining automation with biometric security, it provides a safe, efficient, and accessible method of dispensing medicines. This initiative not only addresses the immediate healthcare needs of underserved communities but also paves the way for a more equitable and healthier society.

1.2. PROBLEM STATEMENT

- **Limited Access to Medications:**
  Many rural and remote areas lack nearby pharmacies or healthcare facilities, leaving residents without essential medications during emergencies, particularly at night when conventional pharmacies are closed.
- **Inefficient Pharmacy Operations:**
  Traditional pharmacies face challenges like long queues, stockouts, and manual processes, leading to delays, errors, and frustration for patients and staff, ultimately compromising timely access to care.
- **Need for 24/7 Healthcare Solutions:**
  The absence of round-the-clock medication access creates service gaps, particularly in emergencies, highlighting the need for automated, reliable systems to ensure timely treatment anytime.

1.3: OBJECTIVES

- User authentication using fingerprint technology
- Integrating sensors (Temperature, FSR) and creating database for fever, cold, vomiting, cough, body pain
- Implementation of vending machine for tablets
- Using cloud technology (Adafruit IO), machine dispenses the medicines as per user’s requirement

CHAPTER 2

2.1: LITERATURE SURVEY

Literature review involves reviewing and analysing relevant literature to identify gaps, trends, and key findings in the field of study. A literature survey serves as a foundation for new research and helps researchers understand the current state of knowledge and identify areas for further investigation.

[1] The paper presents an intelligent system created to dispense food, medicine, and water. This system is designed to tackle problems like hunger, thirst, and the spread of infectious diseases by automating the distribution of these essential supplies. The goal is to ensure that people receive the necessary items under specific constraints and conditions.

[2] The paper discusses the design and implementation of a vending machine powered by solar energy to dispense medicines. The main goal of this system is to provide easy access to essential medicines, especially in remote or rural areas where medical stores might not be readily available. The machine is designed to operate 24/7, ensuring that people can obtain necessary medications anytime, without the need for manual intervention. It also incorporates a user-friendly interface for contactless payment and efficient delivery of medicines.

[3] The paper focuses on designing a vending machine that dispenses medicines at any time. The system uses an embedded system to automate the dispensing process, ensuring that users can access their prescribed medications conveniently and reliably. The machine is equipped with a card reader that validates user credentials against an e-prescription stored on a smart card. This ensures that only authorized users can access the medicines they need. The system is designed to be especially useful in remote or rural areas where access to medical stores might be limited.

[4] The paper explores the development of an unmanned pharmacy system that leverages advanced technologies such as artificial intelligence (AI), 5G communication, and big data. The system aims to provide efficient and intelligent management of medical resources, especially in remote or underserved areas. By integrating AI and remote connection technologies, the system facilitates auxiliary diagnosis and health management, ensuring that patients can access medical care and medications more easily. The cloud database enables fine management, resource allocation, and epidemic prediction through data analysis, creating a closed-loop health management system.

[5] The paper introduces the Automatic Health Machine (AHM), leveraging IoT and AI to provide medical assistance during pandemics and emergencies. The system offers virtual health check-ups, connects users with doctors online, and can book appointments for tests or ambulances. It also dispenses emergency medicines and provides electronic prescriptions, ensuring access to healthcare even in challenging circumstances.

[6] The study describes an innovative system that uses IoT technology to assist patients, particularly the elderly, in taking their medications accurately and on time. Designed to enhance medication adherence, it prevents missed or incorrect doses and facilitates real-time communication between patients and caregivers. This system ultimately aims to promote patient independence and improve overall healthcare outcomes.

[7] The paper explores the implementation of medicine vending machines along highways to provide immediate first-aid assistance in remote areas. These machines are designed to be accessible 24/7, offering essential medical supplies to accident victims or individuals in urgent need of first aid. The goal is to reduce the time it takes for people in remote locations to receive critical medical help, potentially saving lives.

[8] Leveraging RFID technology, the Smart Drug Delivering System in Hospitals enhances medication management by automating the dispensing process. This system ensures accurate and timely delivery of drugs, reduces human errors, and improves overall efficiency in hospital pharmacies.

[9] Medicine Vending Machine Using RFID employs RFID technology to streamline medication dispensing. This innovative system ensures precise and timely drug delivery, minimizes human errors, and boosts overall efficiency in healthcare settings, particularly hospitals.

CHAPTER 3

PROPOSED WORK

3.1: METHODOLOGY

- User Authentication Using Fingerprint Technology: The system employs a fingerprint scanner to authenticate users based on their Aadhaar profiles. This ensures secure and reliable identification, enabling access only to verified individuals. The fingerprint data is matched against a pre-stored database, ensuring the correct user is served.

- Integration of Sensors and Processing via Arduino MEGA: Multiple sensors are integrated into the system to collect user health parameters:
  - Temperature Sensor detects the user’s body temperature.
  - FSR Sensor provides input for touch or pressure-based interactions.

  The Arduino MEGA processes real-time data from these sensors to assess the user’s condition and determine appropriate medications.

- Database Creation for Common Symptoms Using Adafruit IO Cloud: A symptom-based database is developed using the Adafruit IO cloud platform. The system categorizes user inputs and sensor readings into predefined medical conditions such as fever, cold, vomiting, cough, and body pain. This database facilitates accurate mapping between user symptoms and recommended medications.

- Automation of the Medicine Vending Machine: The vending machine operates automatically, dispensing tablets based on processed data. Once the system identifies the condition and selects the appropriate medication, the Arduino MEGA sends a command to release the specific medicine. This ensures a seamless and error-free dispensing process.

- Voice Assistant System: A voice assistance module integrated with Arduino MEGA provides audio guidance to users. It helps navigate the system, explains medication options, and ensures a user-friendly experience, particularly for individuals with limited technical knowledge or literacy.

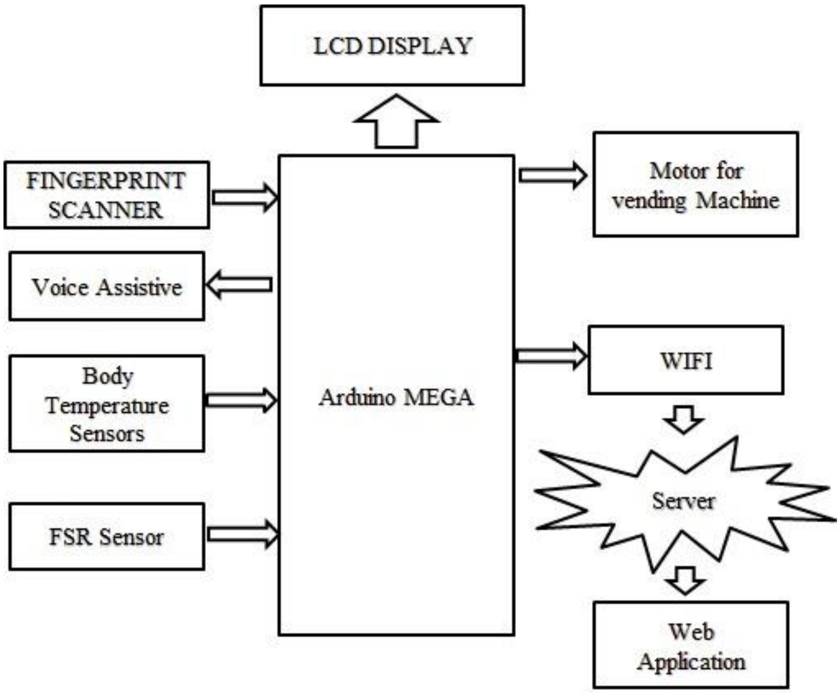


Fig. 3.1 Block Diagram of SMVM system

3.2: REQUIREMENTS AND SPECIFICATIONS

a) Arduino MEGA 2560 Rev3 CP2102 Board:

The Arduino Mega 2560 is a popular microcontroller board based on the ATmega2560. It is designed for projects that require more input/output (I/O) pins, memory, or processing power compared to boards like the Arduino Uno. Below are the detailed specifications of the Arduino Mega 2560:

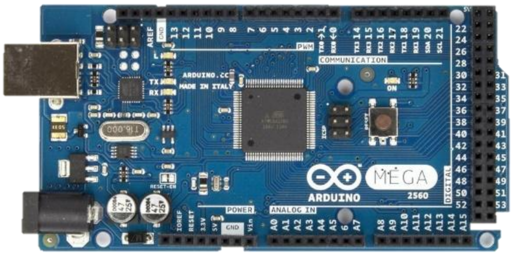


Fig. 3.2 Arduino MEGA Board

Specifications

Feature	**Description**
Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 can be used as PWM outputs)
Analog Input Pins	16

PWM Pins	15
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB (8 KB used by the bootloader)
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	Pin 13
Dimensions	101.52 mm x 53.3 mm
Weight	Approximately 37 g
USB Connector	Type-B USB port

Key Features

1. **Multiple Communication Options**:
    - 4 UARTs (hardware serial ports)
    - I2C and SPI support
2. **Programming**:
    - Can be programmed using Arduino IDE via a USB connection.
    - Supports in-system programming (ISP).
3. **Power Options**:
    - Powered via the USB cable or an external power supply (barrel jack).
    - Automatic power selection between USB and external power.
4. **Reset Protection**:
    - Includes an auto-reset function to simplify programming.
5. **Compatibility**:
    - Compatible with most Arduino shields designed for the Uno or similar form factors.

The Mega 2560 is ideal for complex projects requiring many sensors, actuators, or extensive connectivity. Examples include robotics, home automation, and data logging systems.

b) 2-channel Relay Module:

A 2-channel relay module is a small circuit board that allows you to control two separate appliances or devices using a low-voltage signal from a microcontroller. It essentially acts as a switch, but can handle higher currents than your microcontroller can safely provide.

Input Signal Voltage: The voltage required to trigger the relay on a single channel, often ranging from 3.3V to 5V.

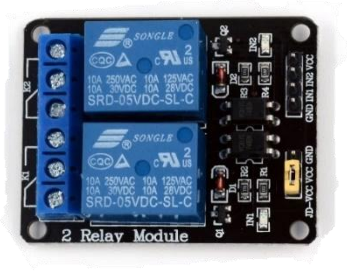


Fig. 3.3 2-channel relay module

Output Voltage: This varies depending on the model but can handle switching AC voltages in the range of 24V to 220V and DC voltages up to 30V.

Additional Features:

- **Optocouplers:** Provide electrical isolation between the control input and the high-voltage output for safety.
- **Status LEDs:** LEDs indicate the on/off state of each relay for visual indication.
- **Trigger Type:** Can be active high or active low depending on the model, requiring a high or low voltage signal respectively to activate the relay.

c) Single-channel Relay with Fan as coolant for Vending Machine:

A **single-channel relay** acts as a switch that allows a low-power circuit (e.g., from a microcontroller like **Arduino MEGA 2560** or **NodeMCU ESP8266**) to control a high-power device like a fan. The relay enables the activation or deactivation of the fan based on the temperature readings or any specific conditions set in the system.

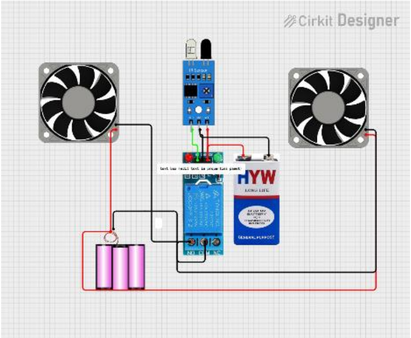


Fig. 3.4 Circuit of Single-channel relay module with fan connection

Specifications of a Single-Channel Relay

Specification	**Details**
Type	Electromechanical or Solid State Relay
Input Voltage	3V to 5V (commonly 5V or 3.3V for control circuit)
Control Signal	Digital signal (High or Low) from a microcontroller (e.g., Arduino)

Relay Switching Voltage	110V to 250V AC or 10V to 30V DC
Relay Switching Current	10A (AC) or 30A (DC) depending on the relay model
Relay Coil Resistance	Typically around 60Ω to 100Ω
Relay Contact Type	Normally Open (NO), Normally Closed (NC)
Relay Type	Single channel (can control one device)
Isolation	Optocoupler isolation (for signal protection)
Operating Temperature Range	-40°C to 85°C (depending on the relay)
Dimensions	Typically, small (e.g., 5 cm x 3 cm)

Working of the Relay

- The **microcontroller** sends a signal (usually HIGH or LOW) to the relay's input side.
- The relay closes the switch, allowing current to flow from the power supply to the **fan** or cooling device.
- The relay can be turned ON or OFF based on the logic in the controller, which can be linked to temperature sensors or a predefined timer.

Fan as a Cooling System in Vending Machines

The **fan** connected to the relay acts as the **cooling system** for the vending machine. This is crucial for maintaining a consistent temperature to preserve the quality of the products inside. Typically, an **AC fan** or **DC fan** is used based on the system's voltage requirements.

Specifications of Fans for Vending Machine Cooling

Parameter	**Details**
Type	AC or DC fan
Voltage	Typically, 5V, 12V, or 24V (DC) or 110V to 220V (AC)
Current Draw	0.2A to 0.5A (for small DC fans), 0.1A to 0.3A (for 12V DC)
Fan Speed	1000 to 5000 RPM (depends on fan size and application)
Airflow (CFM)	30 to 200 CFM (Cubic Feet per Minute)
Noise Level	20 dB to 50 dB (depends on fan size and design)
Size	40mm to 120mm diameter (depending on application)
Lifespan	Typically, 30,000 to 50,000 hours (depending on quality)
Mounting Type	Screws, brackets, or mounting holes depending on fan design

Cooling Capacity	Based on airflow (CFM) and power consumption. Suitable for small cooling needs like in vending machines.

Types of Fans Used in Vending Machines

1. **DC Fans (12V or 5V)**:
   - Used in **low-power** applications, such as small vending machines or where power consumption is a concern.
   - Can be controlled directly via the relay, with the **microcontroller** switching the fan ON and OFF based on environmental conditions.
2. **AC Fans (110V or 220V)**:
   - Used in **larger vending machines** that require stronger cooling, such as those storing beverages.
   - The relay must be rated to handle the high-voltage current (AC) for controlling the fan.
   - Provides higher airflow compared to typical DC fans.
3. **Brushless Fans**:
   - Long lifespan and low noise, ideal for 24/7 operations in vending machines.

d) Linear Voltage Regulator Boards:

A linear voltage regulator board is a device that converts a higher input voltage to a stable, lower output voltage, providing regulated power for electronic components and circuits. Linear voltage regulators are simple to use and provide clean, noise-free output but are less efficient compared to switching regulators.

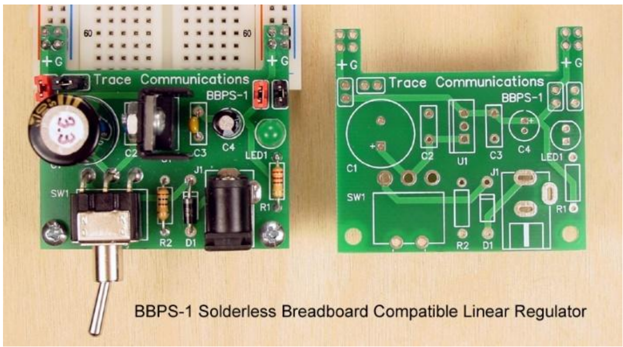


Fig. 3.5 Linear Voltage Regulator

Here’s a general overview and key specifications of linear voltage regulator boards:

Specifications

Parameter	**Description**
Input Voltage Range	Typically, 7V to 35V (varies by model, e.g., 7805 supports up to 35V).

Output Voltage	Fixed or adjustable (e.g., 5V, 3.3V, 12V, or user-configurable).
Output Current	Typically ranges from 100 mA to 2A, depending on the regulator (e.g., LM7805 provides up to 1.5A).
Dropout Voltage	The minimum voltage difference between input and output for proper regulation (e.g., 2V for LM7805).
Efficiency	Low, typically 40%-60% (energy is dissipated as heat).
Quiescent Current	The small amount of current consumed by the regulator for its operation (typically a few milliamperes).
Thermal Shutdown	Many regulators include protection against overheating.
Short-Circuit Protection	Built-in protection against output overcurrent or short circuits.
Operating Temperature	Usually -40°C to +125°C (depends on the specific IC).
Power Dissipation	Dependent on input-output voltage difference and output current (P = (Vin - Vout) × Iout).
PCB Dimensions	Varies based on the design; typical small boards are ~20mm x 30mm.

Common Linear Voltage Regulators

1. **Fixed Voltage Regulators**:
   - **LM7805**: Provides a 5V output.
   - **LM7812**: Provides a 12V output.
   - **LM317L**: Provides a fixed voltage with lower current capacity.
2. **Adjustable Voltage Regulators**:
   - **LM317**: Adjustable output voltage (1.25V to 37V) with external resistors.
   - **LD1117**: Low-dropout adjustable regulator (supports 3.3V fixed and adjustable outputs).

Applications

- Power supply for microcontrollers (e.g., Arduino, Raspberry Pi).
- Noise-sensitive circuits (e.g., audio amplifiers).
- Battery chargers and small electronic devices.

If you need specific details about a particular regulator board, let me know!

e) aPR33A3 V2 Board:

The **aPR33A3 V2 board** is a versatile voice recording and playback module designed for embedded applications. It features the aPR33A3 IC, which is capable of storing multiple voice messages with high-quality playback. The board is often paired with a speaker module for standalone or integrated audio systems.

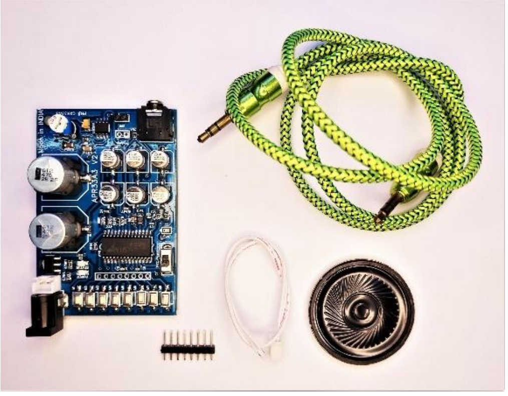


Fig. 3.6 aPR33A3 V2 board with speaker module

Specifications of aPR33A3 V2 Board

Parameter	**Details**
Chipset	aPR33A3
Memory Type	Non-volatile flash memory (voice data is retained without power).
Recording Duration	Up to **93 seconds** at 8 kHz sampling rate (higher quality results in shorter duration).
Channels	1 channel (mono recording and playback).
Sampling Rates	6 kHz, 8 kHz, 10 kHz, or 12 kHz (adjustable for varying quality and duration).
Control Interface	Push buttons (direct trigger) or microcontroller control (e.g., via GPIO pins).
Trigger Modes	Edge trigger, level trigger, or key trigger.
Input/Output Options	Built-in microphone input, auxiliary input, and speaker or line output.
Operating Voltage	3.3V to 5V DC.
Operating Current	Approx. 20-30 mA during operation.
Speaker Compatibility	Can drive an 8-ohm speaker directly through the onboard amplifier.
PCB Dimensions	Approximately 50mm x 30mm.

Features:

1. **Voice Recording**:
   - Supports direct voice recording via an onboard microphone or external audio input.
   - Provides high-quality audio storage in non-volatile memory.
2. **Voice Playback**:
   - Can play back stored audio through a connected speaker or line output.
   - Supports looping playback and individual message playback.
3. **Standalone Operation**:
   - Operates without the need for a microcontroller when using onboard push buttons.
4. **Microcontroller Integration**:
   - Easily controlled by a microcontroller (e.g., Arduino, Raspberry Pi) for advanced applications.
   - Simple GPIO interface for recording and playback commands.
5. **Multiple Modes**:
   - Configurable playback modes (single track, sequential tracks, looping).

Specifications of the Speaker Module:

Parameter	**Details**
Speaker Impedance	8 ohms
Speaker Size	Typically 28mm to 40mm diameter (varies based on the module).
Power Rating	0.5W to 1W (matches the board's output capabilities).
Frequency Range	300 Hz to 5 kHz (suitable for voice applications).
Connector Type	Solder pads or screw terminals (varies by design).

Applications:

- Voice-guided systems.
- Alarm and notification systems.
- Toys and interactive devices.
- Audio prompts for industrial equipment.
- Educational and DIY projects.

f) NodeMCU ESP8266 Board with built-in WiFi:

The **NodeMCU ESP8266** is a low-cost, open-source development board based on the ESP8266 Wi-Fi module. It is widely used for IoT (Internet of Things) applications due to its built-in Wi-Fi connectivity and ease of use with platforms like Arduino and Lua.

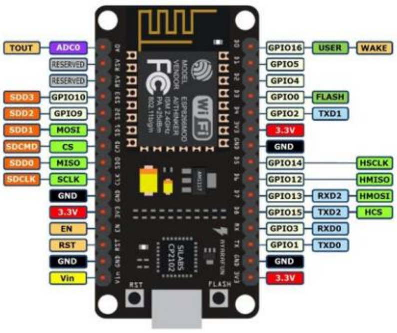


Fig. 3.7 NodeMCU ESP8266 WiFi board

Specifications

Feature	Details
Microcontroller	ESP8266 (Tensilica Xtensa LX106, 32-bit RISC processor)
Operating Voltage	3.3V
Input Voltage (via VIN)	4.5V to 10V
Digital I/O Pins	11 (can be configured as GPIO, PWM, I2C, etc.)
Analog Input Pin	1 (ADC input range: 0-3.3V)
Flash Memory	4 MB (varies by model, e.g., some have 1 MB or 8 MB)
RAM	128 KB (SRAM)
Wi-Fi Standards	IEEE 802.11 b/g/n (2.4 GHz)
Wi-Fi Security	WEP, WPA, WPA2
Clock Speed	80 MHz (can be overclocked to 160 MHz)
USB Connector	Micro-USB
Interfaces	UART, SPI, I2C, PWM, ADC
Dimensions	Approximately 58mm x 31mm
Power Consumption	Active mode: ~70 mA; Sleep mode: <20 µA

Key Features

1. **Built-in Wi-Fi**:
   - Integrated 802.11 b/g/n Wi-Fi module with support for both client and access point (AP) modes.
   - Supports TCP/IP protocol stack for internet connectivity.
2. **Programming Options**:
   - Programmable using the Arduino IDE, Lua script, or MicroPython.

  - USB-to-serial converter (CP2102 or CH340) simplifies programming via micro-USB.

3. **Versatile I/O**:
   - GPIO pins can be used for digital input/output, PWM, I2C, SPI, and more.
   - One ADC pin for reading analog signals.
4. **Power Options**:
   - Can be powered via the micro-USB port or through the VIN pin.
   - Voltage regulator onboard to ensure 3.3V operation.
5. **Compact Design**:
   - Small form factor, ideal for IoT and embedded systems.

Applications

- **IoT Projects**:
  - Smart home automation.
  - Remote monitoring systems.
  - Wi-Fi-enabled appliances.
- **Networking Projects**:
  - HTTP server or client for web-based interfaces.
  - MQTT protocols for IoT communication.
- **DIY Electronics**:
  - Sensor networks.
  - Robotics.
- **Prototyping**:
  - Quick development of connected applications.

Pinout

The NodeMCU board has labeled pins, but here's a summary:

Pin	Function
VIN	Input voltage (4.5V–10V).
3V3	3.3V output (regulated).
GND	Ground.
D0–D8	Digital GPIO pins (configurable).
A0	Analog input (0-3.3V).

TX/RX	Serial communication (UART).
RST	Reset pin.

g) Servomotors in Vending Machines

Servomotors are commonly used in vending machines for controlling the mechanical movements, such as dispensing products, rotating trays, or opening/closing doors. Servos offer precise control over the position, speed, and torque, making them ideal for applications requiring high accuracy and reliability.

Fig. 3.8 Servomotor

Specifications of Servomotors Used in Vending Machines

Here are the key specifications and features of servomotors typically used in vending machines:

Parameter	**Details**
Type	Standard (e.g., SG90, MG90, MG996R) or High Torque Servos.
Torque	1.5 kg·cm to 20 kg·cm or higher (depending on the servo).
Voltage	Typically 4.8V to 6.0V (standard); high-torque models may operate at higher voltages (up to 12V).
Operating Speed	0.12 sec/60° to 0.2 sec/60° at 4.8V (varies based on servo model).
Rotation Angle	Typically 180° (standard servos), but some can rotate 360° (continuous rotation).
Control Signal	PWM (Pulse Width Modulation) signal (usually a frequency of 50Hz).
Current Draw	Around 100-250mA at no load; can increase up to 1A or more under load.
Size and Weight	Varies based on model, but generally small and compact (e.g., SG90: 22.5mm x 11.5mm x 29mm).
Duty Cycle	Typically 10-20 ms (with a 1.5 ms neutral position).
Servo Motor Gear Type	Plastic or metal gears (metal gears are stronger and last longer).
Operating Temperature	Usually -20°C to +60°C, but higher temperatures may require specialized servos.
Feedback Mechanism	Potentiometer-based feedback for position control.
Mounting Type	Standard 3-pin header or a specific mounting hole depending on model.

Life Expectancy	Varies, but typically servos are rated for several thousand to tens of thousands of cycles.

Role of Servomotors in a Vending Machine

1. **Product Dispensing**: Servos control mechanisms that release or dispense products from a tray or compartment in a vending machine. The servo ensures that the product is released accurately and at the right time.
2. **Tray Rotation**: Some vending machines have rotating trays to select products. Servos precisely rotate trays based on user input or a fingerprint authentication system.
3. **Access Control**: In machines with secure product compartments (e.g., for high-value items), servos can control the opening and closing of locks, ensuring that the correct item is dispensed only after successful authentication.
4. **Coin/Token Dispensing**: For machines that dispense coins or tokens, servos can be used to release them based on the transaction amount.
5. **Lift Mechanisms**: For vending machines that require lifting or shifting trays (e.g., to avoid jamming or reach a higher storage compartment), servos can be used to move these trays with precision.

3.3: IMPLEMENTATION

Hardware Components:

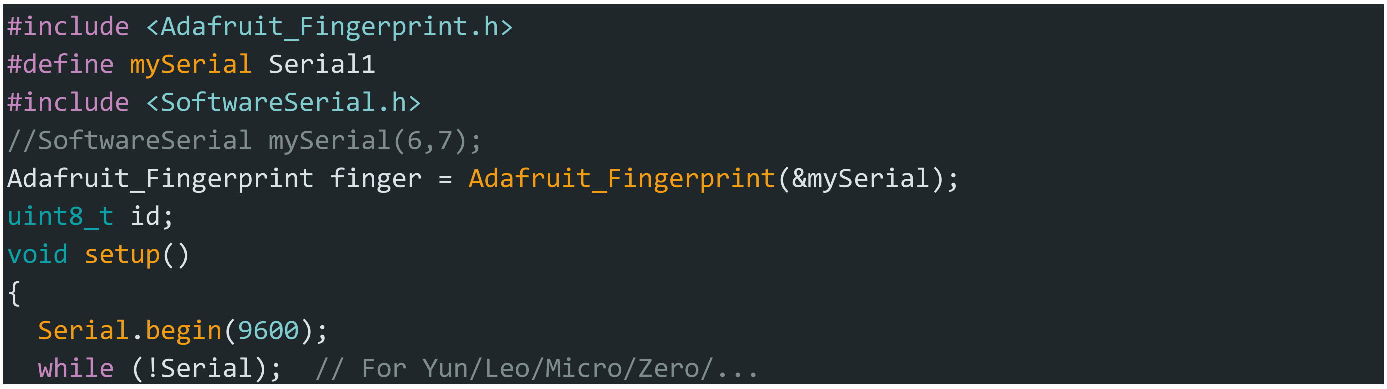
- **Microcontroller:** Arduino MEGA board
- **Sensors:**
  - 1 x Dallas Temperature Sensors
  - 1 x Adafruit Optical Fingerprint Scanner
- **Relays:** 2 x 2-channel relays
- **Other Components:**
  - Plywood for circuit support
  - Jumper Wires
  - Voltage regulators – (1. 5V-2A; 2. 5V-1A)
  - 4 x Servomotors
  - Metal body for vending machine

Implementation Steps:

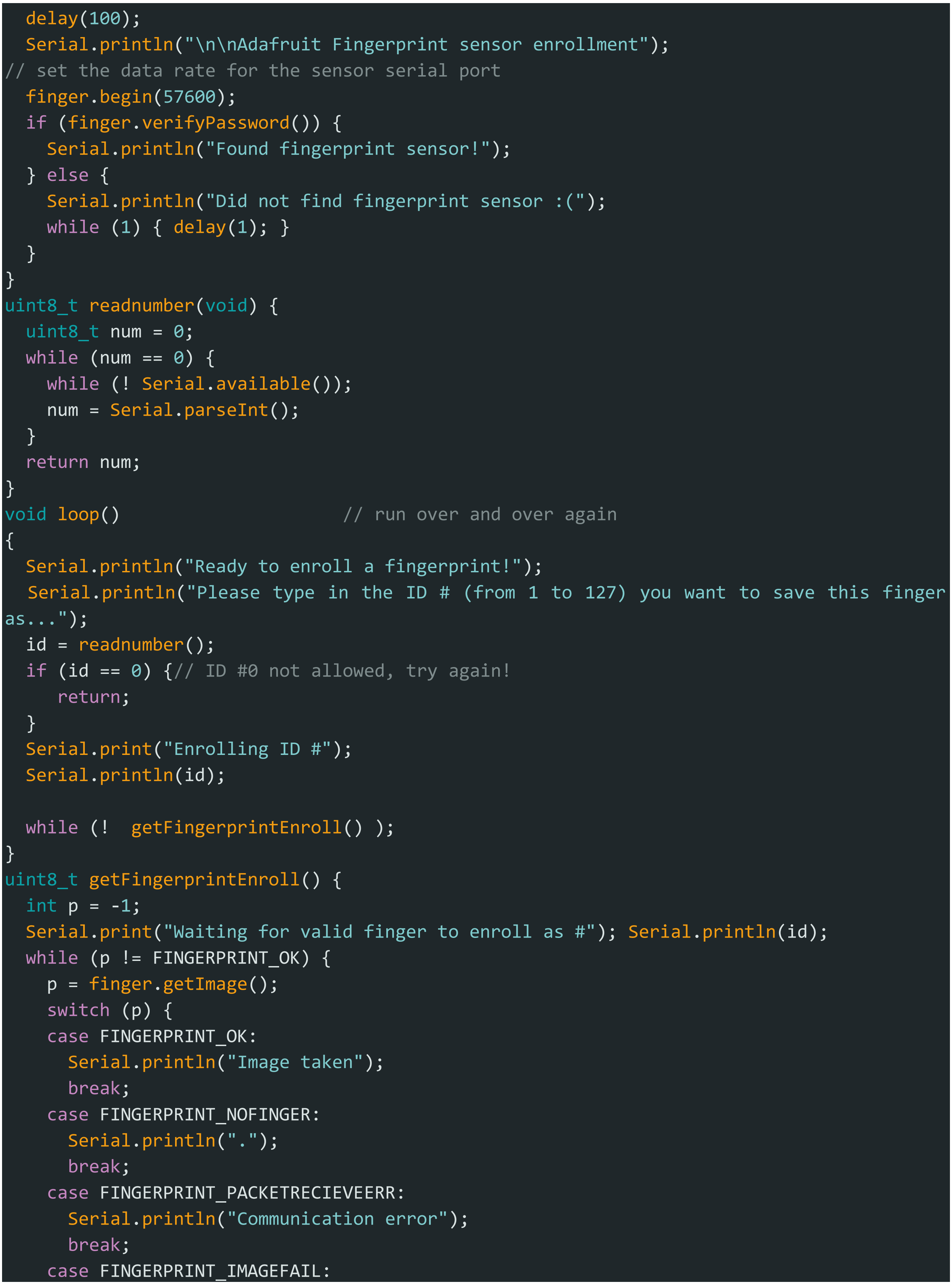
1. **Circuit Design:**
   - Design the circuit schematic on paper or using software tools.
   - Connect the PIR sensors and sound sensor to the microcontroller's digital I/O pins through appropriate resistors.
   - Connect the relays to the microcontroller's digital I/O pins and power them with a separate power supply based on their specifications.
2. **Microcontroller Programming:**
   - Program the Arduino MEGA using C/C++.
   - The code will:
     - Initialize I/O pins for sensors and relays.
     - Implement signal processing routines to read data from temperature sensors and the sound sensor.
     - Define thresholds for temperature sensor detection and triggers.
     - Based on sensor data, control the relay outputs to turn on/off connected servomotors.
   - Consider incorporating debouncing techniques for the sensors to avoid false triggers due to transient signals.
3. **Assembly and Testing:**
   - Assemble the circuit on a breadboard following the designed schematic.
   - Upload the compiled code to the microcontroller board.
   - Ensure relays activate and deactivate servomotors as programmed.

Program:

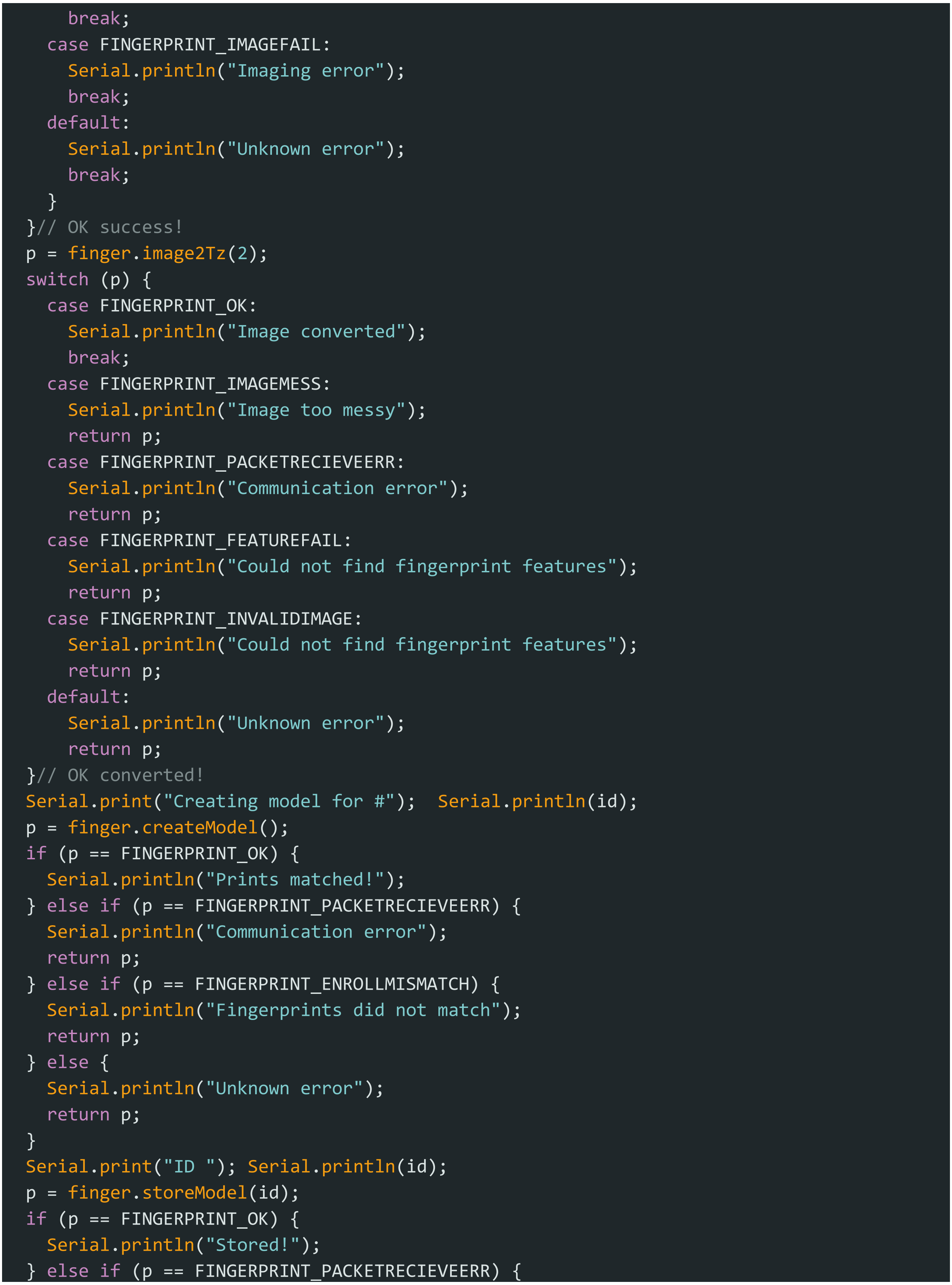
1. Program to read and store the fingerprint of a user for future authorization:



```
#include <Adafruit_Fingerprint.h>
#define mySerial Serial1
#include <SoftwareSerial.h>
//SoftwareSerial mySerial(6,7);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
uint8_t id;
void setup()
{
  Serial.begin(9600);
  while (!Serial);  // For Yun/Leo/Micro/Zero/...
```



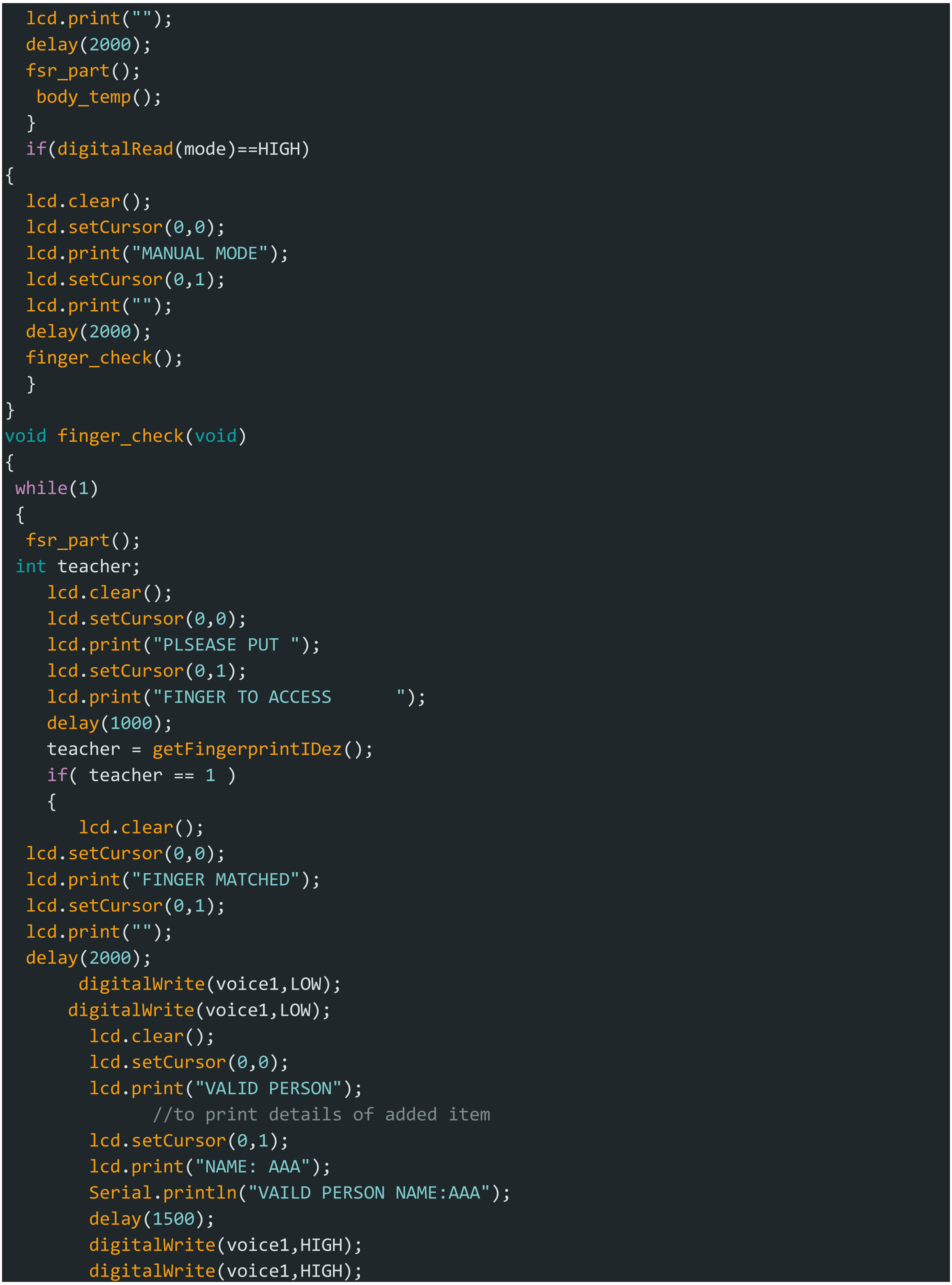
```
  delay(100);
  Serial.println("\n\nAdafruit Fingerprint sensor enrollment");
// set the data rate for the sensor serial port
  finger.begin(57600);
  if (finger.verifyPassword()) {
    Serial.println("Found fingerprint sensor!");
  } else {
    Serial.println("Did not find fingerprint sensor :(");
    while (1) { delay(1); }
  }
}
uint8_t readnumber(void) {
  uint8_t num = 0;
  while (num == 0) {
    while (! Serial.available());
    num = Serial.parseInt();
  }
  return num;
}
void loop()                       // run over and over again
{
  Serial.println("Ready to enroll a fingerprint!");
  Serial.println("Please type in the ID # (from 1 to 127) you want to save this finger
as...");
  id = readnumber();
  if (id == 0) {// ID #0 not allowed, try again!
     return;
  }
  Serial.print("Enrolling ID #");
  Serial.println(id);

  while (!  getFingerprintEnroll() );
}
uint8_t getFingerprintEnroll() {
  int p = -1;
  Serial.print("Waiting for valid finger to enroll as #"); Serial.println(id);
  while (p != FINGERPRINT_OK) {
    p = finger.getImage();
    switch (p) {
    case FINGERPRINT_OK:
      Serial.println("Image taken");
      break;
    case FINGERPRINT_NOFINGER:
      Serial.println(".");
      break;
    case FINGERPRINT_PACKETRECIEVEERR:
      Serial.println("Communication error");
      break;
    case FINGERPRINT_IMAGEFAIL:
        Serial.println("Imaging error");
        break;
      default:
        Serial.println("Unknown error");
        break;
      }
  }// OK success!
  p = finger.image2Tz(1);
  switch (p) {
    case FINGERPRINT_OK:
      Serial.println("Image converted");
      break;
    case FINGERPRINT_IMAGEMESS:
      Serial.println("Image too messy");
      return p;
    case FINGERPRINT_PACKETRECIEVEERR:
      Serial.println("Communication error");
      return p;
    case FINGERPRINT_FEATUREFAIL:
      Serial.println("Could not find fingerprint features");
      return p;
    case FINGERPRINT_INVALIDIMAGE:
      Serial.println("Could not find fingerprint features");
      return p;
    default:
      Serial.println("Unknown error");
      return p;
  }

  Serial.println("Remove finger");
  delay(2000);
  p = 0;
  while (p != FINGERPRINT_NOFINGER) {
    p = finger.getImage();
  }
  Serial.print("ID "); Serial.println(id);
  p = -1;
  Serial.println("Place same finger again");
  while (p != FINGERPRINT_OK) {
    p = finger.getImage();
    switch (p) {
    case FINGERPRINT_OK:
      Serial.println("Image taken");
      break;
    case FINGERPRINT_NOFINGER:
      Serial.print(".");
      break;
    case FINGERPRINT_PACKETRECIEVEERR:
      Serial.println("Communication error");
```



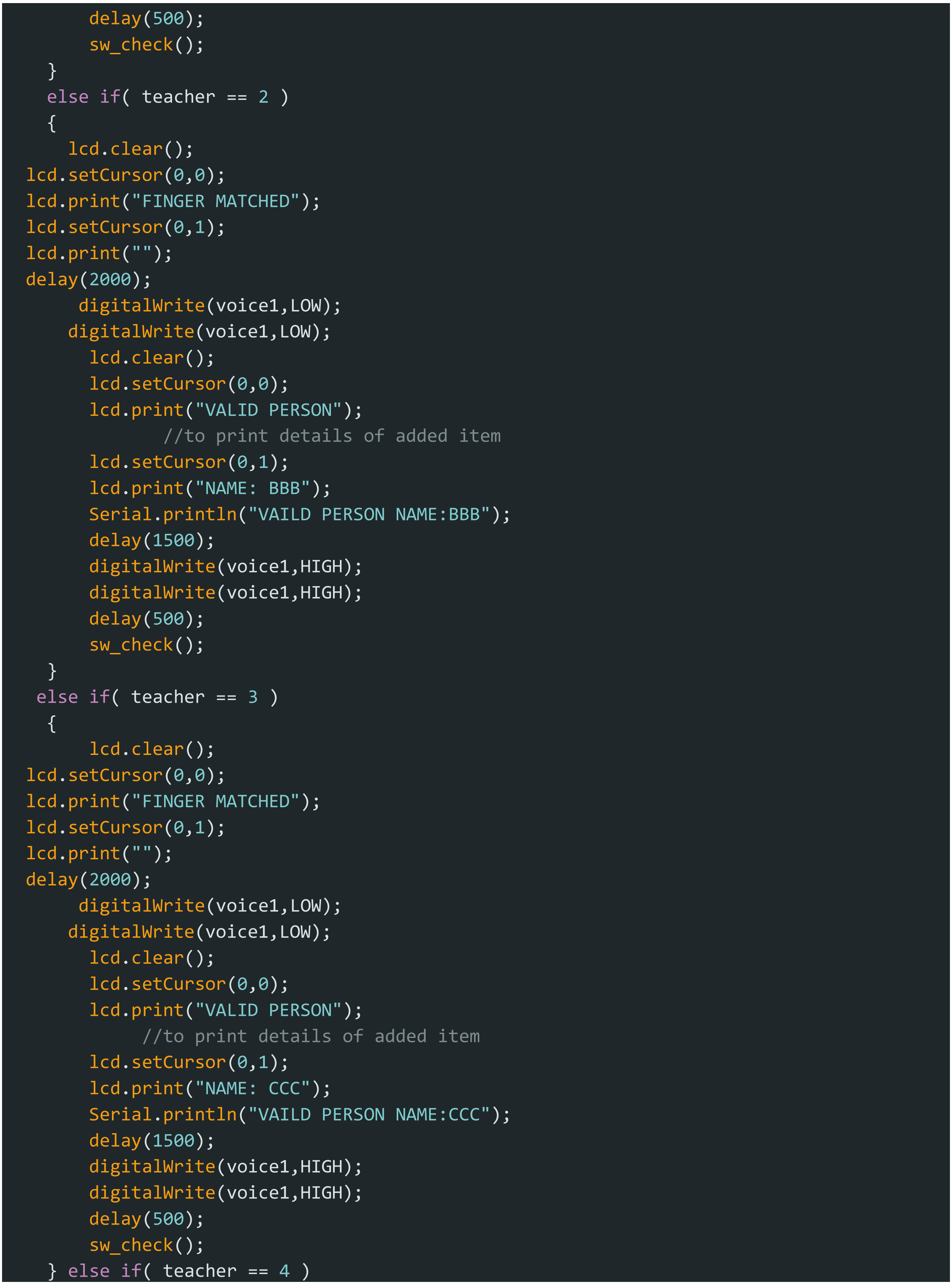
```
      break;
    case FINGERPRINT_IMAGEFAIL:
      Serial.println("Imaging error");
      break;
    default:
      Serial.println("Unknown error");
      break;
    }
  }// OK success!
  p = finger.image2Tz(2);
  switch (p) {
    case FINGERPRINT_OK:
      Serial.println("Image converted");
      break;
    case FINGERPRINT_IMAGEMESS:
      Serial.println("Image too messy");
      return p;
    case FINGERPRINT_PACKETRECIEVEERR:
      Serial.println("Communication error");
      return p;
    case FINGERPRINT_FEATUREFAIL:
      Serial.println("Could not find fingerprint features");
      return p;
    case FINGERPRINT_INVALIDIMAGE:
      Serial.println("Could not find fingerprint features");
      return p;
    default:
      Serial.println("Unknown error");
      return p;
  }// OK converted!
  Serial.print("Creating model for #");  Serial.println(id);
  p = finger.createModel();
  if (p == FINGERPRINT_OK) {
    Serial.println("Prints matched!");
  } else if (p == FINGERPRINT_PACKETRECIEVEERR) {
    Serial.println("Communication error");
    return p;
  } else if (p == FINGERPRINT_ENROLLMISMATCH) {
    Serial.println("Fingerprints did not match");
    return p;
  } else {
    Serial.println("Unknown error");
    return p;
  }
  Serial.print("ID "); Serial.println(id);
  p = finger.storeModel(id);
  if (p == FINGERPRINT_OK) {
    Serial.println("Stored!");
  } else if (p == FINGERPRINT_PACKETRECIEVEERR) {
    Serial.println("Communication error");
    return p;
  } else if (p == FINGERPRINT_BADLOCATION) {
    Serial.println("Could not store in that location");
    return p;
  } else if (p == FINGERPRINT_FLASHERR) {
    Serial.println("Error writing to flash");
    return p;
  } else {
    Serial.println("Unknown error");
    return p;
  }
}
```

2. Program for working of SMVM:

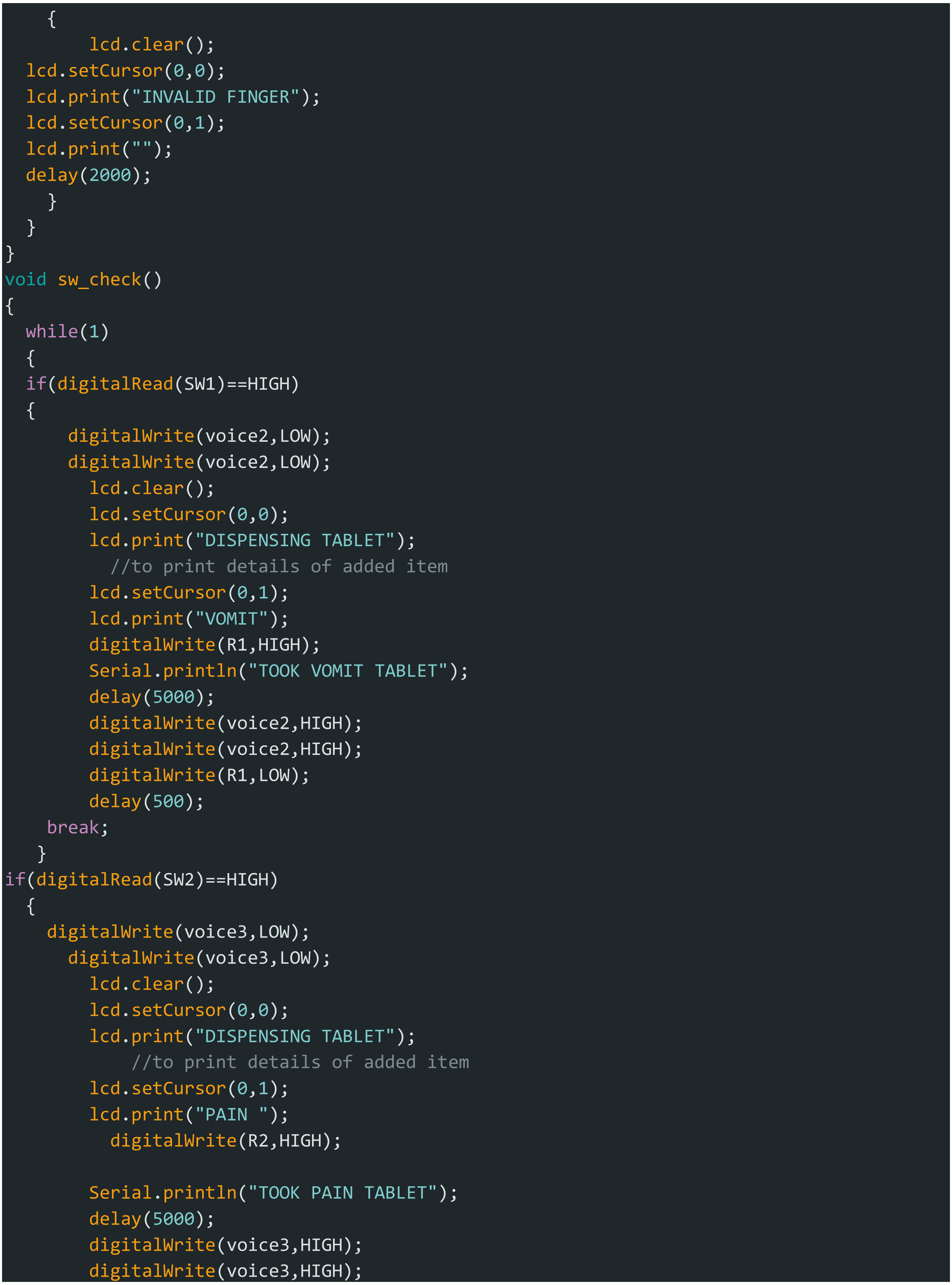
```
#include <Adafruit_Fingerprint.h>
#include <Wire.h>
#include <stdio.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>
const int rs = 8, en = 9, d4 = 10, d5 = 11, d6 = 12, d7 = 13;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
#define mySerial Serial1
// Data wire is plugged into digital pin 2 on the Arduino
#define ONE_WIRE_BUS A1
// Setup a oneWire instance to communicate with any OneWire device
OneWire oneWire(ONE_WIRE_BUS);
// Pass oneWire reference to DallasTemperature library
DallasTemperature sensors(&oneWire);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
char button = 'x';
int temp = 0, i = 0;
char str[15];
//int st1,st2,st3,st4;
const int SW1=2;
const int SW2=3;
const int SW3=4;

const int mode=34;

const int voice1=22;
const int voice2=24;
const int voice3=26;
const int voice4=28;
const int voice5=30;
const int voice6=32;
const int buz=A2;

const int fsr=A0;


const int R1=44;
const int R2=46;
const int R3=48;

const int R4=50;
const int R5=52;

const int R6=7;

int fsrValue = 0;
void fsr_part(void);
void body_temp(void);
void setup()
{
 pinMode(SW1,INPUT);
pinMode(SW2,INPUT);
pinMode(SW3,INPUT);

pinMode(mode,INPUT);

pinMode(R6,OUTPUT);

pinMode(voice1,OUTPUT);
pinMode(voice2,OUTPUT);
pinMode(voice3,OUTPUT);
pinMode(voice4,OUTPUT);
pinMode(voice5,OUTPUT);
pinMode(voice6,OUTPUT);

pinMode(R1,OUTPUT);
pinMode(R2,OUTPUT);
pinMode(R3,OUTPUT);
pinMode(R4,OUTPUT);
pinMode(R5,OUTPUT);

pinMode(buz,OUTPUT);

digitalWrite(buz,LOW);

digitalWrite(voice1,HIGH);
digitalWrite(voice2,HIGH);
digitalWrite(voice3,HIGH);
digitalWrite(voice4,HIGH);
digitalWrite(voice3,HIGH);
digitalWrite(voice4,HIGH);
digitalWrite(voice3,HIGH);
digitalWrite(voice4,HIGH);

digitalWrite(voice1,HIGH);
digitalWrite(voice2,HIGH);
digitalWrite(voice3,HIGH);
digitalWrite(voice4,HIGH);
digitalWrite(voice3,HIGH);
digitalWrite(voice4,HIGH);


digitalWrite(R1,LOW);
digitalWrite(R2,LOW);
digitalWrite(R3,LOW);
digitalWrite(R4,LOW);
digitalWrite(R5,LOW);
digitalWrite(R6,LOW);
  Serial.begin(9600);
  while (!Serial);  // For Yun/Leo/Micro/Zero/...
  delay(100);
  // set the data rate for the sensor serial port
  finger.begin(57600);
   lcd.begin(16,2);
  lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("VENDING MACHINE");
      lcd.setCursor(0,1);
      lcd.print("FOR MEDIKIT");
      delay(3000);
digitalWrite(R6,HIGH);
  lcd.clear();
  if (finger.verifyPassword()) {
  //  Serial.println("Found fingerprint sensor!");
  } else {
  //  Serial.println("Did not find fingerprint sensor :(");
    while (1) { delay(1); }
  }
  finger.getTemplateCount();
}
void loop()                     // run over and over again
{
if(digitalRead(mode)==LOW)
{
   lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("AUTOMATIC MODE");
  lcd.setCursor(0,1);
```



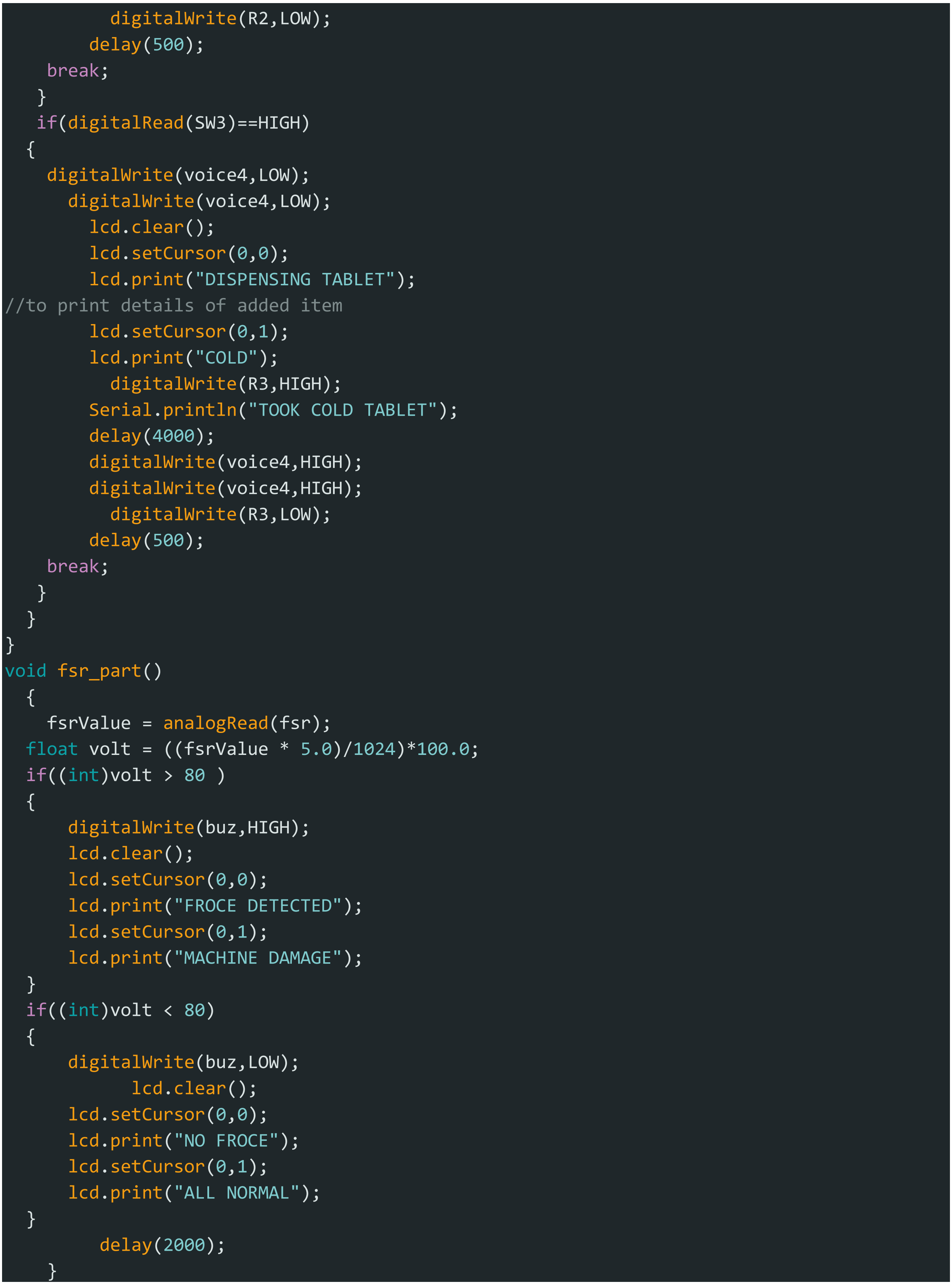
```
  lcd.print("");
  delay(2000);
  fsr_part();
   body_temp();
  }
  if(digitalRead(mode)==HIGH)
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("MANUAL MODE");
  lcd.setCursor(0,1);
  lcd.print("");
  delay(2000);
  finger_check();
  }
}
void finger_check(void)
{
 while(1)
 {
  fsr_part();
 int teacher;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("PLSEASE PUT ");
    lcd.setCursor(0,1);
    lcd.print("FINGER TO ACCESS      ");
    delay(1000);
    teacher = getFingerprintIDez();
    if( teacher == 1 )
    {
       lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("FINGER MATCHED");
  lcd.setCursor(0,1);
  lcd.print("");
  delay(2000);
       digitalWrite(voice1,LOW);
      digitalWrite(voice1,LOW);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("VALID PERSON");
              //to print details of added item
        lcd.setCursor(0,1);
        lcd.print("NAME: AAA");
        Serial.println("VAILD PERSON NAME:AAA");
        delay(1500);
        digitalWrite(voice1,HIGH);
        digitalWrite(voice1,HIGH);
```



```
      delay(500);
      sw_check();
  }
  else if( teacher == 2 )
  {
    lcd.clear();
lcd.setCursor(0,0);
lcd.print("FINGER MATCHED");
lcd.setCursor(0,1);
lcd.print("");
delay(2000);
     digitalWrite(voice1,LOW);
    digitalWrite(voice1,LOW);
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("VALID PERSON");
             //to print details of added item
      lcd.setCursor(0,1);
      lcd.print("NAME: BBB");
      Serial.println("VAILD PERSON NAME:BBB");
      delay(1500);
      digitalWrite(voice1,HIGH);
      digitalWrite(voice1,HIGH);
      delay(500);
      sw_check();
  }
 else if( teacher == 3 )
  {
      lcd.clear();
lcd.setCursor(0,0);
lcd.print("FINGER MATCHED");
lcd.setCursor(0,1);
lcd.print("");
delay(2000);
     digitalWrite(voice1,LOW);
    digitalWrite(voice1,LOW);
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("VALID PERSON");
           //to print details of added item
      lcd.setCursor(0,1);
      lcd.print("NAME: CCC");
      Serial.println("VAILD PERSON NAME:CCC");
      delay(1500);
      digitalWrite(voice1,HIGH);
      digitalWrite(voice1,HIGH);
      delay(500);
      sw_check();
  } else if( teacher == 4 )
```

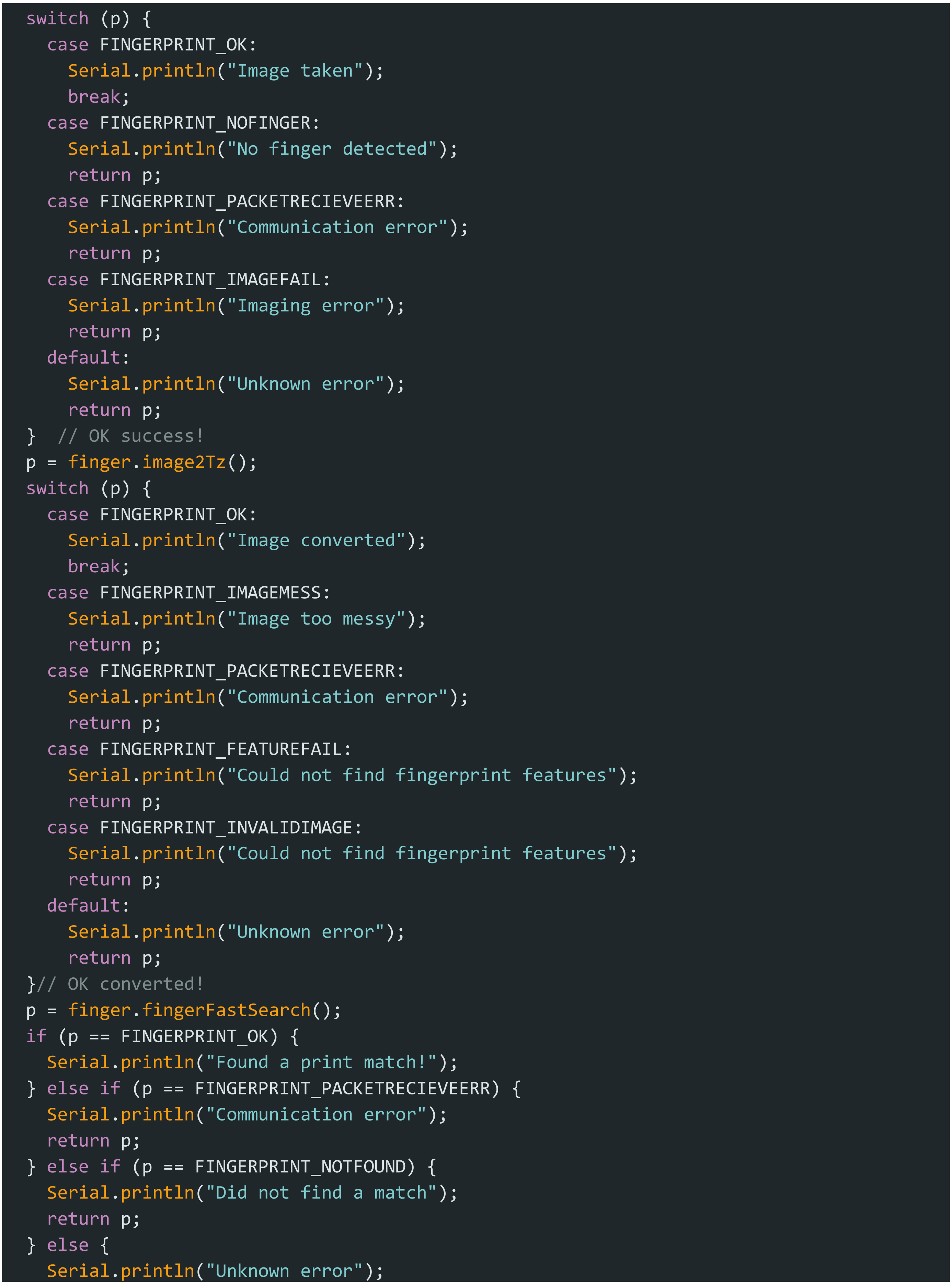


```
    {
        lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("INVALID FINGER");
  lcd.setCursor(0,1);
  lcd.print("");
  delay(2000);
    }
  }
}
void sw_check()
{
  while(1)
  {
  if(digitalRead(SW1)==HIGH)
  {
      digitalWrite(voice2,LOW);
      digitalWrite(voice2,LOW);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("DISPENSING TABLET");
          //to print details of added item
        lcd.setCursor(0,1);
        lcd.print("VOMIT");
        digitalWrite(R1,HIGH);
        Serial.println("TOOK VOMIT TABLET");
        delay(5000);
        digitalWrite(voice2,HIGH);
        digitalWrite(voice2,HIGH);
        digitalWrite(R1,LOW);
        delay(500);
    break;
   }
if(digitalRead(SW2)==HIGH)
  {
    digitalWrite(voice3,LOW);
      digitalWrite(voice3,LOW);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("DISPENSING TABLET");
            //to print details of added item
        lcd.setCursor(0,1);
        lcd.print("PAIN ");
          digitalWrite(R2,HIGH);

        Serial.println("TOOK PAIN TABLET");
        delay(5000);
        digitalWrite(voice3,HIGH);
        digitalWrite(voice3,HIGH);
```



```
          digitalWrite(R2,LOW);
        delay(500);
    break;
   }
   if(digitalRead(SW3)==HIGH)
  {
    digitalWrite(voice4,LOW);
      digitalWrite(voice4,LOW);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("DISPENSING TABLET");
//to print details of added item
        lcd.setCursor(0,1);
        lcd.print("COLD");
          digitalWrite(R3,HIGH);
        Serial.println("TOOK COLD TABLET");
        delay(4000);
        digitalWrite(voice4,HIGH);
        digitalWrite(voice4,HIGH);
          digitalWrite(R3,LOW);
        delay(500);
    break;
   }
  }
}
void fsr_part()
  {
    fsrValue = analogRead(fsr);
  float volt = ((fsrValue * 5.0)/1024)*100.0;
  if((int)volt > 80 )
  {
      digitalWrite(buz,HIGH);
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("FROCE DETECTED");
      lcd.setCursor(0,1);
      lcd.print("MACHINE DAMAGE");
  }
  if((int)volt < 80)
  {
      digitalWrite(buz,LOW);
            lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("NO FROCE");
      lcd.setCursor(0,1);
      lcd.print("ALL NORMAL");
  }
         delay(2000);
    }
```

```
void body_temp()
{
   // Send the command to get temperatures
  sensors.requestTemperatures();
//print the temperature in Celsius
  Serial.print("Temperature: ");
  Serial.println(sensors.getTempCByIndex(0));
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("TEMPEARTURE         ");
  lcd.setCursor(0,1);
  lcd.print(sensors.getTempCByIndex(0));
    delay(1500);
 if(sensors.getTempCByIndex(0)>38)
 {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("TEMPEARTURE         ");
  lcd.setCursor(0,1);
  lcd.print("MORE...    ");
  Serial.println(" TEMPRATURE IS MORE ");
  digitalWrite(voice1,LOW);
      digitalWrite(voice1,LOW);
        lcd.clear();
        lcd.setCursor(0,0);
         lcd.print("VALID PERSON");                                                      //to
print details of added item
        lcd.setCursor(0,1);
        lcd.print("NAME: DDD");
        Serial.println("VAILD PERSON NAME:DDD");
        delay(1500);
        digitalWrite(R4,HIGH);
        delay(5000);
        digitalWrite(R4,LOW);
        Serial.println("TOOK FEVER TABLET");
        delay(100);
        while(1);
 } else {
 lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("TEMPEARTURE         ");
  lcd.setCursor(0,1);
  lcd.print("NORMAL...    ");
  Serial.println("TEMPERATURE IS NORMAL");
  delay(1500);
 }
}
uint8_t getFingerprintID() {
  uint8_t p = finger.getImage();
```



```
  switch (p) {
    case FINGERPRINT_OK:
      Serial.println("Image taken");
      break;
    case FINGERPRINT_NOFINGER:
      Serial.println("No finger detected");
      return p;
    case FINGERPRINT_PACKETRECIEVEERR:
      Serial.println("Communication error");
      return p;
    case FINGERPRINT_IMAGEFAIL:
      Serial.println("Imaging error");
      return p;
    default:
      Serial.println("Unknown error");
      return p;
  }  // OK success!
  p = finger.image2Tz();
  switch (p) {
    case FINGERPRINT_OK:
      Serial.println("Image converted");
      break;
    case FINGERPRINT_IMAGEMESS:
      Serial.println("Image too messy");
      return p;
    case FINGERPRINT_PACKETRECIEVEERR:
      Serial.println("Communication error");
      return p;
    case FINGERPRINT_FEATUREFAIL:
      Serial.println("Could not find fingerprint features");
      return p;
    case FINGERPRINT_INVALIDIMAGE:
      Serial.println("Could not find fingerprint features");
      return p;
    default:
      Serial.println("Unknown error");
      return p;
  }// OK converted!
  p = finger.fingerFastSearch();
  if (p == FINGERPRINT_OK) {
    Serial.println("Found a print match!");
  } else if (p == FINGERPRINT_PACKETRECIEVEERR) {
    Serial.println("Communication error");
    return p;
  } else if (p == FINGERPRINT_NOTFOUND) {
    Serial.println("Did not find a match");
    return p;
  } else {
    Serial.println("Unknown error");
    return p;
  }   // found a match!
  Serial.print("Found ID #"); Serial.print(finger.fingerID);
  Serial.print(" with confidence of "); Serial.println(finger.confidence);
  return finger.fingerID;
}// returns -1 if failed, otherwise returns ID #
int getFingerprintIDez() {
  uint8_t p = finger.getImage();
  if (p != FINGERPRINT_OK)  return -1;

  p = finger.image2Tz();
  if (p != FINGERPRINT_OK)  return -1;

  p = finger.fingerFastSearch();
  if (p != FINGERPRINT_OK)  return -1;
  // found a match!
  return finger.fingerID;
}
```

CHAPTER 4

RESULTS

4.1: RESULTS

The project integrates various hardware components and technologies to deliver a secure, user-friendly, and efficient system. Here's an overview of the results achieved from implementing this project:

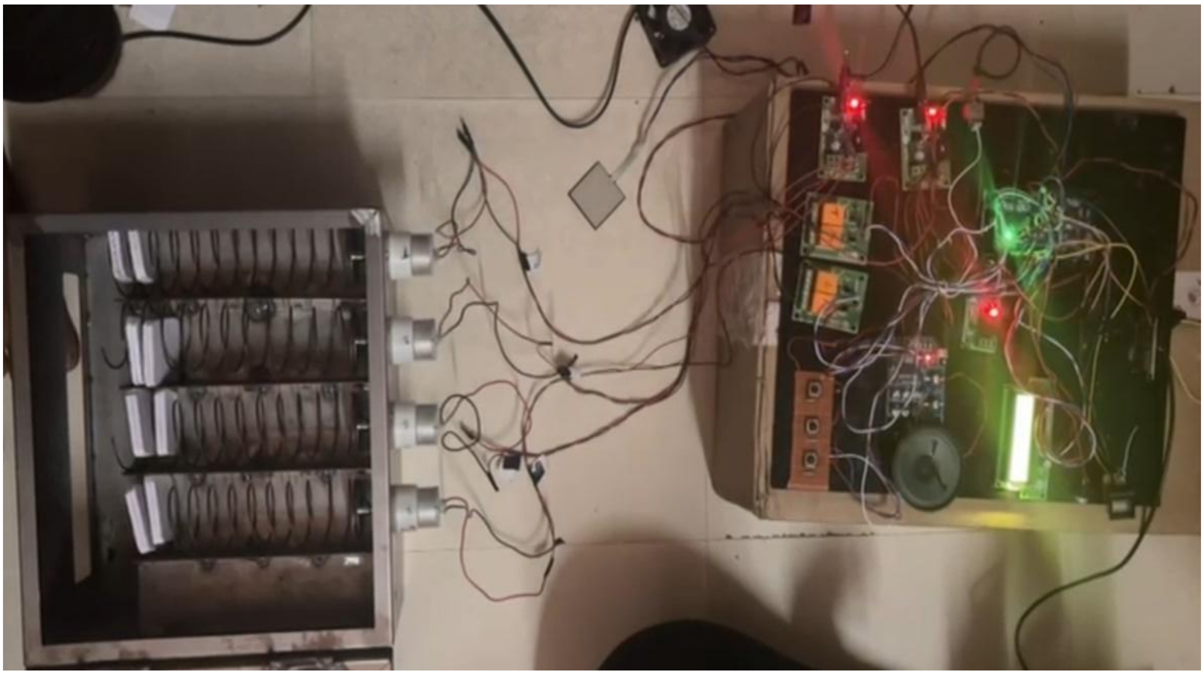


Fig. 4.1: Top view of the model

The figure 4.1 shows the top view of the model in which fingerprint sensor, FSR sensor, temperature sensor and aPR33A3 V2 module are mounted on the system and connected to power supply.

User Authentication:

- The user is authenticated via a fingerprint.
- The system verifies the user against stored data to ensure authorized access.

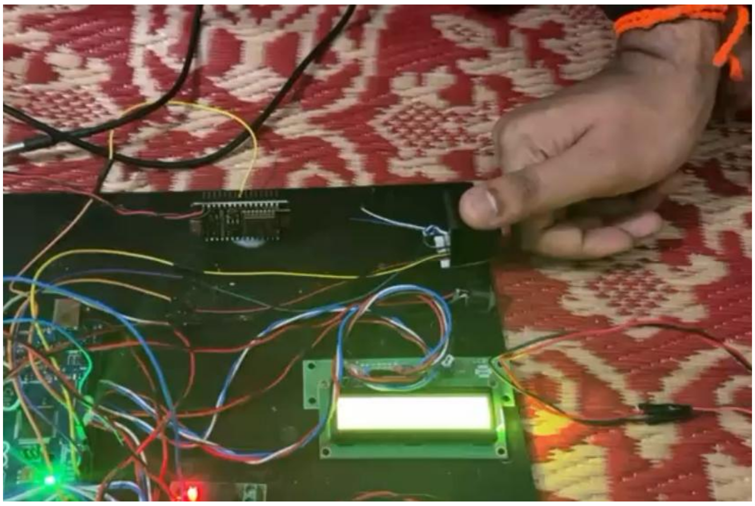


Fig 4.2: Model sensing the fingerprint of the user

Fig 4.2 shows the working of fingerprint sensor when a user tries to give his bio-metrics for authentication purpose.

Manual Mode:

- After authentication, the user interacts with the system (e.g., buttons, touch screens) to manually select their required medicine.
- The system confirms the selection, verifies dosage and timing, and dispenses the medicine.

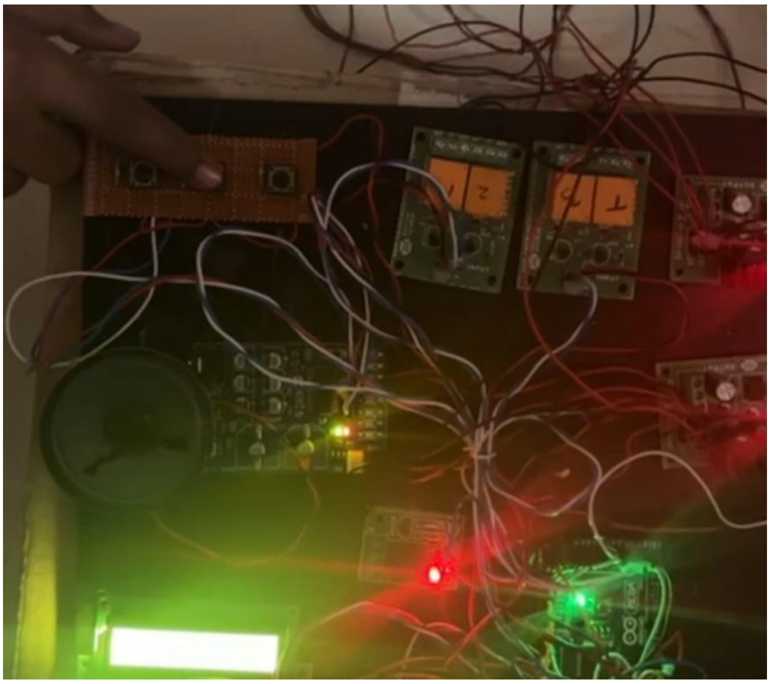


Fig 4.3: Model of user selecting the tablet by pressing buttons

Fig 4.3 shows the working of the system when a user selects the tablet, the buttons are connected to system via switch and helps in dispensing medicine through relays.

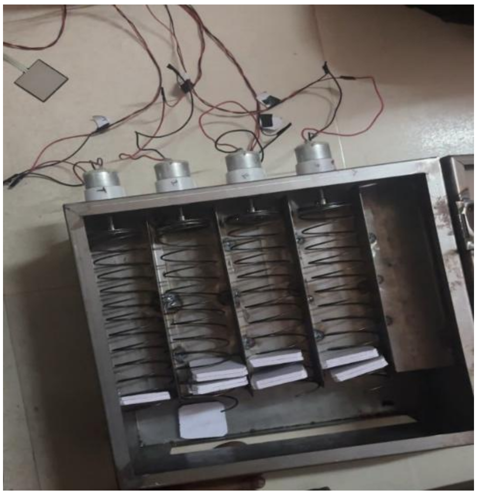


Fig. 4.4: A Proprietary Vending Machine while dispensing

Fig 4.4 shows the working vending machine model when tablet gets dispensed via motors as per the user’s requirement

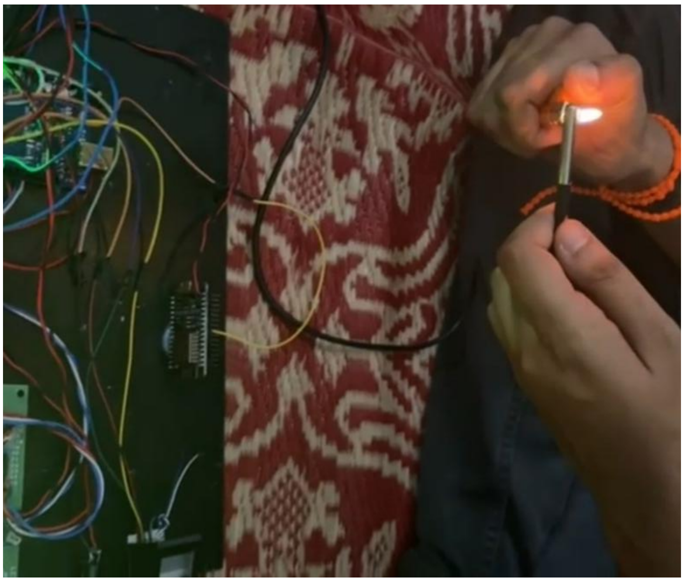
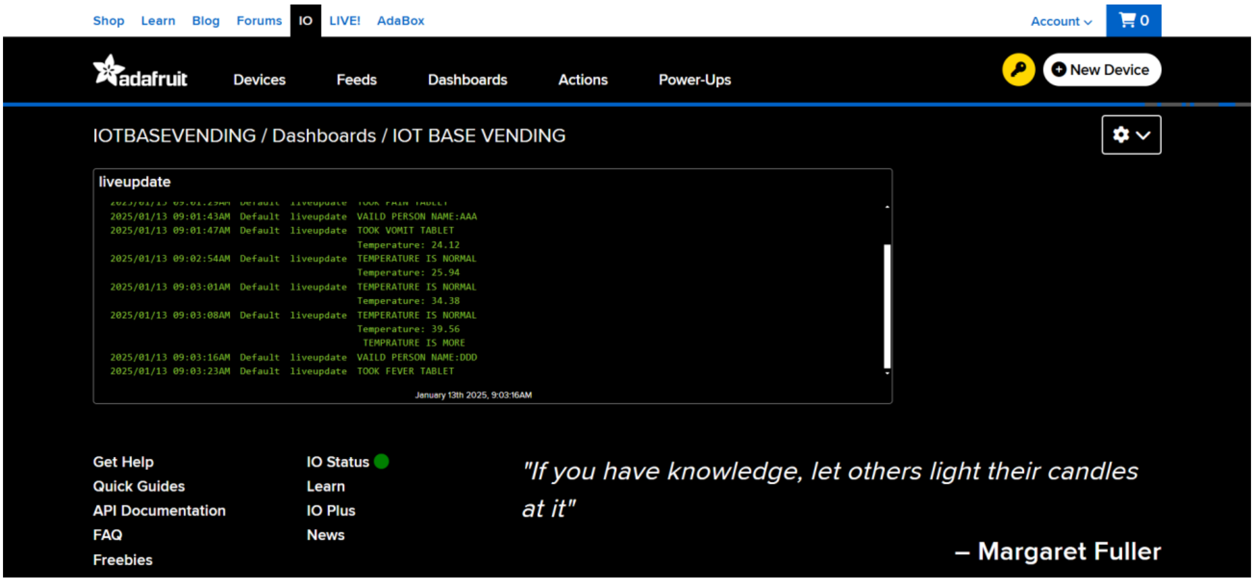


Fig. 4.5: Model of system sensing the temperature

Fig 4.5 shows the model sensing the temperature when temperature sensor is kept in front of flame for indicating high temperature.

Automatic Mode:

- After authentication, the system automatically checks a pre-programmed schedule for the user.
- If it’s time for a dose, the system dispenses the appropriate medicine without additional input from the user.



IOTBASEVENDING / Dashboards / IOT BASE VENDING

liveupdate

```
2025/01/13 09:01:29AM  Default  liveupdate  TOOK PAIN TABLET
2025/01/13 09:01:43AM  Default  liveupdate  VAILD PERSON NAME:AAA
2025/01/13 09:01:47AM  Default  liveupdate  TOOK VOMIT TABLET
                                            Temperature: 24.12
2025/01/13 09:02:54AM  Default  liveupdate  TEMPERATURE IS NORMAL
                                            Temperature: 25.94
2025/01/13 09:03:01AM  Default  liveupdate  TEMPERATURE IS NORMAL
                                            Temperature: 34.38
2025/01/13 09:03:08AM  Default  liveupdate  TEMPERATURE IS NORMAL
                                            Temperature: 39.56
                                             TEMPRATURE IS MORE
2025/01/13 09:03:16AM  Default  liveupdate  VAILD PERSON NAME:DDD
2025/01/13 09:03:23AM  Default  liveupdate  TOOK FEVER TABLET
```

January 13th 2025, 9:03:16AM

Fig. 4.6: Monitoring the dispensing of tablets for users

Fig 4.6 shows the monitoring of tablets dispensed by verifying the user via fingerprint authentication and using Adafruit IO website which is an open source application for monitoring of IoT based projects.

4.2: APPLICATIONS

1. Healthcare Facilities and Pharmacies

- **24/7 Accessibility**: A medicine vending machine can be installed in hospitals, clinics, or pharmacies, providing **24/7 access** to essential medicines for patients and healthcare professionals.
- **Emergency Use**: In emergency situations, especially after hours when pharmacies are closed, the machine can dispense critical medicines such as painkillers, first-aid kits, or emergency prescriptions.
- **Medication Refill**: Patients can use the machine to refill prescribed medications, improving patient compliance and reducing dependency on physical pharmacy visits.

2. Remote or Rural Locations

- **Access to Healthcare in Remote Areas**: In areas with limited access to medical facilities, a medicine vending machine can serve as a **lifeline** for people who need medications urgently. It can be placed in rural communities, isolated regions, or small villages where pharmacies or hospitals are not easily accessible.
- **Self-Diagnosis Assistance**: Using the **Voice Assistant** functionality, users can get **voice-guided assistance** about common ailments and the medications available in the machine, helping those without immediate medical guidance.

3. Hospitals and Clinics

- **Patient Self-Service**: Hospitals can implement a medicine vending machine for outpatient departments, where patients can directly purchase over-the-counter (OTC) medications, first-aid supplies, and minor prescription drugs without waiting in long queues.
- **Controlled Dispensing**: For **controlled or prescription-based drugs**, the system can be integrated with a patient’s medical record (via **fingerprint authentication**) to ensure that the correct medication is dispensed only to authorized individuals.

- **Medicine Tracking**: Using **remote connectivity (via NodeMCU ESP8266)**, administrators can remotely monitor medicine usage, ensure stock levels are maintained, and prevent potential theft or misuse.

4. Airports, Train Stations, and Public Areas

- **Instant Access to OTC Medications**: In busy transit hubs like airports or train stations, where travelers may suddenly need over-the-counter medicines (e.g., pain relievers, antihistamines, cold remedies), the vending machine offers immediate access.
- **Convenience for Tourists**: For travelers from other countries, the machine can be programmed to offer medication in various languages via **Voice Assistance**, and include widely available global medications, reducing the need to visit local pharmacies.
- **First-Aid Kits**: The vending machine can dispense **basic first-aid items** (e.g., band-aids, antiseptic, painkillers) during emergency situations in public spaces.

5. Corporate Offices and Workplaces

- **Workplace Health and Safety**: Companies can place medicine vending machines in their offices or production areas, giving employees quick access to common medications like painkillers, bandages, or cold and flu medicine.
- **Employee Wellness**: The machine can be part of an overall employee wellness program, offering employees immediate access to over-the-counter medicines in a convenient manner.
- **Employee Authentication**: With **fingerprint authentication**, the system can ensure that only registered employees can access specific items (for example, medicines for corporate health plans), preventing misuse.

6. Universities and Educational Institutions

- **Student Health Management**: In large universities or college campuses, the vending machine can provide students with easy access to **OTC medicines**, vitamins, or emergency supplies. The **fingerprint authentication** ensures that the medications are dispensed only to eligible individuals.
- **Study-Related Health Issues**: During exam periods or stressful times, students can access medicine for common conditions like stress, headaches, or lack of sleep, without needing to leave campus.
- **Health Education**: The **Voice Assistant** feature can educate students about **common illnesses**, preventive measures, and first-aid practices, contributing to campus health education.

7. Disaster Relief Operations

- **Post-Disaster Medicine Distribution**: During natural disasters (earthquakes, floods, etc.), medicine vending machines can be deployed in affected areas to distribute critical medicines, including pain relievers, antibiotics, and first-aid supplies.
- **Quick Distribution in Crisis**: In places with limited healthcare infrastructure during emergencies, these machines can provide **quick and automated** medicine distribution without waiting for medical personnel.

8. Fitness Centers and Gyms

- **Supplement and Medicine Vending**: For fitness-conscious individuals, the machine can offer **vitamins**, **supplements**, and **OTC medications** such as pain relief or energy boosters. This adds value to gym facilities by offering easy access to health-related products.
- **Convenience**: After a workout session, members can quickly grab painkillers, anti-inflammatory medications, or muscle relief ointments.

9. Home Healthcare

- **Medications for Elderly or Disabled**: In home care settings, where individuals may have difficulty traveling to the pharmacy, the **medicine vending machine** can provide **self-service** access to medications.
- **Remote Monitoring for Caregivers**: Caregivers can monitor the machine's status and medication usage remotely, ensuring that medications are dispensed correctly and consistently, especially for patients with chronic illnesses.

10. Medical Research and Clinical Trials

- **Controlled Drug Distribution**: In clinical trials, a vending machine could be used to dispense trial medications in a **controlled** and **automated** manner, ensuring accurate record-keeping and preventing errors.
- **Patient Tracking**: **Fingerprint authentication** can ensure that only participants in clinical trials receive specific medications, reducing the risk of human error.

11. Smart Hospitals

- **Integration with Smart Hospital Systems**: In a **smart hospital**, a medicine vending machine can be integrated with the hospital's **electronic health records (EHR)** system, allowing patients to authenticate and access their prescribed medications directly from the machine.
- **Inventory Management**: The machine can be connected to the hospital's **inventory management system** to automatically alert staff when stock is low and trigger restocking requests.

12. Insurance Companies and Telemedicine

- **Telemedicine Access**: In conjunction with **telemedicine services**, the vending machine can dispense medications that are prescribed by **remote doctors** during virtual consultations, offering an integrated healthcare experience.
- **Insurance Integration**: The machine can also integrate with **health insurance programs**, allowing users to access medications covered under their policy via a secure, authenticated process.

4.3: ADVANTAGES

1. Enhanced Security

- **Fingerprint Authentication**: Using **biometric authentication** provides a high level of security for users, ensuring that only authorized individuals can access the machine and make transactions. This is especially useful in scenarios where restricted or premium products are involved.
- **Personalized Access**: Only registered users can access specific products or services, enhancing control over the vending process.

2. Convenient User Interaction

- **Voice Assistance**: The **aPR33A3 V2 Voice Assistant** provides a hands-free interaction with the vending machine. Users can inquire about available products, request assistance, and complete transactions, all through voice commands.
- **Ease of Use**: The combination of **fingerprint authentication** and **voice commands** makes the machine user-friendly, particularly for those who prefer voice over manual input.

3. Remote Monitoring and Control

- **NodeMCU ESP8266 Wi-Fi Connectivity**: The integration of **Wi-Fi** allows real-time monitoring and control of the vending machine. Operators can remotely check inventory levels, manage machine health, and even perform software updates without being physically present.
- **Inventory Management**: Machine operators can monitor the stock, manage refills, and receive alerts if a product is running low or if the machine experiences a malfunction.

4. Energy Efficiency

- **Automatic Cooling System**: The **fan** connected to a **single-channel relay** provides efficient cooling, ensuring that the machine operates in optimal conditions. The fan is only activated when the temperature exceeds a certain threshold, saving energy.
- **Low Power Consumption**: Components like **Arduino MEGA 2560**, **NodeMCU ESP8266**, and **aPR33A3 V2** are designed for low power usage, ensuring the machine remains energy-efficient.

5. Customization and Scalability

- **Modular Design**: The project is scalable, and additional features can be easily added, such as different user authentication methods (e.g., face recognition, PIN code), a wider range of products, or additional voice commands.
- **Multi-Functionality**: The system can serve a variety of purposes beyond vending, such as in **secure product lockers**, **medicine dispensers**, or **automated ticket dispensers**.

6. Improved Customer Experience

- **Personalized Interaction**: The voice assistant can provide personalized greetings and instructions, improving the overall user experience.
- **Fast and Convenient Transactions**: The integration of **fingerprint authentication** ensures quick and secure transactions, eliminating the need for cash or cards.

7. Cost-Effective and Simple Design

- The components used in the project are affordable and readily available, making it a cost-effective solution for creating an automated vending machine system.

- The design leverages simple **Arduino-based microcontrollers** and **low-cost components** like the **Adafruit Optical Fingerprint Scanner**, **NodeMCU ESP8266**, and **single-channel relay**, ensuring that the overall cost remains manageable.

4.4: DISADVANTAGES

1. Limited Product Variety

- **Physical Size Constraints**: While the system is capable of dispensing a range of products, the physical design of the machine may limit the types or sizes of products it can hold.
- **Manual Refilling**: Depending on the design, the machine might require frequent manual refilling and maintenance, especially if it is used for a variety of products.

2. Security Risks with Fingerprint Data

- **Privacy Concerns**: Storing fingerprint data, even though biometric authentication provides security, raises privacy concerns. The fingerprint data must be securely stored and encrypted to prevent unauthorized access.
- **Data Breaches**: Any security vulnerability in the system could lead to the exposure of sensitive biometric data. Implementing strict security measures and adhering to data protection laws is essential.

3. Voice Recognition Limitations

- **Environmental Noise**: The voice assistant may struggle to accurately recognize commands in noisy environments or areas with poor acoustics, affecting its usability.
- **Limited Vocabulary**: While the voice assistant can handle basic commands, expanding the vocabulary and adding complex functionalities may require additional programming and resources.
- **Language Barriers**: If the system is not multilingual, it could limit its user base in diverse regions.

4. Hardware Limitations

- **Relay Overload**: The **single-channel relay** can only control one device (e.g., fan), which may limit scalability if additional devices need to be controlled simultaneously. Multiple relays may be required for more complex operations, increasing system complexity.

- **Overheating or Fan Failure**: The **fan** used for cooling may fail over time due to mechanical wear and tear or overheating, leading to potential system malfunctions. Regular maintenance would be required to ensure the cooling system works efficiently.

5. Limited Cooling System

- **Basic Cooling Mechanism**: The **fan-based cooling system** may not be sufficient for large vending machines or those located in areas with high ambient temperatures. For such environments, more advanced **cooling systems** (e.g., air conditioning units) may be necessary, which could increase both cost and complexity.
- **Power Draw of Fan**: Although the system is energy-efficient, using a fan continuously or under heavy load may cause the system to consume more power than anticipated, leading to higher operational costs.

6. Dependence on Internet Connection

- **Wi-Fi Dependency**: The **NodeMCU ESP8266 Wi-Fi board** relies on a stable internet connection for remote monitoring and control. If the connection is lost, it may hinder the operation of remote management features, affecting overall system performance.
- **Network Security Risks**: If not properly secured, the machine's **Wi-Fi connectivity** could become a target for hacking, allowing unauthorized access to the system.

7. Maintenance and Debugging

- **System Maintenance**: The integration of multiple technologies such as **fingerprint scanners**, **voice assistants**, and **Wi-Fi connectivity** increases the need for ongoing maintenance and troubleshooting. Components like **fingerprint scanners** and **fans** may require periodic servicing or replacement.
- **Complexity in Debugging**: The interaction between multiple components (e.g., sensors, motors, Wi-Fi, fingerprint scanners, and the voice assistant) may complicate debugging, requiring a deeper technical knowledge to identify and resolve issues.

8. Cost of Advanced Components

- While the project components are generally cost-effective, adding advanced features like **high-end fingerprint scanners**, **high-torque servos**, or a **high-efficiency cooling system** could increase the cost, especially if scaling to large quantities of machines.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1: CONCLUSION

The Semi-Automatic Vending Machine demonstrates a successful combination of hardware components and software integration. By utilizing fingerprint authentication, IoT connectivity, and voice assistance, the system offers enhanced security, improved user experience, and efficient vending operations. The integration of Arduino MEGA 2560, NodeMCU ESP8266, and the aPR33A3 V2 module ensures scalability and adaptability for various applications.

5.2: FUTURE SCOPE

The **Semi-Automatic Vending Machine** project using **Fingerprint Authentication** (Adafruit Optical Fingerprint Scanner), **Arduino MEGA 2560**, **NodeMCU ESP8266**, and **aPR33A3 V2 Voice Assistant** has demonstrated a solid foundation for secure and efficient vending operations. However, there are several areas for improvement and expansion to enhance the system's functionality, user experience, and scalability.

1. Enhanced User Authentication & Security

- **Face Recognition Integration**: Combine **face recognition technology** with fingerprint scanning to offer multi-factor authentication, making the vending machine even more secure.
- **Encrypted Data**: Implement data encryption techniques to ensure that user data (such as fingerprints) is securely stored and transmitted, especially when using remote cloud systems for storage.
- **Real-time Surveillance**: Integrate **CCTV cameras** with motion detection that triggers alerts in case of suspicious activities (e.g., tampering with the machine or vandalism).

2. Expanded Voice Assistant Features

- **Natural Language Processing (NLP)**: Improve the voice assistant by adding more advanced NLP capabilities to allow users to ask questions, inquire about product details, or request assistance (e.g., “What are the available items?” or “Can I get a discount?”).

- **Multilingual Support**: Enable the voice assistant to support multiple languages to cater to a wider user base in global markets.
- **Customizable Responses**: Allow users to program or personalize voice responses, making the system more adaptable for specific use cases or brands.

3. IoT and Cloud Integration

- **Remote Control and Monitoring**: Extend the functionality of the **NodeMCU ESP8266** to allow full remote control via a mobile app or web-based dashboard. This could include monitoring inventory, machine status (e.g., out of service, in stock), and even rebooting the system remotely.
- **Cloud Data Storage**: Store user data, transaction logs, and machine status in the **cloud** for data analysis, machine performance optimization, and predictive maintenance.
- **Usage Analytics**: Implement analytics to track user behavior and preferences, which can help optimize inventory management, pricing, and promotions.

4. Payment Integration

- **Contactless Payment Systems**: Add support for **RFID/NFC**, **QR codes**, or **mobile wallet payments (e.g., Google Pay, Apple Pay)**, allowing users to pay directly without needing cash or coins.
- **Prepaid Accounts**: Enable users to create **prepaid accounts** where their fingerprint can be linked to a balance that gets deducted after each transaction, adding a layer of convenience.

5. Machine Learning for Predictive Maintenance

- **Fault Detection**: Utilize **machine learning** to detect potential mechanical failures or faults before they happen by analyzing usage patterns and machine behavior (e.g., slow dispensing, sensor malfunctions).
- **Predictive Stocking**: Use AI to analyze consumption patterns and predict when the machine will need refilling, ensuring that the machine is always stocked with the right items and reducing downtime.

6. Advanced Hardware Features

- **More Advanced Sensors**: Integrate additional sensors such as **temperature** or **humidity sensors** to monitor the condition of the items inside the vending machine, ensuring their quality (particularly for perishable items like beverages or snacks).

- **Modular Design**: Make the vending machine design more **modular**, allowing easy swapping of components such as the fingerprint scanner, voice assistant, or payment interface based on user needs or technological advancements.

7. Sustainability and Energy Efficiency

- **Solar-Powered Vending Machine**: Integrate solar panels to make the vending machine more energy-efficient, especially for outdoor applications where electricity access may be limited.
- **Recyclable and Sustainable Materials**: Use eco-friendly materials for machine housing and packaging, and implement **recycling stations** for users to dispose of packaging or empty items.

8. Interaction with Other Smart Devices

- **Integration with Smart Home Devices**: Link the vending machine to a smart home system where users can operate it using voice assistants like **Amazon Alexa**, **Google Assistant**, or **Apple Siri**.
- **Integration with Wearables**: Allow users to authenticate and interact with the vending machine through wearables like **smartwatches** or **fitness bands**, making the process even more seamless and convenient.

9. Customization for Specific Use Cases

- **Corporate/Office Vending Machines**: Tailor the system for corporate offices, allowing employees to access the vending machine using their work credentials or integrate the system with office attendance records.
- **Medical Vending Machines**: Customize the machine for healthcare environments where medical supplies or medications are dispensed, and users authenticate with their medical ID or fingerprint for secure access.

10. Social Interaction and Gamification

- **Interactive Features**: Add **interactive screens** for entertainment or educational purposes while users wait, turning the vending machine into a more engaging experience.
- **Loyalty Programs**: Integrate loyalty programs or gamification features where users can earn rewards, discounts, or exclusive offers based on their interaction with the machine (e.g., frequent purchases or using the fingerprint authentication).

References:

1. Nilabh Niran, Dhrubajyoti Das, et. al., “*Design of Food, Medicine & Water Dispensing Automation Device*”, Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), IEEE Xplore, 10 November 2020, DOI: 10.1109/I-SMAC49090.2020.9243495
2. Praveen Kumar, Shailaja Singh, et. al., “*Solar Powered Medic Vending Machine*”, 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), IEEE Xplore, 01 March 2021, DOI: 10.1109/ICACCCN51052.2020.9362969
3. Anil Kumar D B, Doddabasappa N, et. al.. “*Embedded system Based Anytime Medicine Vending Machine*”, International Conference on Networking, Communications and Information Technology (NetCIT), IEEE Xplore, 11 March 2022, DOI: 10.1109/NetCIT54147.2021.00025
4. Xu Wang, Haotian Zhang, Zhihui Liu, Rui He, Zeyu Wang, et. al., “*Intelligent Medical System of Unmanned Pharmacy Based on Internet Plus*”, IEEE International Conference on Intelligent Systems, Smart and Green Technologies (ICISSGT), IEEE Xplore, 02 March 2022, DOI: 10.1109/ICISSGT52025.2021.00042
5. Divya Ganesh, Gayathri Seshadri, et. al., “*Automatic Health Machine for COVID-19 and Other Emergencies*”, International Conference on COMmunication Systems & NETworkS (COMSNETS) , IEEE Xplore, 17 February 2021, DOI: 10.1109/COMSNETS51098.2021.9352866
6. C. H. Patil, Nakia Lightwala, Mehul Sherdiwala, et. al., “*An IoT based Smart Medicine Dispenser Model for Healthcare*”, IEEE World Conference on Applied Intelligence and Computing (AIC), IEEE Xplore, 18 August 2022, DOI: 10.1109/AIC55036.2022.9848934
7. Abhinandan Raj, Anuj Kumar Sony, et. al., “*Highway-Situated Medicine Vending Machines: A Technological Solution for Immediate First-Aid Assistance in Remote Areas*”, 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), IEEE Xplore, 25 January 2024, DOI: 10.1109/ICTACS59847.2023.10389972
8. Anu Joy, Anitta D, Kottaimalai Ramaraj, Thilagaraj M, et. al., “*Smart Drug Delivering System in Hospitals Using RFID*”, 5th International Conference on Inventive Research in Computing Applications (ICIRCA), IEEE Xplore, 28 August 2023, DOI: 10.1109/ICIRCA57980.2023.10220582
9. Anu Joy, Pallavi N, Vandana R, Nishath Anjum A, Saniya Siddiqa, Anitta D, et. al., “*Medicine Vending Machine Using RFID*”, 7th International Conference on Design Innovation for 3 Cs Compute Communicate Control (ICDI3C), IEEE Xplore, 16 July 2024, DOI: 10.1109/ICDI3C61568.2023.00071