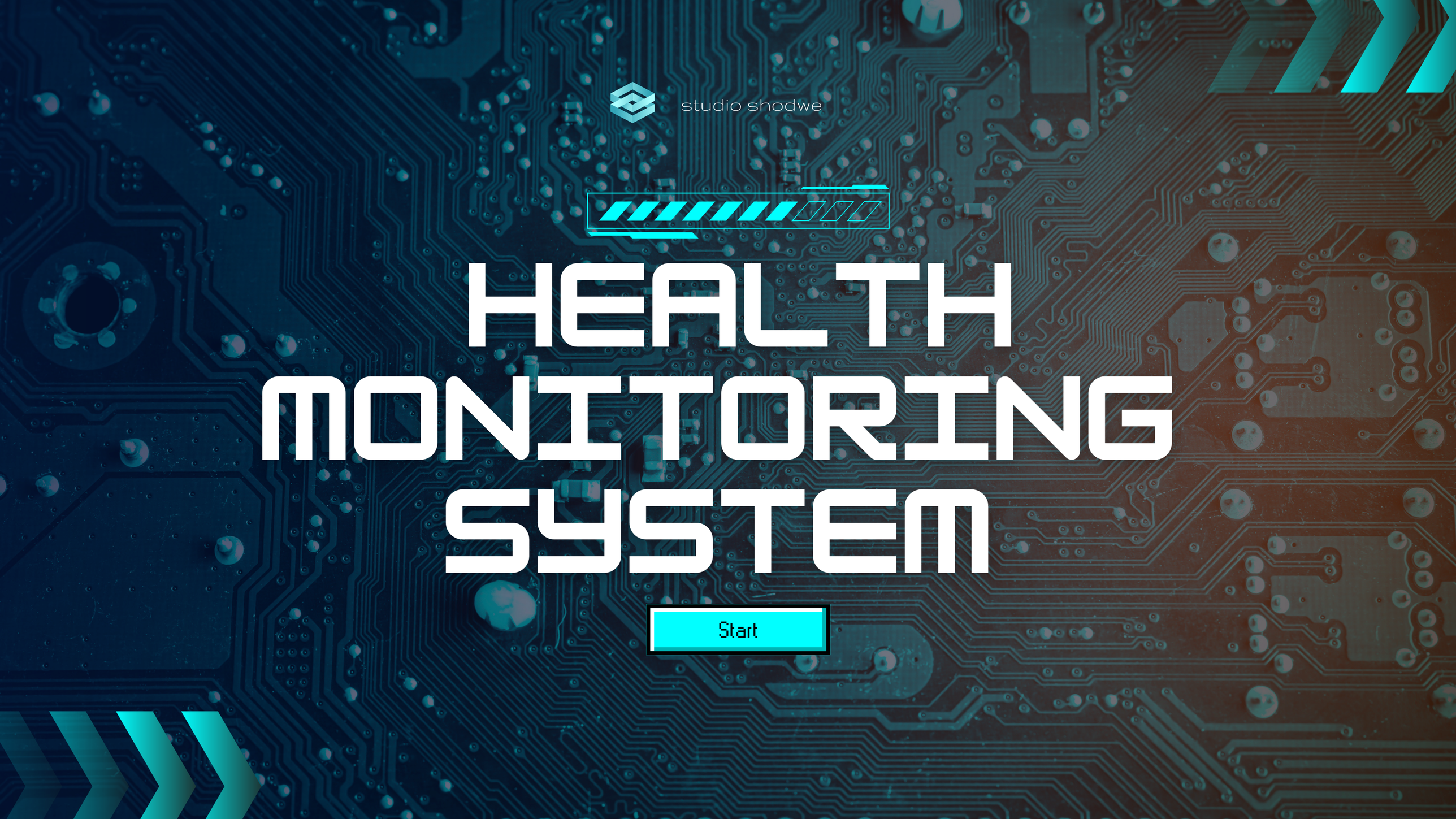

studio shodwe
HEALTH
MONITORING
SYSTEM
Start

studio shodwe

OUR TEAM

1 - Zeiad Elbalat
2 - Mazen fayed
3 - Adham Hani
4- Mahmoud Samy
5- Mostafa Hesham
6 - Aml Mohamed
7- Laila Hamada

1. PROJECT OVERVIEW

Our project is a Health Monitoring System that utilizes the ESP32 microcontroller to measure and monitor vital health parameters such as body temperature and heart rate (ECG signals). The system sends real-time data to the Blynk IoT platform, allowing users to monitor their health remotely via a smartphone. Additionally, the system includes an alert mechanism (buzzer) to notify users of critical conditions such as flatline (no heartbeat).

studio shodwe

2. COMPONENTS USED

- ESP32 Microcontroller (Main processing unit)
- MLX90614 Infrared Temperature Sensor (Non-contact body temperature measurement)
- ECG Sensor (Analog Input) (Heart rate monitoring)
- Buzzer (Alerts for abnormal conditions)
- Blynk IoT Platform (Remote monitoring and notifications)
- WiFi Module (Internet connectivity for data transmission)

studio shodwe

IMPLEMENTATION DETAILS

1. Real-Time Temperature Monitoring
   - The MLX90614 sensor measures body temperature in Celsius (°C) and sends data to Blynk.
2. Heart Rate Detection via ECG
   - The ECG sensor reads analog signals to detect heartbeats and calculates Beats Per Minute (BPM).
   - A threshold-based algorithm identifies pulse detection and flatline conditions.
3. Emergency Alerts
   - A buzzer activates if:
     - No heartbeat is detected for 3 seconds (flatline condition).
     - A pulse is detected (short beep for each heartbeat).
4. Cloud Integration (Blynk)
   - Real-time data is displayed on a Blynk dashboard with graphs for temperature, BPM, and ECG signals.

studio shodwe

OUR CODE👉

```
#define BLYNK_TEMPLATE_ID "TMPL266nrqJeN"
#define BLYNK_TEMPLATE_NAME "Health monitor"
#define BLYNK_AUTH_TOKEN "bsBLCHnLtBEeOMtzAgOLlZytdG8PLmQ4"

#include <WiFi.h>
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <BlynkSimpleEsp32.h>

char ssid[] = "Zeiad";
char pass[] = "zeezoo1234";

const int BUZZER_PIN = 25;
const int ECG_PIN = 36;

Adafruit_MLX90614 mlx = Adafruit_MLX90614();

bool pulseDetected = false;
bool flatline = false;
int threshold = 2200;
int bpm = 0;
unsigned long lastBeatTime = 0;

void setup() {
  Serial.begin(115200);
  pinMode(ECG_PIN, INPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  digitalWrite(BUZZER_PIN, LOW);
  Wire.begin(21, 22);
  mlx.begin();
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
```

ealthMonitor.ino

```
  long sum = 0;
  for (int i = 0; i < 100; i++) {
    sum += analogRead(ECG_PIN);
    delay(10);
  }
  threshold = sum / 100 + 100;
  Serial.println("Threshold: " + String(threshold));
}

void loop() {
  Blynk.run();

  float temperature = mlx.readObjectTempC();
  int ecg = analogRead(ECG_PIN);
  unsigned long now = millis();

  if (!pulseDetected && ecg > threshold) {
    pulseDetected = true;
    unsigned long interval = now - lastBeatTime;
    lastBeatTime = now;
    if (interval > 0 && interval < 2000) {
      bpm = 60000 / interval;
    }
    flatline = false;
    tone(BUZZER_PIN, 2000, 20);
  }

  if (pulseDetected && ecg < threshold - 50) {
    pulseDetected = false;
  }

```

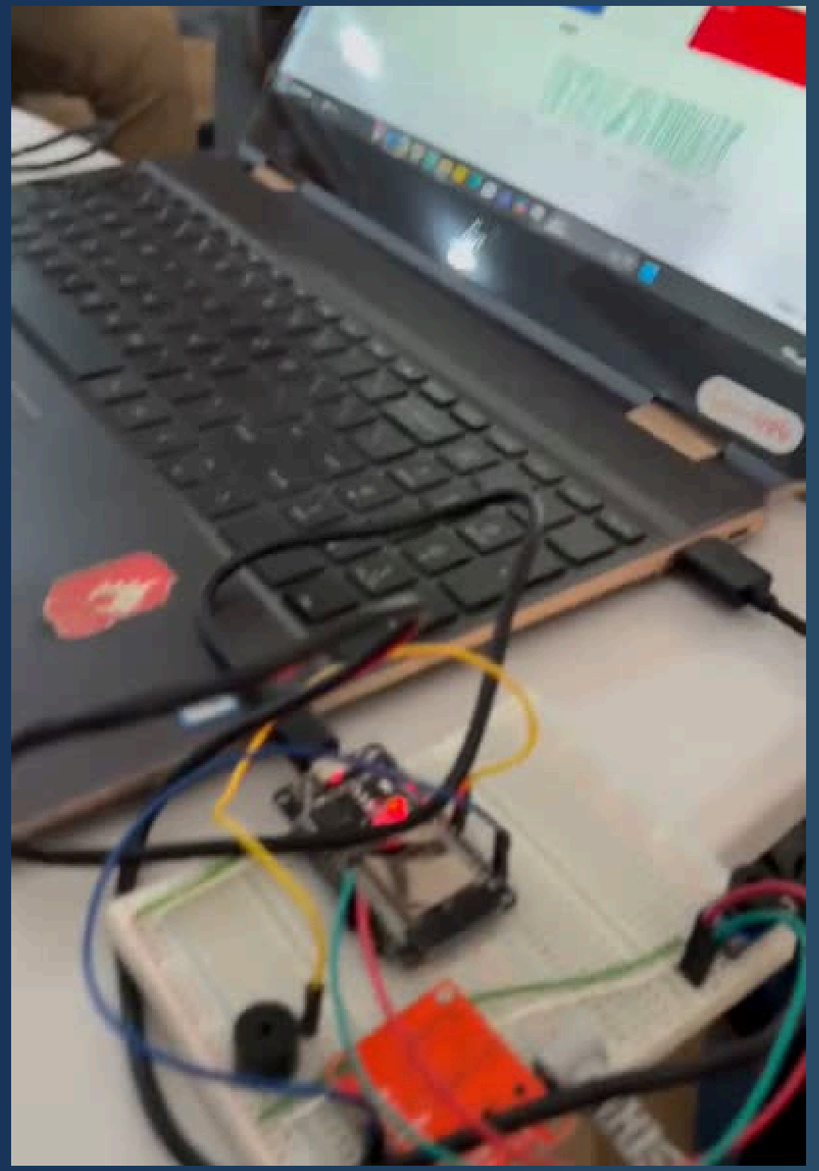
HealthMonitor.ino

```
47    unsigned long now = millis();
48
49    if (!pulseDetected && ecg > threshold) {
50      pulseDetected = true;
51      unsigned long interval = now - lastBeatTime;
52      lastBeatTime = now;
53      if (interval > 0 && interval < 2000) {
54        bpm = 60000 / interval;
55      }
56      flatline = false;
57      tone(BUZZER_PIN, 2000, 20);
58    }
59
60    if (pulseDetected && ecg < threshold - 50) {
61      pulseDetected = false;
62    }
63
64    if (now - lastBeatTime > 3000) {
65      bpm = 0;
66      flatline = true;
67      digitalWrite(BUZZER_PIN, HIGH);
68    } else if (!flatline) {
69      digitalWrite(BUZZER_PIN, LOW);
70    }
71
72    Blynk.virtualWrite(V0, temperature);
73    Blynk.virtualWrite(V1, bpm);
74    Blynk.virtualWrite(V2, ecg);
75
76    Serial.println("Temp: " + String(temperature) + "  BPM: " + String(bpm) + "  ECG: " + String(ecg));
77    delay(100);
78  }
```

studio shodwe

SIMPLE DOCUMENTATION VIDEO SCRIPT FOR HEALTH MONITORING SYSTEM

www.reallygreatsite.com



studio shodwe

CHALLENGES FACED:

1. ECG Signal Noise
   - Initial readings were unstable due to electrical noise.
   - Solution: Implemented threshold calibration during setup to improve accuracy.
2. WiFi Connectivity Issues
   - Sometimes, the ESP32 failed to connect to Blynk.
   - Solution: Added error handling and auto-reconnect in the code.
3. False Flatline Detection
   - The buzzer sometimes triggered incorrectly.
   - Solution: Adjusted the delay and threshold values to reduce false alarms.

studio shodwe

5. FUTURE IMPROVEMENTS (FUTURE WORK)

1. Integration with AI for Predictive Analysis
   - Implement machine learning (TensorFlow Lite) to predict potential health issues based on trends in temperature and heart rate.
2. Mobile App Notifications
   - Enhance Blynk with push notifications for critical alerts.
3. Multi-Patient Monitoring
   - Expand the system to support multiple sensors for hospital or family use.
4. Data Logging & Cloud Storage
   - Store historical data in Google Sheets or Firebase for long-term analysis.
5. Portable Wearable Design
   - Convert the prototype into a compact wearable device with a battery-powered design.


studio shodwe
THANK YOU!