Portable ECG Machine - Complete Code

Arduino Code (ECG Signal Processing)

```
#include <SoftwareSerial.h>
SoftwareSerial btSerial(10, 11); // RX, TX
int ecgPin = A0;

void setup() {
    Serial.begin(115200);
    btSerial.begin(9600);
}

void loop() {
    int ecgValue = analogRead(ecgPin);
    Serial.println(ecgValue);
    btSerial.println(ecgValue);
    delay(5);
}
```

Python Code (Data Processing & Visualization)

```
import serial
import matplotlib.pyplot as plt
import numpy as np

ser = serial.Serial('COM3', 115200)
data = []

plt.ion()
fig, ax = plt.subplots()

while True:
    try:
        val = int(ser.readline().decode().strip())
        data.append(val)
        if len(data) > 100:
            data.pop(0)
        ax.clear()
        ax.plot(data, label='ECG Signal')
        plt.legend()
        plt.pause(0.01)
    except KeyboardInterrupt:
        break
ser.close()
```

Angular Code (Web Application)

```
/* Angular ECG Visualization */
```

```
import { Component, OnInit } from '@angular/core';
import { EcgDataService } from '../ecg-data.service';
import { Chart } from 'chart.js';

@Component({
  selector: 'app-ecg-visualizer',
  templateUrl: './ecg-visualizer.component.html',
  styleUrls: ['./ecg-visualizer.component.css'],
})
export class EcgVisualizerComponent implements OnInit {
  ecgData: number[] = [];
  chart: any;

  constructor(private ecgService: EcgDataService) {}

  ngOnInit(): void {
    this.ecgService.ecgData$.subscribe((data) => {
      this.ecgData = data;
      this.updateChart();
    });
  }

  updateChart() {
    if (!this.chart) {
      const ctx = document.getElementById('ecgChart') as HTMLCanvasElement;
      this.chart = new Chart(ctx, {
        type: 'line',
        data: {
          labels: Array.from({ length: this.ecgData.length }, (_, i) => i.toString()),
             datasets: [{ label: 'ECG Signal', data: this.ecgData, borderColor: 'blue',
fill: false }],
        },
        options: { responsive: true, animation: false },
      });
    } else {
      this.chart.data.datasets[0].data = this.ecgData;
      this.chart.update();
    }
  }
}
```