Graphical User Interface to acquire ECG using MATLAB and BITalino

Nathan Guerreiro[1,2][0009-0007-2879-2248], Luiz Ribeiro[1,2][0009-0002-2452-7264], Mohamed Khalil Chaabani[1,3][0009-0006-1813-6944], Luiz E. Luiz[1,2][0000-0001-9305-091X], Maria Eugenia Dajer[2][0000-0003-1648-3919], and João Paulo Teixeira[1][0000-0002-6679-5702]

[1] CeDRI, SusTEC, Instituto Politécnico de Bragança, 5300-253 Bragança, Portugal
{a61448, a58865, a60546}@alunos.ipb.pt
{luiz.luiz, joaopt}@ipb.pt
https://cedri.ipb.pt/
[2] Federal University of Technology - Paraná, Brazil
{nathanguerreiro, luizotavioribeiro, lluiz}@alunos.utfpr.edu.br
medajer@utfpr.edu.br
https://www.utfpr.edu.br/
[3] Private Higher Polytechnic Institute, Private University of Tunis, 32 Bis, 1002 Avenue Kheired-dine Pacha, Tunis, Tunisia
khalil.chaabani@yahoo.com

Abstract. This study presents a MATLAB-based graphical user interface (GUI) designed to acquire, process, and visualise electrocardiogram (ECG) signals using the BITalino platform. The interface allows users, including healthcare professionals and researchers, to interactively monitor ECG signals in real-time, providing an accessible and effective tool for cardiac assessment. The system ensures noise reduction and signal clarity using a three-electrode configuration, Butterworth, and notch filters. Results highlight the GUI's ability to display detailed and comprehensive ECG visualizations, aiding in identifying key ECG characteristics. This work demonstrates the potential for extending the system with machine learning algorithms for automated ECG pattern recognition.

Keywords: MATLAB · Real-time monitoring · Signal Processing

1 Introduction

Electrocardiography (ECG) is a widely used non-invasive diagnostic tool in cardiology that records the heart's electrical activity. Its simplicity, affordability, and non-invasive nature make it a valuable tool for assessing cardiac health [9].

However, the COVID-19 pandemic highlighted the traditional ECG methods limitations, particularly in telemedicine settings [6]. Concurrently, the surge in wearable technology offers a promising way to bridge this gap, empowering patients and healthcare professionals [2].

This work aims to develop an intuitive and user-friendly Graphical User Interface (GUI) coupled with signal processing techniques. This approach ensures reliable ECG signal acquisition and provides clear visualizations of ECG data.

The acquisition with Lead-I enables quick and reliable identification using only a single-limb ECG, with minimal contact and streamlined data processing [14]. The setup uses three electrodes: The positive electrode is on the left collarbone, and the negative electrode is on the right. The reference is on the iliac crest [12].

The structure of this paper is as follows: the Second Section presents a review of related works, followed by an overview of the technical specifications of the hardware and software utilized. Next, a description of the pre-processing and filtering methods is provided. Finally, the discussion and conclusion sections present this work's results and analysis.

2 Related Works

Low- and high-frequency noise can frequently impact the ECG signal, obstructing the automatic detection and classification of QRS peaks, P, and T waves. To cope with this, Costa et al. [4] used pre-processing techniques such as a moving average and a detrend function to smooth the ECG and minimize local trends. FIR digital filters significantly reduced baseline drift and high-frequency noise. An adjustable threshold based on the Pan-Tompkins algorithm was used to locate R peaks, whilst P, Q, S, and T peaks were discovered by looking for local maxima and minima inside certain zones. Pop-up windows increased the accuracy of semi-automatic peak localization. Using the MIT-BIH Arrhythmia Database, the algorithm attained an accuracy of 98.09% in QRS complex detection and 96.29% in peak identification.

Most algorithms designed to detect disturbances in ECG signals primarily use supervised learning methods [5], particularly in the case of Atrial Fibrillation (AF), a common heart disorder. Accurate manual diagnosis of AF often requires extended ECG monitoring, resulting in large datasets that are challenging to analyze. To tackle this issue, Borghi et al. [3] proposed a diagnostic assistance system that distinguishes between AF and non-AF segments using a multilayer perceptron (MLP) neural network trained on the MIT-BIH Atrial Fibrillation database. Their model achieved an average accuracy of 80.67% when tested using a 10-fold cross-validation approach.

The importance of detecting atrial fibrillation (AF) is emphasized in the study by Lazaretti et al. [8], which explores an automated ECG analysis method. This method utilizes the BITalino platform to acquire ECG signals and transmit data to an Android smartphone via Bluetooth. The signal processing is carried out using an algorithm developed by Borghi et al. [3], within an Android app created in Android Studio. This approach offers a portable solution for real-time analysis, although it does not utilize all available algorithm parameters because of the process limitation.

Using 60-second ECG segments from the MIT-BIH database, a MATLAB-based GUI was created to help in the diagnosis of cardiac disorders [11], notably differentiating between malignant ventricular ectopy (MVE) and normal sinus rhythm (NSR).

Furthermore, some GUI developments are designed to deliver precise R-R interval data for heart rate variability (HRV) studies [7]. This research creates an inexpensive Arduino-based ECG monitoring electronic device. The hardware kit has a modular design and can be utilized in classroom activities. ECG measurements are performed using the smartphone app's graphical operator interface. To determine the specific R wave position, the ECG data is analysed using three types of filtering (median filter, FIR filter, and wavelet filter).

The use of MATLAB to build a GUI for analysing ECG signals is well-established in the literature, with numerous studies demonstrating its effectiveness. However, this study presents a MATLAB-based GUI for real-time ECG signal processing, which allows the use and evaluation of pathology detection algorithms, providing a connection between these systems and portable ambulatory acquisition of ECG signals.

3 Material and Methods

The proposed solution was developed using the MATLAB platform, which provides a comprehensive array of tools for signal processing and graphical user interface development. The interface was constructed with a modular approach, facilitating the acquisition and recording of ECG data after the digital processing of these signals.

3.1 BITalino (r)evolution

BITalino (r)evolution, illustrated in Figure 1 is a versatile toolkit for acquiring biosignals such as ECG, electromyogram (EMG), and electroencephalogram (EEG).

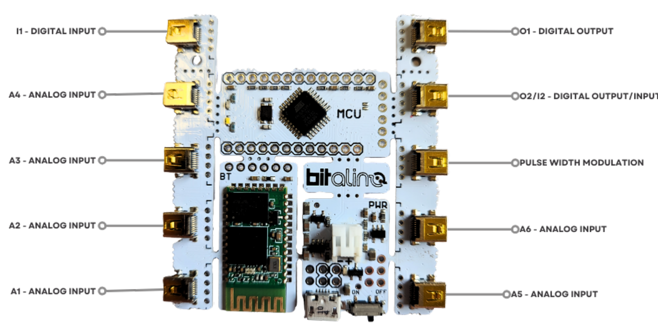

Fig. 1. Schematic of BITalino Board. Author's Elaboration

This platform is specifically designed to meet the needs of the physiological computing community [15].

The MathWorks, responsible for MATLAB, provides a BITalino Toolbox [10] within a comprehensive suite of functionalities, allowing the identification of nearby BITalino devices, simultaneous data acquisition from multiple channels, and the ability to start or stop the data collection process as required.

The Table 1 show the technical specifications of BITalino platform [13].

Table 1. Technical Specifications of the BITalino (r)evolution Core

Specification	**Details**
Sampling Frequency	1, 10, 100, or 1000 Hz
Range	Up to approximately 10 m (line of sight)
Analog Ports	4 inputs (10-bit), 2 inputs (6-bit), 1 auxiliary input (battery)
Digital Ports	2 inputs (1-bit), 2 outputs (1-bit)
Communication	Bluetooth 2.0
Size	65 x 5 mm
Microcontroller (MCU)	Atmel ATmega328p
Battery	Integrated 3.7 V Li-Po rechargeable

3.2 Pre-Processing

In this paper, the BITalino platform served as the hardware component for ECG data acquisition; after acquisition, signal conditioning is performed to prepare the data for visualization. This process is illustrated in Figure 2

Fig. 2. Schematic of processed ECG. Author's Elaboration

The raw ECG data acquired using the BITalino electrode consists of analogue values representing the measured voltage, converted in a 10-bit analog-to-digital converter (ADC) embedded in the BITalino board.

In the GUI, the digital values are divided by the maximum value from the 10-bit ADC to normalise the signal, resulting in a range between 0 and 1. Subsequently, a constant is subtracted to shift the range to a centred interval around zero. The normalized signal is then scaled by multiplying it by the voltage reference of the ADC and a calibration factor specific to the ECG amplifier. Finally, the units are converted from volts to millivolts by multiplying by a conversion factor 1000.

This processing workflow can also be expressed mathematically, as shown in Equation 1.

$$\mathrm{ECG}_{\mathrm{processed}} = \left[\left(\frac{\mathrm{ECG}_{\mathrm{raw}}}{2^{10} - 1} - 0.5 \right) \times 3.3 \right] \times \frac{1}{1100} \times 1000 \tag{1}$$

3.3 Filter

Butterworth filters were chosen for ECG signal processing and configured by the Nyquist theorem. The low-pass filter has a cutoff frequency of 150 Hz, effectively attenuating high-frequency noise while preserving critical ECG components.

A notch filter at 50 Hz [1] was applied in cascade with the low-pass filter to eliminate power line interference and two harmonics, 100 Hz and 150 Hz, resulting in cleaner ECG signals.

3.4 MATLAB and App Design

MATLAB, a comprehensive computing environment, offers a powerful platform for creating GUI and ensures compatibility with various detection algorithms.

MATLAB and App Designer are used for GUI development, but their programming paradigms differ. MATLAB uses sequential execution of commands, while App Designer uses an event-driven approach, triggering actions when specific events occur, like button clicks or text box changes.

This allows developers to create more flexible and interactive interfaces without defining the order in which functions are called.

4 Results & Discussion

This section presents the work results for a better understand of the GUI system's workflow, a visual representation is provided on Figure 3.

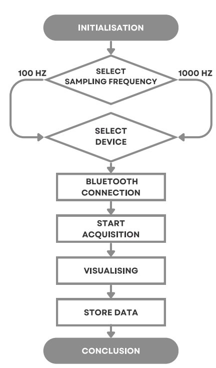

Fig. 3. GUI for ECG visualisation. Author's elaboration

Raw ECG signals were initially captured and subjected to the signal conditioning and filtering steps outlined in the previous section.

The MATLAB-based graphical user interface, illustrated on Figure 4 for signal visualisation and user interaction, highlights its usability and practical application in real-time monitoring scenarios.

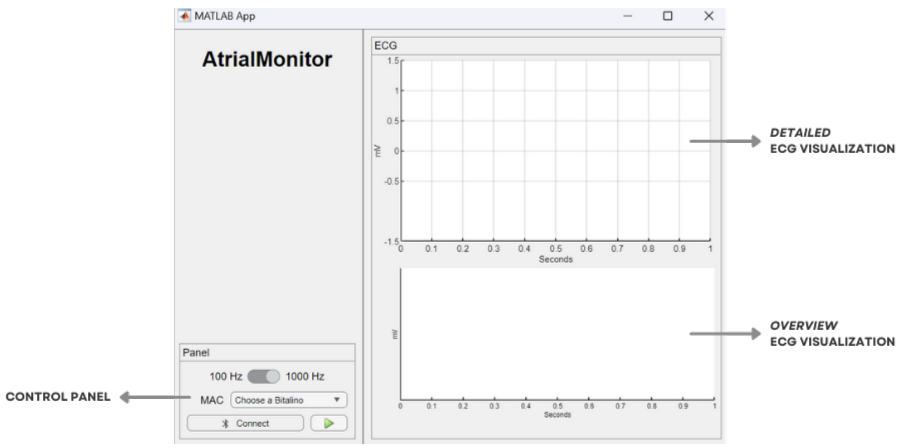

Fig. 4. GUI for ECG visualization. Author's elaboration

The platform showed in the Figure 4 was organized into three sections:

1. **Control Panel:** This section offered users various options for interacting with the system, including selecting the sampling frequency, choosing the BITalino device, establishing a connection to the BITalino, and initiating data analysis.
2. **Detailed ECG Visualization:** This area displayed a magnified view of the last five seconds of the ECG signal, allowing for a close examination of specific features and events.
3. **Overview ECG Visualization:** This section presented a comprehensive overview of the ECG recording, providing context and enabling the identification of overall trends and patterns.

As depicted in Figure 5, the GUI effectively processes an ECG signal. A notable feature of the **Detailed ECG Visualization** is the inclusion of a grid, which provides visual cues to assist users in identifying key ECG characteristics.

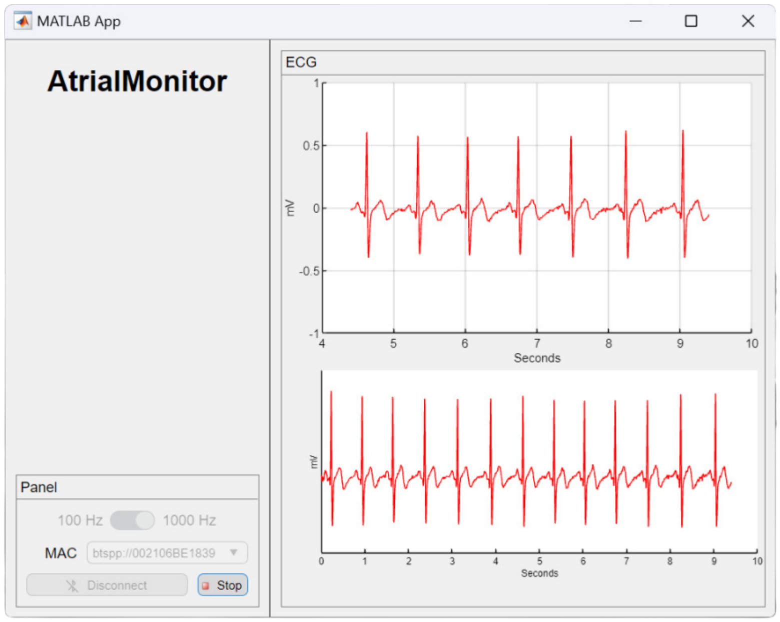

Fig. 5. GUI with ECG signal. Author's elaboration

5 Conclusion

This research was to develop a comprehensive GUI for acquiring, processing, filtering, and visualizing ECG signals.

The GUI was designed to empower users, including healthcare professionals and researchers, with a powerful tool for analysing ECG data and extracting valuable insights.

Future improvements could enhance the system by incorporating a segmentation algorithm specifically designed to identify R-wave peaks within the QRS complex of ECG signals. This would provide more relevant information for subsequent analysis.

Additionally, exploring the application of machine learning models to classify ECG signals could offer a promising avenue for automated atrial fibrillation detection. By leveraging the power of machine learning, the system could be accurate and efficient, and the user experience could be evaluated.

Acknowledgment

This work was supported by national funds through FCT/MCTES (PIDDAC): CeDRI, UIDB/05757/2020 (DOI: 10.54499/UIDB/05757/2020) and UIDP/05-

757/2020 (DOI: 10.54499/UIDP/05757/2020); and SusTEC, LA/P/0007/2020 (DOI: 10.54499/LA/P/0007/2020).

References

1. Alcaraz, R., Hornero, F., Rieta, J.J.: Dynamic time warping applied to estimate atrial fibrillation temporal organization from the surface electrocardiogram. Medical Engineering & Physics **35**(9), 1341–1348 (2013). https://doi.org/10.1016/j.medengphy.2013.03.004, https://www.sciencedirect.com/science/article/pii/S1350453313000581
2. Bisht, R.S., Jain, S., Tewari, N.: Study of wearable iot devices in 2021: Analysis & future prospects. In: 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM). pp. 577–581 (2021). https://doi.org/10.1109/ICIEM51511.2021.9445334
3. Borghi, P.H., Borges, R.C., Teixeira, J.P.: Atrial fibrillation classification based on mlp networks by extracting jitter and shimmer parameters. Procedia Computer Science **181**, 931–939 (2021). https://doi.org/10.1016/j.procs.2021.01.249, https://www.sciencedirect.com/science/article/pii/S1877050921002921, CENTERIS/ProjMAN/HCist 2020
4. Costa, R., Winkert, T., Manhães, A., Teixeira, J.P.: Qrs peaks, p and t waves identification in ecg. Procedia Computer Science **181**, 957–964 (2021). https://doi.org/10.1016/j.procs.2021.01.252, https://www.sciencedirect.com/science/article/pii/S1877050921002957, CENTERIS/ProjMAN/HCist 2020
5. Kaplan Berkaya, S., Uysal, A.K., Sora Gunal, E., Ergin, S., Gunal, S., Gulmezoglu, M.B.: A survey on ecg analysis. Biomedical Signal Processing and Control **43**, 216–235 (2018). https://doi.org/10.1016/j.bspc.2018.03.003, https://www.sciencedirect.com/science/article/pii/S1746809418300636
6. Khalili, M., GholamHosseini, H., Lowe, A., Kuo, M.M.Y.: Motion artifacts in capacitive ecg monitoring systems: a review of existing models and reduction techniques. Medical & Biological Engineering & Computing (Jul 2024). https://doi.org/10.1007/s11517-024-03165-1, http://dx.doi.org/10.1007/s11517-024-03165-1
7. Lai, Y.H., Su, K., You, Y.Q., Liang, Y.X., Lan, L., Lee, M.H., Huang, F., Chen, D.Q.: Development of arduino based ecg device for stem education and hrv applications. In: 2022 8th International Conference on Control, Automation and Robotics (ICCAR). pp. 329–333 (2022). https://doi.org/10.1109/ICCAR55106.2022.9782605
8. Lazaretti, G.S., Teixeira., J.P., Kuhn., E.V., Borghi., P.H.: Android-based ecg monitoring system for atrial fibrillation detection using a bitalino® ecg sensor. In: Proceedings of the 15th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2022) - BIODEVICES. pp. 177–184. INSTICC, SciTePress (2022). https://doi.org/10.5220/0010905400003123
9. Liu, X., Wang, H., Li, Z., Qin, L.: Deep learning in ecg diagnosis: A review. Knowledge-Based Systems **227** (2021). https://doi.org/10.1016/j.knosys.2021.107187
10. MathWorks Instrument Control Toolbox Team: Bitalino toolbox - file exchange - matlab central. https://www.mathworks.com/matlabcentral/fileexchange/53983-bitalino-toolbox-deprecated (2024), accessed: 2024-10-20

11. Mohonta, S.C., Tara, K., Ali, M.F.: Cardiac disorder detection based on features analysis and cnn in gui from ecg signals. In: 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI). pp. 1–5 (2021). https://doi.org/10.1109/ACMI53878.2021.9528206
12. PLUX Biosignals: Bitalino (r)evolution home guide experimental guides to meet & learn your biosignals. `https://support.pluxbiosignals.com/wp-content/uploads/2022/04/HomeGuide2_ECG.pdf` (2020), accessed: 2024-10-30
13. PLUX Biosignals: Bitalino (r)evolution core component data sheet. `https://bitalino.com/storage/uploads/media/bitalino-core-datasheet.pdf` (2024), accessed: 2024-10-30
14. Sahadat, M.N., Jacobs, E.L., Morshed, B.I.: Hardware-efficient robust biometric identification from 0.58 second template and 12 features of limb (lead i) ecg signal using logistic regression classifier. In: 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. pp. 1440–1443 (2014). https://doi.org/10.1109/EMBC.2014.6943871
15. da Silva, H.P., Guerreiro, J., Lourenço, A., Fred, A., Martins, R.: Bitalino: A novel hardware framework for physiological computing. In: Proceedings of the International Conference on Physiological Computing Systems - Volume 1: PhyCS,. pp. 246–253. INSTICC, SciTePress (2014). https://doi.org/10.5220/0004727802460253