Software Requirements Specification for RwaveDetection

Junwei Lin

April 19, 2025

Contents

Revision History

Date	Version	Notes
January 16, 2025	1.0	Creation
April 10, 2025	1.1	Revision

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	**unit**	**SI**
s	time	second
V	voltage	volt
Hz	frequency	hertz (Hz = s^{-1})

Exceptionally, some units officially accepted for use with the SI are used as described below.

symbol	**unit**
dB	logarithmic ratio quantity

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the existing documentation for the R-wave detector. The symbols are listed in alphabetical order.

symbol	**unit**	**description**
a	N/A	a sequence of digital filter weighting coefficients
A	mV	a sequence of index of the annotated R wave
b	N/A	a sequence of digital filter feedback coefficients
f_c	Hz	cutoff frequency of the filter
f_s	Hz	sampling frequency of the ECG signal
H	N/A	transfer function of digital filter
R	N/A	a sequence of index of the calculated R wave
u	mV	a sequence of discrete-time ECG signal
z	N/A	complex frequency variable in the Z-domain, $z = e^{j2\pi f}$

1.3 Abbreviations and Acronyms

symbol	**description**
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
RwaveDetection	R-wave Detection
TM	Theoretical Model
ECG	Electrocardiography
ADC	Analog-to-Digital Converter
RMSE	Root Mean Square Error
QRS	Cardiac cycle

1.4 Mathematical Notation

In the fields of electronics and signal processing, j represents the imaginary unit. In other word, we define $j^2 = -1$.

In discrete signal processing, square brackets are used to represent discrete signals and round brackets are used to represent continuous signals. For instance, $u(t)$ represents the signal amplitude of a continuous-time signal at time t, and $u[n]$ represents the signal amplitude of a discrete-time signal at the nth index.

2 Introduction

R-wave detection is a critical task in ECG signal processing, serving important purposes such as heart rate calculation, arrhythmia analysis, cardiac conduction evaluation, and heart disease diagnosis. The patient's ECG signal is sampled by the sensor and converted into a digital signal, where we can detect R-waves. In the ECG waveform, the R-wave refers to the peak of the QRS complex, which represents the electrical depolarization of the heart's ventricles — typically the most prominent feature in an ECG cycle.

However, since the sampling is not under an ideal condition, a large amount of clutter wave is mixed into the original signal, such as electrical noise and utility frequency interference (commonly 50 or 60 Hz depending on region, originating from power lines).

There are many algorithms for detecting R-waves. This project will focus on reimplementing the Pan-Tompkins [1] algorithm. All filter parameters used in the algorithm will be automatically derived according to requirements without relying on external libraries. The software system designed in this project is named RwaveDetection.

To clarify the concepts described above, a sample ECG figure with R-wave is provided in the Figure 1.

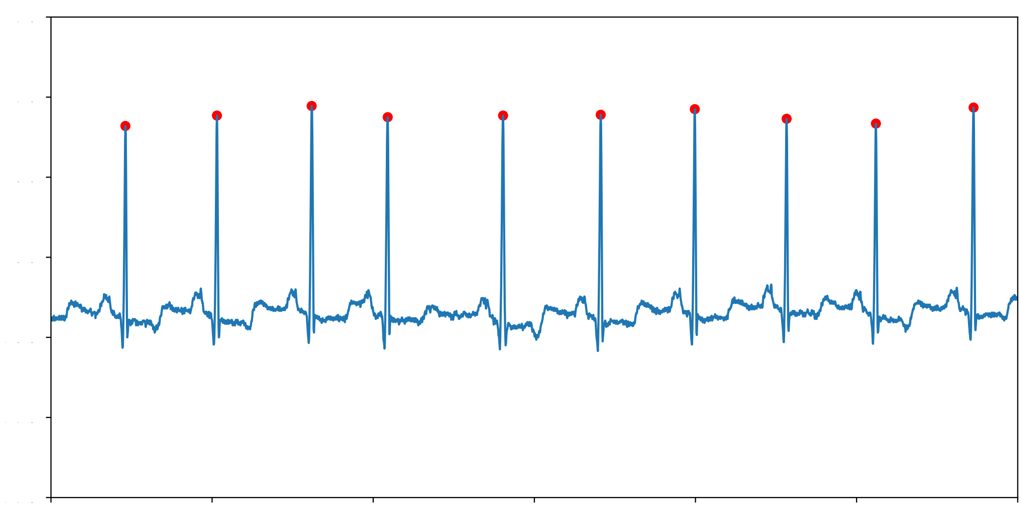


Figure 1: A sample ECG waveform with red dots indicating the position of each R-wave.

2.1 Purpose of Document

The primary purpose of this document is to record the requirements of RwaveDetection. Goals, assumptions, theoretical models, definitions, and other model derivation information are specified, allowing the reader to fully understand and verify the purpose and scientific basis of RwaveDetection. With the exception of system constraints, this SRS will remain abstract, describing what problem is being solved, but not how to solve it.

This document will be used as a starting point for subsequent development phases, including writing the design specification and the software verification and validation plan. The design document will show how the requirements are to be realized, including decisions on the numerical algorithms and programming environment. The verification and validation plan will show the steps that will be used to increase confidence in the software documentation and the implementation. Although the SRS fits in a series of documents that follow the so-called waterfall model, the actual development process is not constrained in any way.

2.2 Scope of Requirements

The scope of the requirements includes processing sampled data from a single channel.

2.3 Characteristics of Intended Reader

The intended readers of this document are upper-level undergraduate or graduate students, researchers, and professionals with a background in signal processing, biomedical engineering, or software development, particularly those experienced in digital filters and ECG signal analysis.

2.4 Organization of Document

The organization of this document follows the template for an SRS for scientific computing software. For readers that would like a more bottom up approach, they can start reading the instance models and trace back to find any additional information they require.

The goal statements are refined to the theoretical models and the theoretical models to the instance models. IM1 describes a series of processes for R-wave detection, which solved GS1. IM2 established a model to calculate RMSE, which solved GS2.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

3.1 System Context

The RwaveDetecion system receives sample frequency f_s and discrete time domain ECG signal u from the ADC(Analog-to-Digital Converter). Based on this information, the index of each R-wave peak R can be calculated and returned to the user. If the user provides annotated data A, which is optional, the system will return the RMSE between A and R additionally.

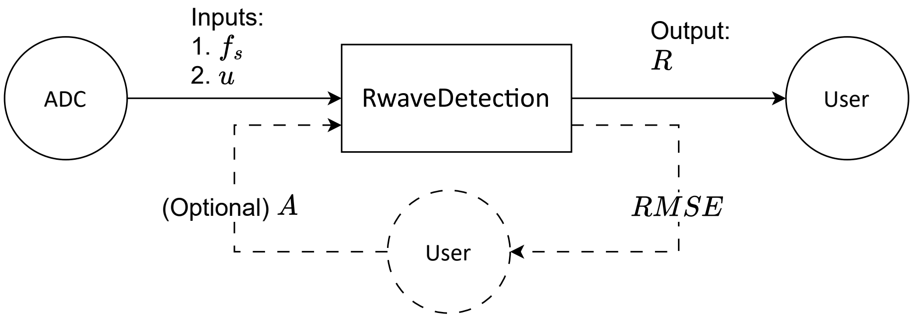


Figure 2: System Context

- User Responsibilities:
  - Provide integer ADC sample frequency f_s
  - Provide discrete time domain ECG signal real sequence u sampled from ADC
  - (Optional) Provide correctly annotated data real sequence A
- RwaveDetection Responsibilities:
  - Detect data type mismatch, such as a string of characters instead of a real number
  - Return the integer sequence index of each R-wave peak R detected from u
  - Render a graph that shows u and R at the same time
  - Return the RMSE between A and R if required

3.2 User Characteristics

The end user of RwaveDetection needs to have a basic understanding of the QRS complex in ECG. Advanced user who hopes to evaluate the accuracy of the system should have an understanding of undergraduate level mathematics.

3.3 System Constraints

RwaveDetection must process data in real time with a maximum latency of 50 milliseconds per data sample, ensuring suitability for real-time monitoring systems. The algorithm must be based on the Pan-Tompkins method for QRS complex detection to maintain accuracy and efficiency. To support various low-performance embedded devices, the implementation must prioritize efficiency and compatibility with the target hardware.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

RwaveDetection is intended to find the R-wave peak position accurately from single channel ECG data containing clutter and noise.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- QRS complex: the combination of three of the graphical deflections seen on a typical electrocardiogram (ECG or EKG) [2], comprising three distinct graphical deflections — the Q wave, R wave, and S wave. It corresponds to the depolarization of the ventricles, which is a crucial phase in the heart's electrical cycle, reflecting the contraction of the ventricles. This complex is often used to assess heart rhythm and detect abnormalities
- R-wave: represents the peak of ventricular depolarization
- Filter: a device or process that removes some unwanted components or features from a signal [3]
- Utility frequency: the nominal frequency of the oscillations of Alternating Current (AC) in a wide area synchronous grid transmitted from a power station to the end-user. [4]

4.1.2 Physical System Description

The physical system of RwaveDetection, includes the following elements:

PS1: The heart generates electrical impulses that propagate through the body, causing voltage differences at the skin surface.

PS2: Electrodes placed on the body detect the electrical signals from the heart. These signals are captured in the form of voltage differences between the electrode pairs.

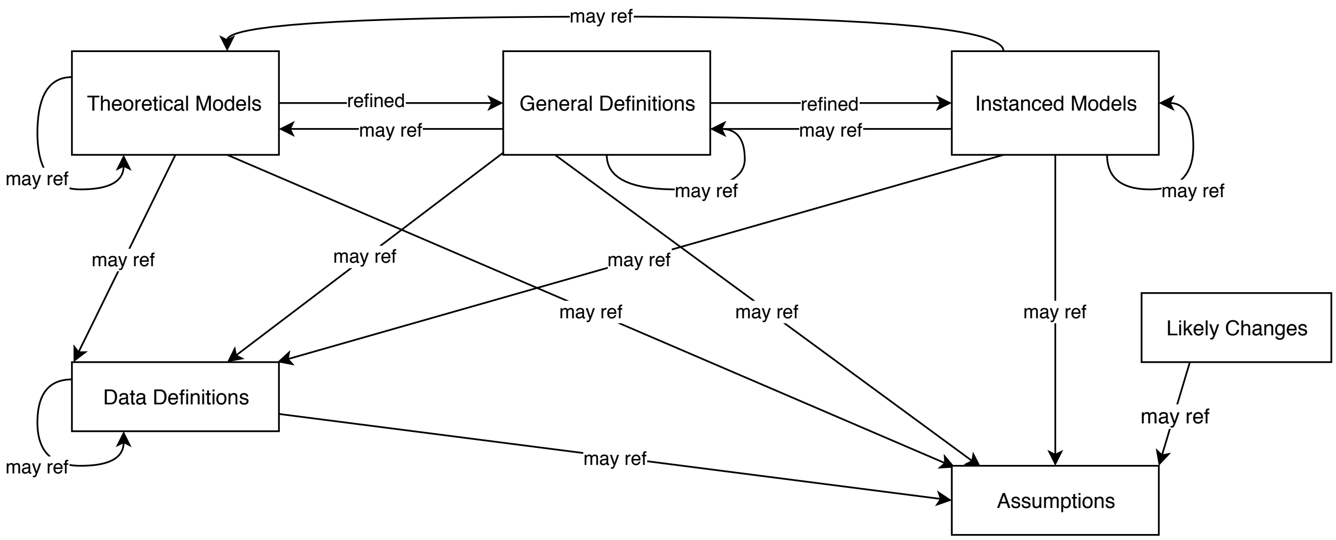
PS3: Noise elements such as muscle artifacts, motion artifacts, and electrical interference from external sources (e.g., power lines) can corrupt the ECG signal.

4.1.3 Goal Statements

GS1: Given a single-channel unfiltered ECG signal, find the index of each R-wave peak.

GS2: Given correct annotated data, calculate the RMSE between each detected R-wave peak time and annotated time.

4.2 Solution Characteristics Specification



The instance models that govern RwaveDetection are presented in Subsection 4.2.5. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the theoretical model [TM], general definition [GD], data definition [DD], instance model [IM], or likely change [LC], in which the respective assumption is used.

A1: The sampling frequency is not lower than the Nyquist frequency.

A2: Noise in data can be filtered out, within expectations.

A3: The R-wave has peak characteristics that can be discerned by the naked eye.

A4: The input ECG signal is of high quality.

4.2.2 Theoretical Models

RefName: **TM1**

Label: Analog Filter Equation

Equation: $H(s) = \frac{\sum_{i=0}^{m} b_i s^i}{\sum_{i=0}^{n} a_i s^i}$

Description:
H is the transfer function of the filter.
s is the Laplace variable (complex frequency).
b_m, b_{m-1}, ..., b_0 are the numerator coefficients.
a_n, a_{n-1}, ..., a_0 are the denominator coefficients, where $a_n \neq 0$ to ensure causality and stability.
n and m represent the highest order of the denominator and numerator polynomials, respectively. Usually, $n \geq m$ to maintain realizability (strictly causal system).

Notes: Transfer functions are commonly used in the analysis of systems such as single-input single-output filters in signal processing, communication theory, and control theory.

Source: https://en.wikipedia.org/wiki/Transfer_function

Ref. By: TM2, TM3, GD1, GD2, IM1

Preconditions for TM1: None

Derivation for TM1: Not Applicable

RefName: **TM2**

Label: Butterworth Filter

Equation: $G_n(\omega) = \frac{G_0}{\sqrt{1+(\frac{\omega}{\omega_c})^{2n}}}$

Description:
G_n is the gain function of the nth-order Butterworth filter.
G_0 is the DC gain (gain at zero frequency).
ω is the angular frequency of the signal.
ω_c is the angular cutoff frequency (Hz) which defines the -3dB point.
n is the order of the filter.

Notes: The frequency response of the Butterworth filter is maximally flat (i.e., has no ripples) in the passband and rolls off towards zero in the stopband.

Source: https://en.wikipedia.org/wiki/Butterworth_filter

Ref. By: GD1, GD2, IM1

Preconditions for TM2: None

Derivation for TM2: Not Applicable

RefName: TM3

Label: Chebyshev Filter

Equation: $G_n(\omega) = \frac{1}{\sqrt{1+\epsilon^2 T_n^2(\omega/\omega_c)}}$

Description:
G_n is the gain function of the nth-order Butterworth filter.
ϵ is the ripple factor, $|\epsilon| < 1$
ω is the angular frequency of the signal.
ω_c is the angular cutoff frequency (Hz) which defines the -3dB point.
$T_n(\frac{\omega}{\omega 0})$ is the nth order Chebyshev polynomials.

Notes: Chebyshev filters have a steeper roll-off than Butterworth filters, and have either passband ripple (type I) or stopband ripple (type II).

Source: https://en.wikipedia.org/wiki/Chebyshev_filter

Ref. By: GD1, GD2, IM1

Preconditions for TM3: None

Derivation for TM3: Not Applicable

RefName: **TM4**

Label: Threshold Detect

Equation: $T[n] = \begin{cases} x[n] & \text{if } x[n] \geq \theta \\ 0 & \text{if } x[n] < \theta \end{cases}$

Description:
x is the input signal.
T is the output signal.
θ is the threshold, which is not guaranteed to be a static number all the time and may change as the state changes.

Notes: threshold comparison detection is a simple signal processing technique where the input signal is compared to a threshold value.

Source:

Ref. By: IM1, IM2

Preconditions for TM4: None

Derivation for TM4: Not Applicable

4.2.3 General Definitions

This section collects the laws and equations that will be used in building the instance models.

Number	GD1
Label	**Bilinear transform**
Equation	$s \approx \frac{2}{T_s} \cdot \frac{z-1}{z+1}$
Description	The bilinear transform is used in digital signal processing and discrete-time control theory to transform continuous-time system representations to discrete-time. s is the Laplace variable (complex frequency). z is the complex frequency variable in the Z-domain. T_s is the sampling period. When A1 is satisfied, this equation represents replacing all s terms in $H(s)$ S-domain transfer function directly with $\frac{2}{T} \cdot \frac{z-1}{z+1}$, after doing this we can get Z-domain transfer function H(z). Analog filters in TM2 and TM3 can be transformed into digital filter in this way.
Source	https://en.wikipedia.org/wiki/Bilinear_transform
Ref. By	TM1, TM2, TM3, GD2, GD3, IM1

Number	GD2
Label	**Filter Difference Equation**
Equation	$H(z) = \frac{\sum_{l=0}^{N} b_l z^{-l}}{1+\sum_{k=1}^{M} a_k z^{-k}} \leftrightarrow y[n] = -\sum_{k=1}^{M} a_k y[n-k] + \sum_{l=0}^{N} b_l x[n-l]$
Description	$x[n]$ is the input signal at the n-th time sample. $y[n]$ is the output signal at the n-th time sample. b_l is the coefficients for the input signal. a_k is the coefficients for the output signal. N is the maximum delay for the input signal. M is the maximum delay for the output signal. z is the complex frequency variable in the Z-domain. This equation reflects the conversion relationship between the difference equation and the Z-domain transfer function $H(z)$ comes from GD1.
Source	https://en.wikipedia.org/wiki/Digital_filter
Ref. By	TM1, TM2, TM3, GD1, GD3, IM1

Number	GD3
Label	**Differential filter**
Equation	$y[n] = \frac{1}{8}(2x[n] + x[n-1] - x[n-3] - 2x[n-4])$
Description	$x[n]$ is the input signal at the n-th time sample. $y[n]$ is the output signal at the n-th time sample. The five-point differentiation formula is a numerical method used to approximate the derivative of a discrete signal. This filter is actually a special digital filter using difference equations in GD2. In this program, differential filter can be used to enhance the slope of the QRS complex.
Source	https://en.wikipedia.org/wiki/Digital_filter
Ref. By	TM1, IM1

4.2.4 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given.

Number	DD1
Label	**Period of sampling**
Symbol	T_s
SI Units	s
Equation	$T_s = \frac{1}{f_s}$
Description	T_s is the period of sampling (s). f_s is the frequency of sampling (Hz).
Sources	https://en.wikipedia.org/wiki/Sampling_(signal_processing)#Sampling_rate
Ref. By	GD1, IM1

4.2.5 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.4 to replace the abstract symbols in the models identified in Sections 4.2.2 and 4.2.3.

The goal GS1 is solved by IM1 and GS2 is solved by IM2.

Number	IM1
Label	**Algorithm to detect R-wave peak index**
Input	$u,\ f_s,\ N$
Output	$$u_f[n] = -\sum_{k=1}^{M} a_k u_f[n-k] + \sum_{l=0}^{N} b_l u[n-l]$$ $$u_w[n] = \frac{1}{W}\sum_{k=0}^{W-1} u_f[n-k]^2$$ $$u_{th}[n] = \begin{cases} u_w[n] & \text{if } u_w[n] \geq \theta \\ 0 & \text{if } u_w[n] < \theta \end{cases}$$ $$R[i] = \arg\max_n u_{th}[n]$$
Description	u is the sequence of original ECG signal input data. a and b are sequence of digital filter parameters. M and N are the length of filter parameters. W is the window size of sliding window integral. This size needs to be adjusted manually. θ is the R-wave detecting threshold, which is designed to be automatically adjusted by the program. R is the final output of the program, which is a sequence of R-wave index we hope to detect in GS1.
Sources	https://en.wikipedia.org/wiki/Pan%E2%80%93Tompkins_algorithm
Ref. By	TM1, TM2, TM3, TM4, GD1, GD2, GD3, DD1, IM2

Details of IM1

When calculating u_f, The filter parameters a and b are linear combinations of a series of filter parameters from GD1. The sequence of filters including low-pass and high-pass filter from TM2 and TM3 and differential filter from GD3.

We use the square function and sliding window integral to calculate u_w, in this way the R-wave can be amplified and the signal can be smoothed.

TM4 is used when calculating u_{th}, so we can focus on extracting the R-wave and ignore the influence of other signals.

In the last equation, the variable n has a range limitation, we only take a continuous period of n where $u_{th}[n]$ is non-zero.

Number	IM2
Label	**calculate RMSE between A and R**
Input	A, R
Output	$$\text{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{(A[i] - R[i])^2}{n}}$$ In special cases: If the lengths of A and R are not the same, for each element of A, find the closest element in R within the tolerance range, and then calculate the RMSE based on this.
Description	A is the sequence of integers containing annotated index. R is the sequence of integers containing calculated index from IM1.
Sources	None
Ref. By	TM4, IM1

4.2.6 Input Data Constraints

Table 2 shows the data constraints on the input output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise.

The specification parameters in Table 2 are listed in Table 4.

Table 2: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value
f_s	$f_s > 0$	$f_{s\ \mathrm{min}} \leq f_s \leq f_{s\ \mathrm{max}}$	360 Hz
u	$u[n] \geq 0$	$0 \leq u[n] \leq u_{\mathrm{max}}$	0.5mV
N	$N > 0$	$0 < N \leq N_{\mathrm{max}}$	36000
A	$A[n] \geq 0$	$0 \leq A[n] \leq A_{\mathrm{max}}$	15000

Table 4: Specification Parameter Values

Var	Value
$f_{s\ \mathrm{min}}$	100 Hz
$f_{s\ \mathrm{max}}$	5000 Hz
u_{max}	10 mV
N_{max}	1,000,000
A_{max}	100,000

5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

R1: Basic IIR and FIR filters using difference equations, where the input is discrete signal data and filter coefficients, and the output is the filtered signal.

R2: Parameter derivation of Butterworth filters, where the input is the desired filter order and cutoff frequency, and the output is the Butterworth filter coefficients.

R3: Parameter derivation of Chebyshev filters, where the input is the desired filter order, cutoff frequency, and ripple factor, and the output is the Chebyshev filter coefficients.

R4: Squaring function, where the input is a signal or value, and the output is the squared signal or value.

R5: Thresholding function, where the input is a signal or value with a threshold level, and the output is the signal with values above the threshold retained and others set to zero (or based on the chosen condition).

R6: RMSE calculation, where the input is a predicted signal and a reference (actual) signal, and the output is the RMSE value.

R7: Combine mathematical functions and filters to calculate the R-wave position, where the input is the filtered signal and mathematical functions, and the output is the calculated R-wave position.

R8: Combine annotated data and detected R-wave index to calculate RMSE, where the input is the annotated data and the detected R-wave index, and the output is the RMSE value.

5.2 Nonfunctional Requirements

NFR1: **Usability** The system must generate user manuals automatically using external tools.

NFR2: **Maintainability** The system must be fully tested with a comprehensive verification and validation plan.

NFR3: **Portability** The code must be cross-platform and work on at least Linux and Windows.

NFR4: **Reusability** The code must be modular to facilitate reuse in different contexts.

6 Likely Changes

LC1: The types of digital filters such as TM2 and TM3 may be increased according to actual needs

LC2: Some specific parameters in IM1 may be changed, such as the default filter order number, cutoff frequency, window size etc.

LC3: Some ECG data other than MITBIH may be used.

7 Unlikely Changes

LC4: The method of converting continuous-time systems to discrete-time systems will be bilinear transform in GD1 and does not change.

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 7 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 8 shows the dependencies of instance models, requirements, and data constraints on each other. Table 6 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

	A1	A2	A3	A4
TM1		X		
TM2		X		
TM3		X		
TM4		X	X	
GD1	X			
GD2	X			
GD3	X			
DD1				
IM1	X	X	X	X
IM2		X	X	X
LC1	X			
LC2	X		X	
LC3	X	X	X	X
LC4	X			

Table 6: Traceability Matrix Showing the Connections Between Assumptions and Other Items

The purpose of the traceability graphs is also to provide easy references on what has to be additionally modified if a certain component is changed. The arrows in the graphs represent dependencies. The component at the tail of an arrow is depended on by the component at the head of that arrow. Therefore, if a component is changed, the components that it points to should also be changed.

	TM1	TM2	TM3	TM4	GD1	GD2	GD3	DD1	IM1	IM2
TM1					X	X	X		X	
TM2	X				X	X			X	
TM3	X				X	X			X	
TM4									X	X
GD1	X	X	X			X		X	X	
GD2	X	X	X		X				X	
GD3					X	X			X	
DD1									X	
IM1	X	X	X	X	X	X	X	X		X
IM2				X					X	

Table 7: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM1	IM2
R1	X	
R2	X	
R3	X	
R4	X	
R5	X	
R6		X
R7	X	
R8		X

Table 8: Traceability Matrix Showing the Connections Between Requirements and Instance Models

References

[1] Jiapu Pan and Willis J. Tompkins. A real-time qrs detection algorithm. *IEEE Transactions on Biomedical Engineering*, BME-32(3):230–236, 1985.

[2] Wikipedia. QRS complex — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=QRS%20complex&oldid=1271538165, 2025. [Online; accessed 26-January-2025].

[3] Wikipedia. Filter (signal processing) — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Filter%20(signal%20processing)&oldid=1268309950, 2025. [Online; accessed 26-January-2025].

[4] Wikipedia. Utility frequency — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Utility%20frequency&oldid=1265911887, 2025. [Online; accessed 26-January-2025].