

Android Based Heart Rate Detection Tools with Arduino Nano

Hidayatul Muttaqin
STMIK Widya Cipta Dharma,
Samarinda, 75123, Indonesia
jscdayat022@gmail.com

Ita Arfyanti*
STMIK Widya Cipta Dharma,
Samarinda, 75123, Indonesia
itaarfyanti@wicida.ac.id
**Corresponding author*

Wahyuni
STMIK Widya Cipta Dharma,
Samarinda, 75123, Indonesia
wahyuni@wicida.ac.id

Received: 2021-02-27; Revised: 2021-03-11; Accepted: 2021-03-11; Published: 2021-03-14

Abstract- Android-based Heart Rate Detector Using an Android-Based Fingerprint Using Arduino Nano at Midwife Dwi Inggini's Maternity Clinic with the hope of helping and simplifying the medical team in checking the heart rate of pregnant women without having to carry devices that are not portable, improving services and errors due to blackouts PLN electricity. The software development method used is the prototype method which includes data collection, design, prototyping, the testing phase by conducting Black Box and White Box testing. To access this tool the user must first connect the bluetooth android device with bluetooth HC-05 on the Arduino device, after the two Bluetooth devices are connected.

Keywords- Prototype, Android, Heart rate detection tool, Arduino Nano

I. INTRODUCTION

The tools commonly used to check heart rates are a stethoscope that is attached to a pregnant woman's chest or uses a more sophisticated device called an Electrocardiograph (ECG), this is of course less practical and takes a long time if you have to go to the hospital and consult a heart specialist. it requires a heart rate detector that is practical, economical and easy to use and can help medical teams who are still calculating their heart rate using a stopwatch and even a heart rate detector that is difficult to carry everywhere and still uses a regular monitor. As for the benefits or functions to be achieved, namely, this tool can detect the pulse in the finger of pregnant women and process it. Then we get the number of heartbeats of pregnant women and from the number of heartbeats it will be known how many heart rates of pregnant women per minute, so the authors want to design a heart rate detector from an Android-based finger pulse using Arduino Nano and the data will be displayed to the application. on the smartphone. (Lusi, 2014; Sulistyono, 2018).

II. LITERATURE REVIEW

A. Arduino

Arduino is an open source microcontroller for which no feedback is present on the microcontroller. (Alfaruq, 2011; Asrori. Et al., 2017). This Arduino module consists of an I2C bus, which can transfer data from the Arduino board to the output device. This Arduino board is programmed via an RS232 serial connection with the Atmega Arduino microcontroller. Operating voltages range from 5v. The recommended input voltage for the Arduino microcontroller is from 7v and a maximum of 12v. The DC current input given to the Arduino board is in the range of 40mA (Sudhan, et al, 2015).

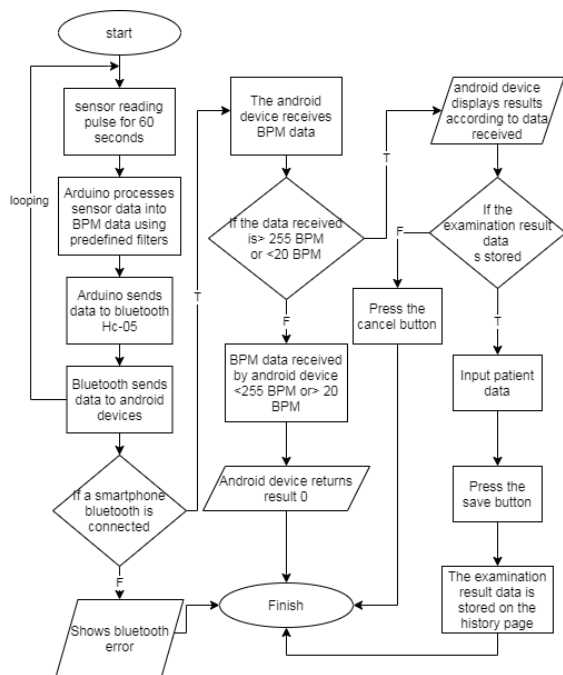
B. Prototyping Method

Software prototyping is a system life cycle method based on the concept of a working model (Jaya, 2018; Deepa, 2016). The goal is to develop the model into a final system. This means that the system will be developed faster than traditional methods and the costs will be lower. There are many ways to do prototyping, as well as use, and this model is used by the author in analyzing the system to be worked on (Lugina, 2015; Kadir, 2013). The phases in Prototyping are as follows:

1. Analysis
At this stage the developer identifies the software and all system requirements to be made.
2. Build prototyping
Build prototyping by making temporary designs that focus on serving customers (for example by creating input and output formats).
3. Encoding the system
This stage the prototyping is converted into a programming language.
4. Test the system
At this stage, it is carried out to test the software system that has been created.
5. Evaluation
The finished software will be evaluated by the customer to find out whether the system is as expected.
6. Using the system
Customer-tested and approved software is ready to use.

III. RESEARCH METHODS

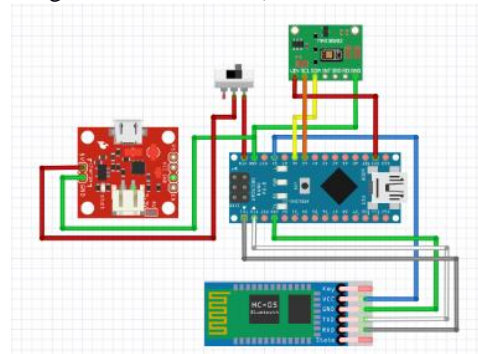
Arduino processes the data captured by the sensor using predetermined filters into BPM or beats per minute data. after the data is processed, Arduino sends data to the Bluetooth module to be sent and displayed by the application on the smartphone. If Bluetooth is connected, the Android device will receive the data sent by Arduino. If not, there will be a Bluetooth error notification on the Smartphone device and the system is complete. If the data received by the Android Device is greater than 255 BPM or less than 20 BPM, then the data displayed by the Android device will be worth 0, but if the data received is less than 255 BPM or greater than 20 BPM, then the data that is received displayed by the Android device will be valued according to the data received from the Microcontroller. After the examination result data is displayed by the Android device, the user can save the examination results by filling in the patient data and pressing the save button on the Android application, if you do not want to save the examination result data, the user can press the cancel button, and the display will return to display the result 0. 1 is a flowchart of an Android-based heart rate detection tool for finger pulse using Arduino nano. (Mardiansah, 2019; Pressman, 2015). Tool system flowchart can be seen in Picture 1. In Picture 1 is the Flowchart of the Android-Based Heart Rate Detection Tool System Using Arduino nano. (Mardiansah, 2019; Pressman, 2015). Tool system flowchart can be seen in Picture 1. In Picture 1 is the Flowchart of the Android-Based Heart Rate Detection Tool System Using Arduino nano. (Mardiansah, 2019; Pressman, 2015). Tool system flowchart can be seen in Picture 1.



Picture 1. Tool System Flowchart

B. Encoding the System

The pulse sensor used is the Pulse Sensor, to send the examination result data to the Android device the HC05 Bluetooth module is used and the media for displaying the results of the examination using the android device. (Sari, 2014). Picture 2 is the overall image of the circuit for the heart rate detector circuit, in this image everything is connected to the Arduino according to the function of each device. This design is made to make it easier to connect the cables from each device so that errors do not occur when assembling the heart rate detector. (Pamungkas & Husni, 2020).



Picture 2. Schematic of a Heart Rate Detector Circuit

IV. RESULTS AND DISCUSSION

In Picture 3, the display shows the shape of the heart rate detector, from outside the heart rate detector, there are components such as a Pulse Sensor, jumper cable, cover, data cable and Powerbank. Appearing from the outside can be seen in Picture 3.



Picture 3. Appears from the outside

In Picture 4 shows a display of the shape of a heart rate detector, from inside the heart rate detector there are components such as a Bluetooth Module, Arduino nano, Charger Module, jumper cables and PCB boards.



Picture 4. Appears from within

In picture 5 is the opening screen of the Heart rate Application before entering the Main screen.



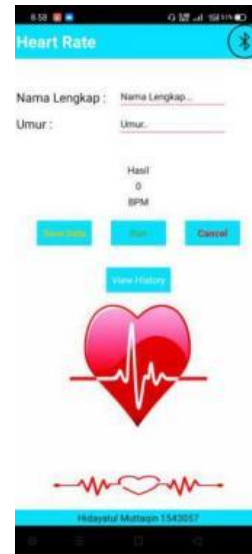
Picture 5. The opening screen of the Heart Rate Application

Picture 6 shows the Bluetooth Connection Heart rate application when you want to connect the Arduino device with an Android device.



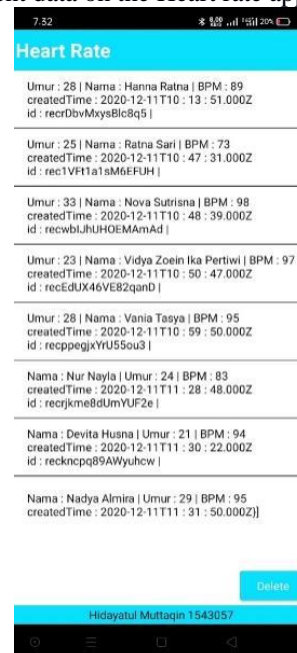
Picture 6. Display of Bluetooth Connection in Heart Rate Application

Picture 7 shows the main display of the Heart rate application. In the main display there is a column name and age to save patient data, besides that there is a Run button to start the examination, a Save Data button to save the examination data, a Bluetooth button to connect the Arduino device with an Android device, the Cancel button to cancel the examination or it can also be used to removes the examination results, and there is also a View History button to view the examination history. (Pressman, 2014)



Picture 7. Main Display of Heart Rate Application

In Picture 8 is a historical display of examination results and patient data on the Heart rate application



Picture 8. Display History of Heart Rate Application Examination

Here's how the tool works as a whole:

1. Press the on switch on the pulse detector to activate all components of the heart rate detector.
2. Open the Heart Rate app and turn on Bluetooth on Android.
3. Connect the Bluetooth HC-05 to the Heart rate app on the Android device.
4. Place and place your finger on the pulse sensor, and fill in the patient's name and age.
5. Press the run button on the Heart rate app to process the heart rate check on the pulse of the finger.
6. Wait for 5 seconds to see the results of the check that will be displayed on the Android device.
7. After the results are displayed, press the save button on the Heart Rate application to save the examination result history.
8. Press the reset button to reset the examination results.

D. Black Box Testing

Black Box testing is testing the functions of the tool, so it can be seen to what extent the tool can run well. Black Box testing can be seen in table 1.

Table 1. Black Box Testing

Case Test	Scenario Test	The results are Expected	Number of Tries	
			Success	Failed
Sensor readings	Filter and delay reading time	The accuracy rate of the sensor reads and counts the pulse up to above 90%	3	1
Bluetooth module performance	Bluetooth Connection with Android Device	Bluetooth can connect with Android devices and send data	3	1
Arduino nano performance	There are no syntax errors	Arduino can process data from sensors into BPM data and send BPM data to the Bluetooth module	2	1
Smartphone Application Display	The application display is easy for users to understand	The Application page is easy for the user to understand and manages to display and save the examination results sent by the Arduino device	2	-

So, cyclomatic complexity for Flowgraph, Heartbeat Detector, according to the value relationship of cyclomatic complexity and risk, is included in a fairly

complex procedure with low risk. From the calculation of cyclomatic complexity, an independent path will be determined, of which there are 3 pathways for this Heartbeat Detector, including:

Line 1 = 1 - 2 - 3 - 4 - 13 - 12

Description: The Arduino device does not read the heart rate and displays a 0 BPM result

Line 2 = 1 - 2 - 3 - 5 - 6 - 7 - 8 - 9 - 13 - 12

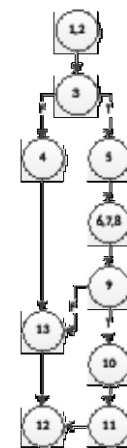
Note: The Arduino device reads the heart rate but the calculated rate is <225 BPM or> 20 BPM, so the processed data will display the result 0 BPM

Line 3 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 9 - 11 - 13

Note: The Arduino device reads a heart rate > 255 BPM or <20 BPM, so the processed data will display the appropriate BPM results. The value of cyclomatic complexity V (G) for the flow graph is calculated by the formula:

1. Flowgraph has 3 regions
2. $V(G) = 11 \text{ edges} - 10 \text{ nodes} + 3 = 4$
3. $V(G) = 2 P + 1 = 3$

Flowgraph of a heart detector can be seen in Picture 9.



Picture 9. Flowgraph of Heartbeat Detector

The value of cyclomatic complexity V (G) for the flow graph is calculated by the formula:

1. Flowgraph has 3 regions
2. $V(G) = 11 \text{ edges} - 10 \text{ nodes} + 3 = 4$
3. $V(G) = 2 P + 1 = 3$

After making the test case, proceed to validation by conducting program trials, so that the expected results can be found in accordance with the results of the application. The following is some of the code used in the test scenario, including:

1. *Valid* indicates that the components that make up the scenario have true or valid values, thus making the system successful.
2. *Invalid* indicates that the components that make up the scenario have invalid or invalid values.

Testing of several program modules on the Android-based heart rate detector from finger pulse using Arduino nano. Testing includes the type of test, type of program,

expected results and test results. White Box testing that is carried out can be seen in table 2. (Susilo, 2014).

Table 2. White Box Testing

Condition	Expected results	Result	Validation
The sensor reads a beat	The sensor sends data to the microcontroller to be processed into BPM data	the data is processed into BPM data	Valid
The sensor does not read ticks	The sensor sends data to the microcontroller to be processed into 0 BPM data	the data is processed into 0 BPM data	Valid
Display of examination result data <255 BPM or > 20 BPM	The data displayed on the Heart Rate application is 0 BPM	The data displayed on the Heart Rate application is 0 BPM	Valid
Display of examination result data > 255 BPM or <20 BPM	The data displayed on the Heart Rate application corresponds to the detected pulse	The data displayed on the Heart Rate application corresponds to the detected pulse	Valid
Bluetooth connected	The Arduino device is connected to an Android device, data can be sent and displayed by the Heart Rate Application	The Arduino device is connected to an Android device, data can be sent and displayed by the Heart Rate Application	Valid
Bluetooth not connected	The Arduino device is not connected to an Android device, data cannot be sent and displayed by the Heart Rate Application	The Arduino device is not connected to an Android device, data cannot be sent and displayed by the Heart Rate Application	Valid
Save examination result data	Examination result data can be saved to the History page	Examination result data can be saved to the History page	Valid

In Table 3, there are some comparisons of data from the direct examination test results used by midwives to patients using Arduino with an EKG.

Table 3. Tools comparison table

Patient's name	Age	Testing result data tools	EKG result data	Level of accuracy
Hanna Ratna	28 years	1) 89 Bpm	1) 88 Bpm	99%
		2) 84 Bpm	2) 83 Bpm	
		3) 85 Bpm	3) 86 Bpm	
Ratna Sari	25 years	1) 73 Bpm	1) 74 Bpm	99%
		2) 77 Bpm	2) 78 Bpm	
		3) 79 Bpm	3) 80 Bpm	

Nova Sutrisna	33 years	1) 98 Bpm 2) 91 Bpm 3) 95 Bpm	1) 98 Bpm 2) 90 Bpm 3) 94 Bpm	99%
Vidya Zoelin	23 years	1) 97 Bpm 2) 96 Bpm 3) 92 Bpm	1) 98 Bpm 2) 95 Bpm 3) 90 Bpm	98%
Vania Tasya	28 years	1) 95 Bpm 2) 90 Bpm 3) 96 Bpm	1) 95 Bpm 2) 91 Bpm 3) 97 Bpm	99%
Nur Nayla	24 years old	1) 83 Bpm 2) 88 Bpm 3) 80 Bpm	1) 82 Bpm 2) 89 Bpm 3) 80 Bpm	99%
Devita Husna	21 years	1) 94 Bpm 2) 91 Bpm 3) 95 Bpm	1) 93 Bpm 2) 90 Bpm 3) 96 Bpm	99%
Nadya Almira	29 years	1) 95 Bpm 2) 93 Bpm 3) 92 Bpm	1) 96 Bpm 2) 92 Bpm 3) 90 Bpm	98%

In picture 10 is a documentation of direct testing performed by the medical team on several patients.



Picture 10. Direct testing of patients

E. Evaluation

This evaluation is carried out to find out whether the systems and tools that have been made are working properly and as desired before being implemented. Evaluation objects and evaluation results can be seen in table 4.

Table 4. Prototype Evaluation

Evaluation Object	Evaluation result
Arduino nano	The microcontroller can process BPM data, sending data via Bluetooth.
Pulse Sensor	The accuracy level of the sensor reads the inspection results above 90% accurate.
Bluetooth HC-05	Bluetooth can connect and can send data to Android devices.
Module Charger	Charger module can provide sufficient power voltage.
Application	Heart rate application can connect with device, can display data and store data

V. CONCLUSION

The application of this tool can make it easier for the medical team to carry out examinations using portable tools. With this tool, it is easier for the medical team to serve patients and have a wider space to move. Tools that are made according to needs and have very small data

differences or an accuracy level of above 90%, aka almost similar.

As for the suggestions that can be put forward, namely, so that in the future this tool can be developed to be neater and simpler. Can send web-based or other online results data. Can use the sensor wirelessly. Can be combined with detecting body temperature and blood pressure.

Sudhan, RH, MG Kumar, AU Prakash, SAR Devi, and SP Arduino Atmega-328 Microcontroller, *Ijireeice*, vol. 3, no. 4, pp. 27–29, 2015.

Sulistyo, E. 2018. *Arduino-Based Pulse Detector Interfaced on a Computer*. Jakarta: SEMNASTEK, Faculty of Engineering, Muhammadiyah University.

Susilo, J. (2014). *IBII Online Jugde White Box Testing Application*. *Journal of Informatics and Business*, 3 (2), 56-6

REFERENCES

- Alfaruq, A. 2011. *Integration of Microcontroller-Based Electrocardiograph and Photoplethysmograph Monitoring Systems*. Thesis, Telkom University Bandung.
- Asrori, A. et.al. 2017. *Design of Portable Maternal Electrocardiogram Based on Blind Source Separator for Monitoring Maternal Heart Activity*. Bachelor of Physics Engineering, Faculty of Industrial Technology, Sepuluh Nopember Institute of Technology.
- Deepa, B&G, FP &. 2016. *The Yummy Marshmallow - Android 6. 0 Versions.* , 3 (2), pp. 115–117.
- Jaya, TS (2018). *Application Testing with the Blackbox Testing Method Boundary Value Analysis b (Case Study: Lampung State Polytechnic Digital Office)*. *Journal of Informatics: Journal of IT Development*, 3 (1), 45-48.
- Kadir, A. 2013. *Introduction to Visual and Interactive Approach Algorithms Using RAPTOR*. Surabaya: YOGYAKARTA ANDI.
- Lugina, RS 2015. *Learning Multimedia Design Assisted by the Explicit Instruction Method for Algorithm and Basic Programming Subjects on Sorting Materials*. Indonesian Education University, libraries.upi.edu, repository.upi.edu.
- Warp. 2014. *Bluetooth Module HC-05*. Thesis. Not published. Polytechnic Faculty. Sriwijaya State Polytechnic: Palembang Sumatra.
- Mardiansah, W. 2019. *Design of a Hospital Patient Heart Rate Monitoring Tool with a Telemetry System Based on Ardiuno UNO R3*. Padang: *Journal of Physics Unand* Vol. 8, No. 4.
- Pamungkas, AR & Husni T., (2020) *Development of Learning Media Using Codulars in Branching and Repetition Materials to Improve Student Understanding*. Thesis, University of Muhammadiyah Surakarta.
- Pressman, R. 2014. *Software Engineering*. Yogyakarta: Andi Offset.
- Pressman, RS 2015. *Software Engineering. Book Practitioner Approach I*. Yogtakarta: Andi
- Putra, RA 2019. *Heart Rate Measuring and Recording Device Based on Arduino Mega 2560 with SMS Gateway*. Essay. Electrical Engineering Education, Department of Electrical Engineering, Faculty of Engineering, Semarang State University.
- Sari, MR 2014. *Pulse Identification Through Earlobe Using Pulse Sensor*. vol. Chapter 3, p. 28, 2014.