

APLIKASI PENGUKURAN *VERTICAL JUMP* SECARA REAL TIME UNTUK ATLET OLAHRAGA

Application of Vertical Jump Measurement Real Time for Sports Athletes

Jhon Gwery Haposan Silalahi¹, Tri Nopiani Damayanti², Dery Rimasa³

^{1,2,3} Telkom University, Bandung

jhongwery@student.telkomuniversity.ac.id¹, damayanti@tass.telkomuniversity.ac.id², deryrimasa@tass.telkomuniversity.ac.id³

Abstrak

Perkembangan teknologi IOT (*internet of things*), merupakan suatu teknologi yang menghubungkan perangkat keras dengan internet sehingga dapat dikontrol dari jarak jauh dengan cara mengirimkan data dari perangkat IoT ke server. Sistem pengukuran *Vertical jump* pada Gedung KONI Bandung masih secara manual dan tidak adanya alat pengukuran secara otomatis. Maka dari itu, dibutuhkan sistem pengukuran *Vertical jump* secara *Real Time* di Gedung KONI Bandung.

Untuk mengatasi permasalahan diatas maka dibutuhkannya aplikasi pengukuran *Vertical Jump* secara *Real Time*, Perancangan alat ini akan dilakukan menggunakan NodeMCU, sensor HC-SR04, I2C, dan pembuatan aplikasi menggunakan MIT App Inventor.

Hasil Proyek Akhir sudah selesai dibuat adalah terimplementasi alat dan aplikasi pengukuran *Vertical Jump*, dan pengerjaan Proyek Akhir tersebut yaitu dapat mengatasi pengukuran *Vertical Jump* secara manual, kinerja alat dan aplikasi ini berupa tingkat akurasi pengukuran berkisar 80%. aplikasi dan alat memiliki delay, delay atau waktu tunggu terdapat 1 detik untuk aplikasi, delay 10,5 detik untuk tampilan tinggi badan dan delay 1,5 detik untuk tampilan tinggi lompatan pada alat. Tetapi secara fungsional aplikasi ini layak diimplementasikan untuk atlet Gedung KONI Bandung.

Kata Kunci: NodeMCU, HC-SR04, I2C.

Abstract

The development of IOT (internet of things) technology is a technology that connects hardware with the internet so that it can be controlled remotely by sending data from IoT devices to the server. The vertical jump measurement system at the KONI Bandung Building is still manual and there is no automatic measurement tool. Therefore, a real time vertical jump measurement system is needed at the KONI Bandung building.

To solve the above problems, a real-time vertical jump measurement application is needed. The design of this tool will be carried out using NodeMCU, HC-SR04, I2C sensors, and making applications using the MIT App Inventor.

The result of the Final Project has been completed is the implementation of the Vertical Jump measurement tool and application, and the final project is able to handle manual vertical jump measurements, the performance of this tool and application in the form of a measurement accuracy of around 80%. Applications and tools have a delay, delay or waiting time of 1 second for applications, 10.5 seconds delay for height display and 1.5 seconds delay for display of jump height on the tool. But functionally this application is feasible to be implemented for athletes in Gedung KONI Bandung.

Keywords: Light-Fidelity, Raspberry PI, Photodetector, ASCII code.

1. PENDAHULUAN

Olahraga merupakan kegiatan fisik terencana dan terstruktur melibatkan Gerakan tubuh berulang-ulang dan ditunjukan untuk kebugaran jasmani. Aktivitas ini merupakan kegiatan yang bisa dinikmati oleh semua orang untuk melatih fisik dan mental mereka. Banyak persiapan yang dilakukan untuk melakukan olahraga,

hal yang paling diutamakan adalah fisik dan kesiapan mental. Apabila kondisi orang tersebut dalam keadaan bagus, maka proses olahraga akan berjalan dengan lancar.

Dalam pengukuran fisik seorang atlet yang semakin meningkat dan terjadinya peningkatan kebutuhan alat untuk pengukuran fisik seorang atlet pada koni bandung, Sehingga mahasiswa diharapkan mampu untuk menyediakan alat yang baik agar dapat melayani kebutuhan seorang atlet pada koni bandung. Tetapi pada kondisi kali ini seorang pelatih tidak kelelahan dalam pengukuran tinggi lompatan seorang atlet. Oleh karena itu, dibutuhkan alat pengukuran *Vertical Jump* secara *Real Time*, agar mempermudah pelatih mengukur tinggi lompatan seorang atlet.

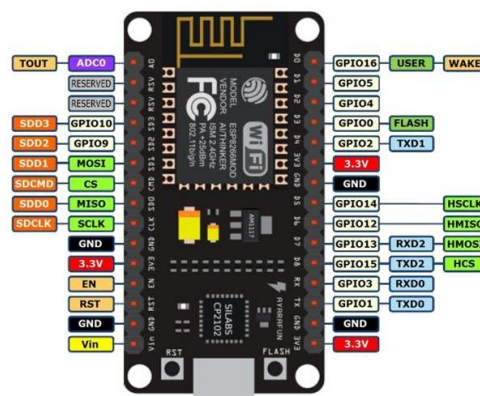
Dari penelitian wahyudi yang berjudul “Sistem pengatur jarak menggunakan sensor *Ultrasonic* purwarupa truck platooning berbasis VLC”, penelitian tersebut memanfaatkan sensor *ultrasonic* HC-SR04 untuk mengukur jarak antar truck agar bisa memperlambat truck saat mendekati truck lain di depan[1], pada kasus kali ini sensor *Ultrasonic* HC-SR04 digunakan untuk pengukuran tinggi badan dan tinggi lompatan di Gedung KONI Bandung.

Pada Proyek Akhir ini akan dilakukan perancangan sebuah alat pengukuran *Vertical Jump* secara *Real Time*, di Gedung KONI, Kota Bandung. Simulasi awal perancangan ini akan menghasilkan pengukuran *Vertical Jump* secara otomatis, Implementasi perancangan ini akan di terapkan pada Gedung KONI Bandung. Perbandingan pengukuran tinggi lompatan ini akan dilakukan dengan membedakan pengukuran secara manual dengan pengukuran secara otomatis.

2. DASAR TEORI

2.1 NodeMCU

NodeMCU terdiri dari kata node dan MCU. Dimana node dalam sistem IoT merupakan sebuah perangkat atau device yang dioperasikan menggunakan jaringan nirkabel yang dapat melakukan tugas seperti mengumpulkan data sensor dan berkomunikasi dengan base station, sedangkan MCU (Micro Chip Unit / Mikrikontroller) merupakan otak dari sistem yang digunakan untuk melakukan proses ketika diberikan input dan menghasilkan output. NodeMCU merupakan sebuah sistem berbasis mikrokontroller yang dapat mengirimkan sinyal informasi berupa data menggunakan jaringan[2]. Terdiri dari perangkat keras berupa *System On Chip* ESP8266 dari ESP8266 buatan *Espressif System*, juga firmware yang digunakan, yang menggunakan bahasa pemrograman *scripting* lua. Istilah NodeMCU secara default sebenarnya mengacu pada firmware yang digunakan dari perangkat keras development kit.[3] NodeMCU berfungsi sebagai pengendali utama perangkat lainnya dan memungkinkan sensor dapat mengirim pesan ke *smartphone* melalui internet. Pada dasarnya NodeMCU adalah pengembangan dari ESP8266 dengan firmware berbasis e-Lua. Selain Bahasa Lua, NodeMCU juga *support* dengan software Arduino IDE. Micro usb port yang terpasang pada NodeMCU digunakan dengan kabel usb untuk mentransfer barisan perintah dari Arduino IDE dan juga sebagai power supply. Sebelum digunakan, NodeMCU perlu diinstal ulang terlebih dahulu, agar NodeMCU *support* dengan *tools* yang digunakan[4]. NodeMCU yang digunakan dapat dilihat pada **Gambar 2. 1**.



Gambar 2. 1 NodeMCU [5]

Spesifikasi perangkat:

- *Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106*
- *Operating Voltage: 3.3V*
- *Input Voltage: 7-12V*
- *Digital I/O Pins (DIO): 16*
- *Analog Input Pins (ADC): 1*
- *UARTs: 1*
- *SPIs: 1*
- *I2Cs: 1*
- *Flash Memory: 4 MB*
- *SRAM: 64 KB*
- *Clock Speed: 80 MHz*
- *USB-TTL based on CP2102 is included onboard, Enabling Plug n Play*
- *PCB Antenna*
- *Small Sized module to fit smartly inside your IoT projects*

2.2 Sensor Ultrasonic HC-SR04

Sensor *Ultrasonic* merupakan sensor yang mengubah gelombang elektik menjadi sinyal listrik atau sebaliknya, sensor *ultrasonic* terdiri dari pengirim dan penerima atau bias dimaksud dengan TX dan RX yang dapat mengukur jarak dari suatu objek tanpa kehilangan keakuratannya. Prinsip kerja sensor ultrasonic ($Jarak = T \times V$) yang dimana $T =$ waktu, $V = 0.000343$ m/us. Sensor *Ultrasonic* juga bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek atau benda tertentu didepan frekuensi kerja pada daerah diatas gelombang suara dari 20kHz hingga 2MHz. [6].

Gelombang ultrasonik adalah gelombang bunyi yang mempunyai frekuensi sangat tinggi yaitu 20.000 Hz. Bunyi ultrasonik tidak dapat didengar oleh telinga manusia. Bunyi ultrasonik bias merambat melalui zat padat, cair dan gas. Reflektivitas bunyi ultrasonik di permukaan zat padat hampir sama dengan reflektivitas bunyi ultrasonik di permukaan zat cair. Akan tetapi, gelombang bunyi ultrasonik akan diserap oleh tekstil dan busa.[7]

**Gambar 2. 2 Sensor HC-SR04.[8]**

Spesifikasi perangkat:

- *Working Voltage: DC 5V*
- *Working Current: 15mA*
- *Working Frequency: 40Hz*
- *Max Range: 4m*
- *Min Range: 2cm*
- *Measuring Angle: 15 degree*
- *Trigger Input Signal: 10μS TTL pulse*
- *Echo Output Signal Input TTL lever signal and the range in proportion*
- *Dimension 45 * 20 * 15mm*

2.3 MIT App Inventor

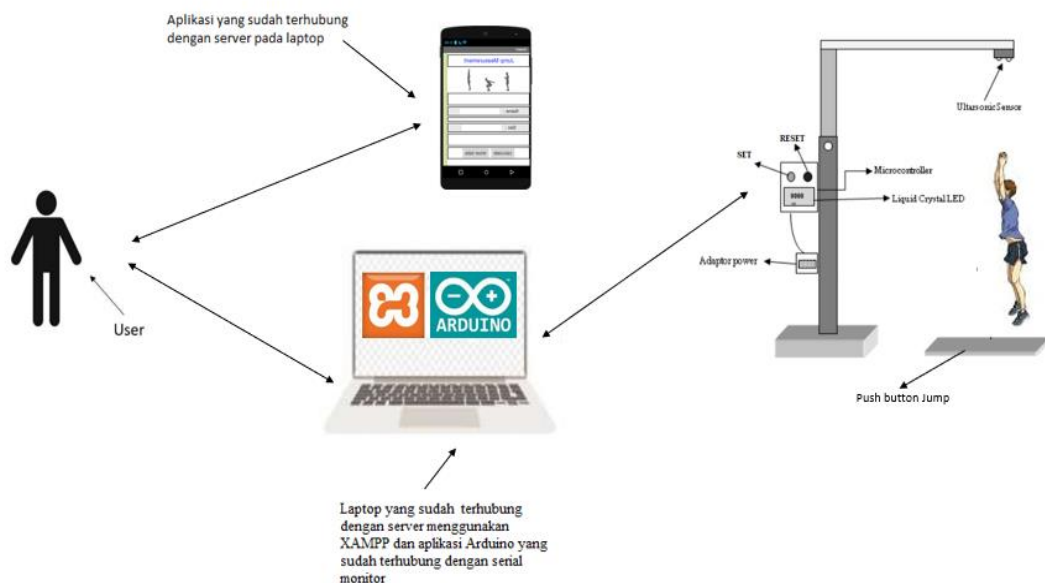
MIT App Inventor adalah lingkungan pemrograman visual yang intuitif yang memungkinkan semua orang - bahkan anak-anak - untuk membuat aplikasi yang berfungsi penuh untuk smartphone dan tablet. Mereka yang baru mengenal MIT App Inventor dapat memiliki aplikasi pertama yang sederhana dan berjalan dalam waktu kurang dari 30 menit. Terlebih lagi, alat berbasis blok kami memfasilitasi pembuatan aplikasi yang kompleks dan

berdampak tinggi dalam waktu yang jauh lebih singkat daripada lingkungan pemrograman tradisional. Proyek MIT App Inventor berupaya untuk mendemokratisasi pengembangan perangkat lunak dengan memberdayakan semua orang, terutama kaum muda, untuk beralih dari konsumsi teknologi ke penciptaan teknologi.[11]

3. MODEL SISTEM DAN PERANCANGAN

3.1 Deskripsi Proyek Akhir

Sistem yang dibuat pada sensor *ultrasonic*. Pada gambar blok diagram keseluruhan dapat dilihat pada Gambar 3.1.



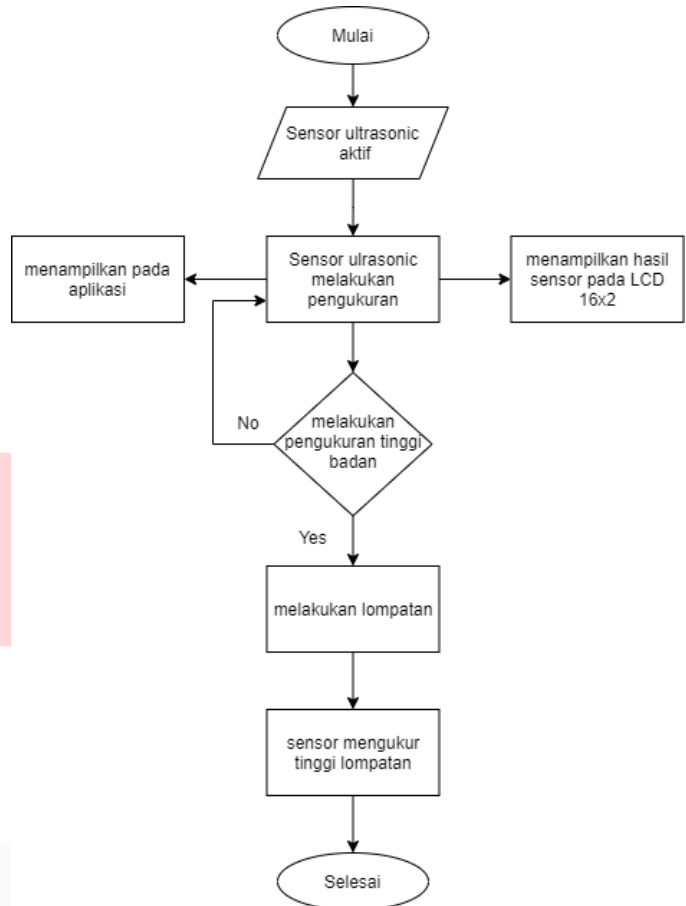
Gambar 3. 1 Blok Diagram Sistem Pengukuran Vertical Jump

Gambar 3.1 menunjukkan blok sistem keseluruhan pada Proyek Akhir terdapat sensor *ultrasonic* yang dimana saat atlet ingin mengukur tinggi lompatan, terlebih dahulu pelatih mengisi kolom *name* dan *sex* untuk menyimpan data hasil pengukuran. Setelah mengisi kolom pelatih dapat menekan button (*calculate*), sebelum mengukur tinggi atlet pelatih terlebih dahulu menekan tombol *push button (Reset)*, setelah atlet siap mengukur tinggi badan maka pelatih dapat menekah tombol *push button (Set)*, untuk mengukur tinggi lompatan terlebih dahulu pelatih menekan tombol *push button (trigger kaki)* setelah ada perintah dari serial monitor “silahkan melompat” maka atlet dapat melompat, setelah setelah melompat pelatih menekan kembali tombol *push button (trigger kaki)* agar dapat melihat hasil tinggi lompatan.

Server file *test.php* data pengukuran *vertical jump* dikirimkan ke file *data.json*, file *data.json* akan menerima data terlebih dahulu dengan cara melakukan *request get* menggunakan url yang berisi ip server dan data file.php pada server. Aplikasi akan membaca hasil pengukuran *vertical jump* pada file *data.json* dan akan di tampilkan pada aplikasi.

3.2 Perancangan Aplikasi

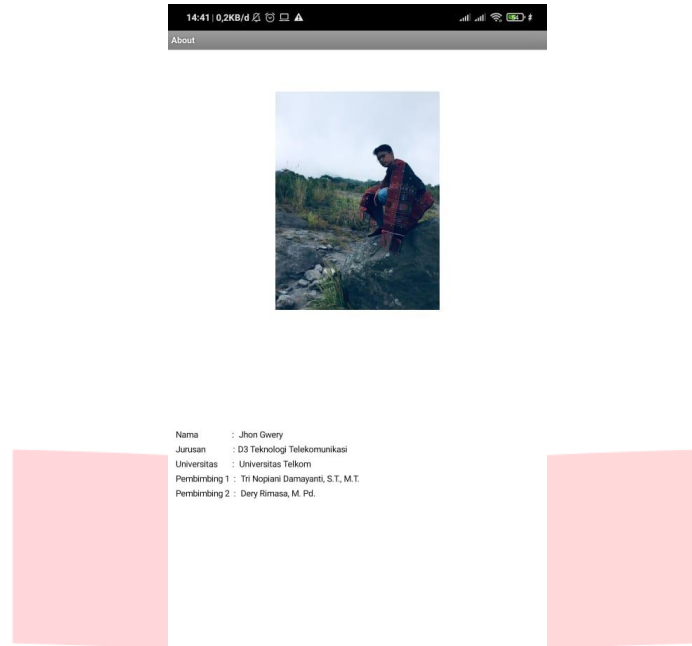
Dalam perancangan aplikasi pengukuran *Vertical Jump* ini menggunakan MIT App Inventor, berikut adalah *flowchart* dari sistem aplikasi :



Gambar 3. 2 flowchart aplikasi

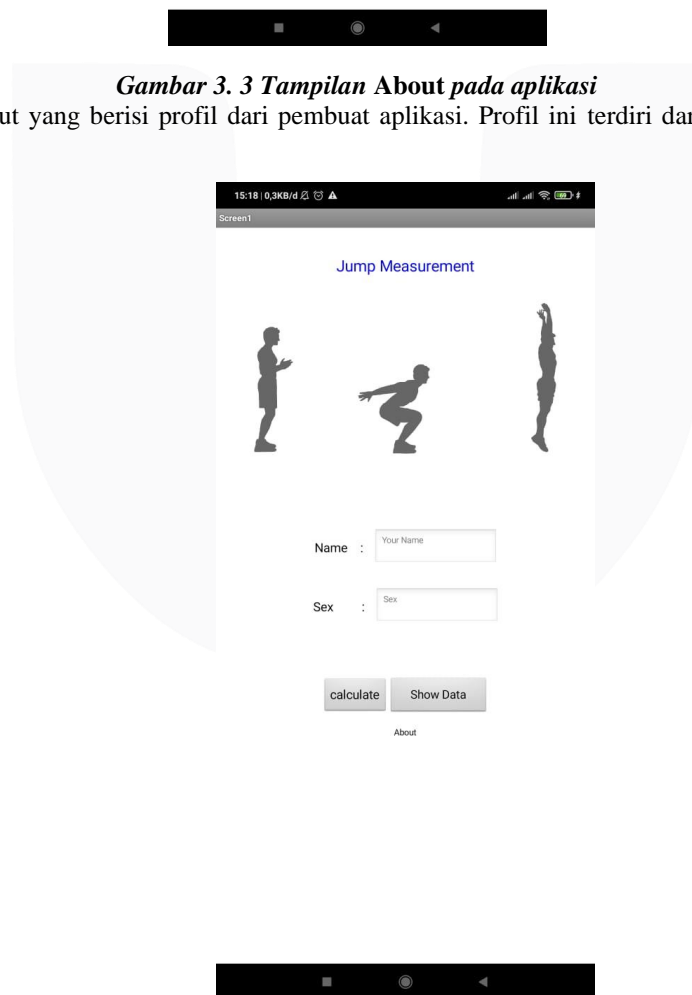
Gambar 3.2 ini menjelaskan urutan dari pengukuran *Vertical Jump* pada atlet. Sistem ini menggunakan sensor *ultrasonic* yang sudah terhubung dengan aplikasi sehingga hasil yang di dapatkan secara *real time*. Pertama sensor *ultrasonic* melakukan pengukuran tinggi badan kemudian menampilkan hasil di LCD dan Aplikasi. Setelah itu sensor *ultrasonic* akan mengukur tinggi lompatan pada atlet dan menampilkan hasil pada LCD dan Aplikasi. Setelah itu, hasil kedua pengukuran akan ditampilkan pada Aplikasi.

Aplikasi mempunyai beberapa tampilan screen sebagai berikut :



Gambar 3. 3 Tampilan About pada aplikasi

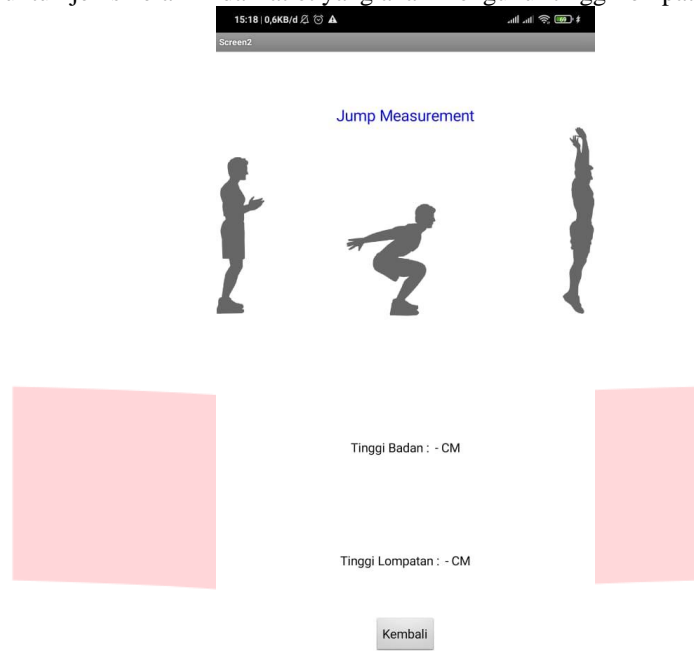
Tampilan about yang berisi profil dari pembuat aplikasi. Profil ini terdiri dari Nama, Nim, Prodi dan Pembimbing 1&2.



Gambar 3. 4 Tampilan Aplikasi Screen 1

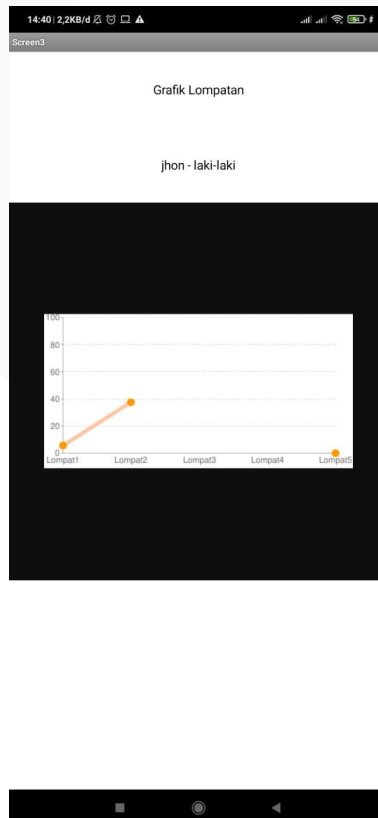
Tampilan screen 1 untuk terdapat:

- Name : kolom untuk mengisi nama atlet yang akan mengukur tinggi lompatan
- Sex : kolom untuk jenis kelamin dari atlet yang akan mengukur tinggi lompatan.



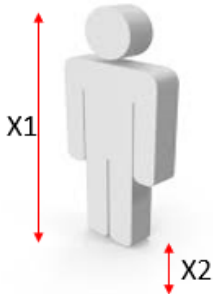
Gambar 3.5 Tampilan Aplikasi Screen 2

Tampilan screen 2 untuk menampilkan angka dari tinggi badan dan tinggi lompatan dari atlet yang akan mengukur tinggi lompatan



Gambar 3. 6 Tampilan Aplikasi Screen 3

Tampilan screen 3 berupa grafik data peningkatan hasil lompatan atlet berdasarkan nama perorang

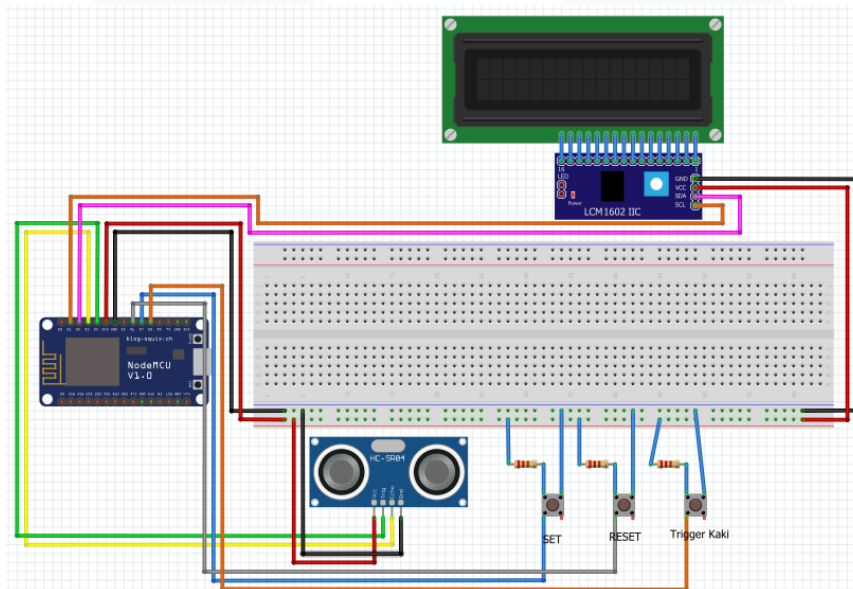
BAGIAN TUBUH YANG DI UKUR	VARIABEL	NAMA
	X1	TINGGI BADAN
	X2	TINGGI LOMPATAN

Gambar 3. 7 Bagian pengukuran vertical jump

Pada gambar 3.7 menjelaskan bagian mana saja yang akan dilakukan pengukuran *Vertical Jump* pada atlet. Dengan memulai pengukuran tinggi badan menggunakan sensor ultrasonik HC-SR04, maka dapat dilakukan pengukuran tinggi lompatan agar mendapatkan nilai tinggi badan dan tinggi lompatana atlet.

3.3 Perancangan Hardware

Sistem perancangan *Hardware* pengukuran *Vertical Jump* sistem ini menggunakan NodeMCU dan sensor *ultrasonic*, berikut adalah block diagram *hardware*:



Gambar 3. 8 Block diagram perancangan Hardware

Pada blok diagram diatas menjelaskan bagaimana perancangan sistem *Hardware* bekerja. Sensor *ultrasonic* digunakan untuk mendeteksi tinggi badan dan tinggi lompatan pada atlet, *Push button* set digunakan untuk proses pengukuran tinggi badan atlet, *push button* reset digunakan untuk melakukan reset data saat ingin melakukan pengukuran, *push button* trigger kaki digunakan pada saat melakukan pengukuran tinggi lompatan atlet, LCD I2C digunakan untuk menampilkan kata hasil pengukuran tinggi badan dan tinggi lompatan, Sistem tersebut menggunakan software Arduino UNO untuk mengupload *coding* pada NodeMCU, agar sensor dapat bekerja.

Tabel 3. 1 Pin Kaki sensor ultrasonik pada NodeMCU

NodeMCU	Sensor ultrasonik HC-SR04
D3	Echo
D4	Trig
GND	GND
VCC	VCC

Tabel 3. 2 Pin kaki LCD I2C pada NodeMCU

Nodemcu	LCD 16x2 I2C
D2	Sda
D1	Scl
GND	Gnd
VCC	VCC

Tabel 3. 3 Pin Kaki Push Button pada NodeMCU

Push button	NodeMCU
Set	D7
Reset	D6
Trigger kaki	D8

Perangkat keras yang digunakan untuk mendukung dalam pengukuran *Vertical Jump* adalah sebagai berikut:

1. Laptop
Spesifikasi : menggunakan laptop yang dapat *support* aplikasi Arduino Uno.
2. Sensor *ultrasonic* HC-SR04



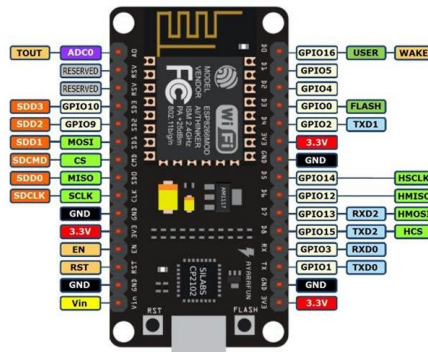
Gambar 3. 9 Sensor Ultrasonic HC-SR04

Spesifikasi:

- a. *Working Voltage: DC 5V*
- b. *Working Current: 15mA*
- c. *Working Frequency: 40Hz*
- d. *Max Range: 4m*
- e. *Min Range: 2cm*
- f. *Measuring Angle: 15 degree*
- g. *Trigger Input Signal: 10µS TTL pulse*

- h. *Echo Output Signal Input TTL lever signal and the range in proportion*
- i. *Dimension 45 * 20 * 15mm*

3. NodeMCU 8266



Gambar 3. 10 NodeMCU 8266

Spesifikasi :

- a. *Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106*
 - b. *Operating Voltage: 3.3V*
 - c. *Input Voltage: 7-12V*
 - d. *Digital I/O Pins (DIO): 16*
 - e. *Analog Input Pins (ADC): 1*
 - f. *UARTs: 1*
 - g. *SPIs: 1*
 - h. *I2Cs: 1*
 - i. *Flash Memory: 4 MB*
 - j. *SRAM: 64 KB*
 - k. *Clock Speed: 80 MHz*
 - l. *USB-TTL based on CP2102 is included onboard, Enabling Plug n Play*
 - m. *PCB Antenna*
 - n. *Small Sized module to fit smartly inside your IoT projects*
- ### 4. LCD I2C 16x2



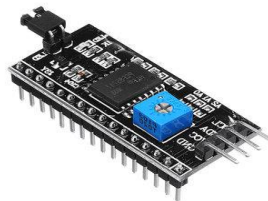
Gambar 3. 11 LCD 16x2

Spesifikasi :

- a. *VSS(Ground) | Ground pin connected to system ground.*
- b. *VDD(+5 Vlot) | Powers the LCD with +5V (4.7V – 5.3V).*
- c. *VE (Contrast V) | Decides the contrast level of display. Grounded to get maximum contrast.*
- d. *Register Select | Connected to Microcontroller to shift between command/data register.*
- e. *Read/Write | Used to read or write data. Normally grounded to write data to LCD.*
- f. *Enable | Connected to Microcontroller Pin and toggled between 1 and 0 for data acknowledgement.*
- g. *Data Pin 0 | Data pins 0 to 7 forms a 8-bit data line. They can be connected to Microcontroller to send 8-bit data.*
- h. *Data pin 1.*
- i. *Data pin 2*

- j. Data pin 3
- k. Data pin 4
- l. Data pin 5
- m. Data pin 6
- n. Data pin 7
- o. LED Positive / Backlight LED pin positive terminal.
- p. LED Negative / Backlight LED pin negative terminal.

5. I2C



Gambar 3. 12 I2C

Spesifikasi:

- a. I2C Address Range / 0x20 to 0x27 (Default=0x27, addressable)
- b. Operating Voltage / 5 Vdc
- c. Backlight / White
- d. Contrast / Adjustable by potentiometer on I2c interface
- e. Size / 80mm x 36mm x 20 mm
- f. Viewable area / 66mm x 16mm

6. Tactile Switches



Gambar 3. 13 Tactile Switches

Spesifikasi:

- a. Operating Temperature Range / -20 ~ +70
- b. Storage Temperature Range / -40 ~ +85
- c. Type of Operation / Tactile Feedback
- d. Circuit Configuration / Push-On Momentary, 1 pole-1 throw
- e. Power Rating / MAX 50mA 24VDC
- f. Contact Resistance / MAX 100
- g. Insulation Resistance | 100MΩ Min. at 100VDC
- h. Dielectric Withstanding Voltage / 250VAC for 1 minute
- i. Contact Bounce | MAX 5
- j. Operating Force / 0.98±0.49 / 1.57±0.49
- k. Return Force | MIN 0.10 / MIN 0.49
- l. Travel / 0.25 +0.2/-0.1

3.4 Hasil Perancangan sistem

Perancangan sistem *software* dan *hardware* pengukuran *vertical Jump* secara *real time*. Aplikasi yang digunakan untuk menampilkan hasil dari pengukuran tinggi badan dan tinggi lompatan berupa grafik, hardware akan menampilkan hasil tinggi lompatan dan tinggi badan pada LCD 16x2 I2C.

3.4.1 Foto aplikasi yang terhubung pada *hardware*



Gambar 3. 14 Software dan hardware

Gambar merupakan tampilan aplikasi dan LCD 16x2 dalam melakukan pengukuran tinggi badan dan tinggi lompatan atlet. Pada tampilan hardware terdapat *push button* set reset dan trigger kaki, *push button* reset digunakan untuk mereset data sebelum melakukan pengukuran, *push button* digunakan untuk melakukan pengukuran tinggi badan, *push button* trigger kaki digunakan saat melakukan pengukuran tinggi lompatan atlet.



Gambar 3. 15 Alat pengukuran Vertical jump

Gambar 3.15 merupakan tampilan alat pengukuran *vertical Jump* secara *real time*, pada tampilan terdapat sensor *ultrasonic*, LCD 16x2 dan *push button set, reset, trigger* kaki.



Gambar 3. 16 Pengukuran tinggi lompatan

Gambar 3.16 merupakan alat dan aplikasi saat pengukuran tinggi lompatan dari atlet, nilai tinggi badan dan tinggi lompatan terdapat pada aplikasi.

calculate

Gambar 3. 17 Tampilan Push Button 1 pada screen 1

Pada gambar 3.17 merupakan *push button* 1 pada *screen* 1 yang berfungsi untuk menampilkan hasil pengukuran tinggi badan dan tinggi lompatan pada *screen* 2.

Show Data

Gambar 3. 18 Tampilan push button 2 pada screen 1

Pada gambar 3.18 merupakan *push button* 2 pada *screen* 1 yang berfungsi untuk menampilkan hasil pengukuran tinggi lompatan berupa grafik pada *screen* 3.

About

Gambar 3. 19 Tampilan push button 3 pada screen 1

Pada gambar 3.19 merupakan tampilan *push button* 3 pada *screen* 1 yang berfungsi untuk menampilkan data diri dari pembuat aplikasi.

4. ANALISIS SIMULASI PENGUJIAN

4.1 Hasil Pengujian Aplikasi *Vertical Jump*

Hasil akhir dari perancangan aplikasi ini berupa hasil tinggi badan, tinggi lompatan dan pengimpunan nama atlet. Aplikasi ini dapat digunakan di *smartphone* android dan ios. Pengujian dilakukan pada *Smartphone Android* dengan spesifikasi berikut :

- *Smartphone* : Xiaomi Redmi Note 8 Pro
- Versi *Android* : *Android* 10
- Ram : 6 GB
- CPU : Octa-core Max 2.05GHz

4.2 Skenario Pengujian

Skenario pengujian dilakukan dengan cara menguji dari parameter tinggi badan atlet yaitu mengukur tinggi badan atlet terlebih dahulu dengan sensor *ultrasonic* HC-SR04 atlet berdiri tepat diatas tanda yang sudah di buat, lalu hasil pengukuran akan muncul pada LCD 12x6 dan di aplikasi pada *Screen*2. Setelah selesai pengukuran tinggi badan maka dilakukan tahap kedua dengan mengukur tinggi lompatan atlet, atlet akan menekan *push button* dengan kaki setelah ada pemberitahuan dilayar monitor “Silahkan Melompat” maka saatnya atlet melompat setelah atlet selesai melompat maka tekan kembali *push button*nya. Sebelum melakukan pengukuran secara otomatis pada sensor *ultrasonic* HC-SR04, perlu dilakukan pengukuran secara manual menggunakan seperti rol meter. Setelah mendapat kedua hasil pengukuran, maka akan dilakukan perbandingan pengukuran yang didapatkan oleh sensor *ultrasonic* HC-SR04 dan pengukuran yang dilakukan secara manual dengan alat – alat pengukur. Data hasil pengukuran secara otomatis akan tersimpan pada aplikasi, dan hasil dapat berupa grafik untuk melihat peningkatan dari atlet tersebut.

4.2.1 Metode pengukuran secara manual yang baik dan benar



Gambar 4. 1 pengukuran secara manual yang baik dan benar

- a. Atlet bersiap-siap di tempat test dengan memberikan tanda pada jari-jari tangan yang akan digunakan untuk mencapai Raihan.
- b. Atlet menjulurkan lengan ke tempat ukuran sebagai Raihan awal.
- c. Atlet kemudian melakukan lompatan ke atas tanpa awalan dengan menjulurkan dan menempelkan jari tangan setinggi mungkin.
- d. Atlet mencapai hasil Raihan dan waktu lompat saat lepas landas sampai dengan jarak Raihan.
- e. Pelatih mencatat hasil Raihan atlet dan kecepatan waktu lompat.

4.2.2 Metode pengukuran secara otomatis yang baik dan benar



Gambar 4. 2 pengukuran secara otomatis dengan baik dan benar

- a. Atlet berdiri tepat di tanda yang diberikan dibawah sensor *ultrasonic*.
- b. Pelatih melakukan pengukuran tinggi badan atlet dengan menekan *push button (set)*.
- c. Setelah itu atlet dapat melompat setelah dapat arahan dari pelatih, sebelum melakukan lompatan atlet terlebih dulu menginjak *push button* setelah dapat arahan melompat dari pelatih maka atlet dapat melompat dan menginjak Kembali *push button* agar hasil lompatan dapat di lihat.
- d. Hasil pengukuran tinggi badan dan tinggi lompatan dapat di lihat pada aplikasi dan dapat dilihat grafik lompatan setiap atlet.

4.3 Hasil Pengukuran *Vertical Jump* secara manual dan otomatis

Sub bab ini membahas tentang pengukuran tinggi lompatan yang akan ditampilkan adalah data pengukuran dengan 2 cara yang pertama adalah menggunakan alat yang sudah dibuat dan sudah tersambung menggunakan aplikasi.

Hasil pengukuran tinggi lompatan dengan menggunakan alat yang tersambung ke aplikasi, dan membandingkan pengukurannya dengan pengukuran secara manual menggunakan rol meter sebagai alat pengukur. Hasil yang akan ditunjukkan pada aplikasi akan menunjukkan tingkat akurasi aplikasi dan alat dengan cara membandingkan dengan hasil secara manual.

4.2.1 Hasil pengukuran *Vertical Jump* secara manual

Pengukuran secara manual ini menggunakan menteran, berikut adalah hasil dari 3 orang atlet yang pengukuran secara manual dapat di lihat pada **tabel 4.1**

Tabel 4. 1. Tabel hasil pengukuran secara manual

Nama	Manual			Otomatis			Persentasi kesalahan aplikasi
	Tinggi badan	tinggi lompatan	Hasil lompatan	Tinggi badan	tinggi lompatan	Hasil lompatan	
Jhon	220	265	45	179,33	46,63	46,63	0,036%
	220	260	40	177,44	44,8	44,8	0,120%
	220	265	45	172,95	46,65	46,65	0,037%
	220	263	43	182,74	44,78	44,78	0,041%
	220	258	38	181,37	39,15	39,15	0,030%
Jandri	212	255	43	178,43	48,49	48,49	0,128%
	212	256	44	181,13	43,99	43,99	0%
	212	255	43	178,39	43,76	43,76	0,018%
	212	250	38	171,05	46,95	46,95	0,236%
	212	255	43	185,77	45,52	45,52	0,059%
Regi	202	233	31	169,23	35,05	35,05	0,131%
	202	230	28	170,42	29,35	29,35	0,048%
	202	234	32	176,61	33,47	33,47	0,046%
	202	233	31	179,63	30,9	30,9	-0,003%
	202	236	34	177,3	34,68	34,68	0,020%

Tabel 4.1 menunjukkan hasil pengukuran vertical jump secara manual dan otomatis, berdasarkan hasil pengukuran terhadap tiga atlet dapat dilihat bahwa terdapat perbedaan antara hasil pengukuran vertical jump secara manual dan otomatis. Dari tiga atlet diatas diperoleh rata-rata persentase kesalahan aplikasi pada atlet jhon 0,264%, atlet jandri 0,088%, atlet regi 0,048%. Rata persentasi kesalahan aplikasi yaitu 0,063%.

4.2.2 Hasil pengukuran *Vertical Jump* secara otomatis

Pengukuran secara otomatis menggunakan aplikasi yang sudah terhubung dengan alat *microcontroller*. Aplikasi ini digunakan pada *smartpone xiaomi* ddengan versi *android 10*, Hasil pengukuran secara otomatis dapat di lihat pada **tabel 4.2**

Tabel 4. 2 Pengukuran Vetical jump secara otomatis

Pengukuran vertical Jump secara otomatis		
Nama	TB	TL
Jhon	182.33	46,63
	167.44	44,8

	172.95	46,65
	182.74	44,78
	181.37	39,15
Jandri	178,43	48,49
	181,13	43,99
	178,39	43,76
	171.05	46,95
	185,77	45,52
Regi	169.23	35,05
	170,42	29,35
	176.61	33,47
	179,63	30,9
	177,3	34,68

Tabel 4.1 menunjukkan hasil pengukuran vertical jump secara manual dan otomatis, berdasarkan hasil pengukuran terhadap tiga atlet dapat dilihat bahwa terdapat perbedaan antara hasil pengukuran vertical jump secara manual dan otomatis. Dari tiga atlet di atas diperoleh rata-rata persentase kesalahan aplikasi pada atlet jhon 0,264%, atlet jandri 0,088%, atlet regi 0,048%. Rata persentasi kesalahan aplikasi yaitu 0,063%.

4.4 Pengujian *delay* pada aplikasi dan alat *microcontroller*

Delay merupakan parameter pengukuran keterlambatan waktu dari sisi pengirim (TX) ke sisi penerima (RX), skenario delay ini dibutuhkan agar dapat melihat seberapa lama dilai yang didapatkan pada *software* dan *hardware*, hasil pengukuran *delay* pada aplikasi dapat dilihat pada **tabel 4.3**

Tabel 4. 3 Pengujian *delay* button pada aplikasi

Pengukuran <i>delay</i> pada aplikasi						
No	Delay (s)					Rata-Rata
	1	2	3	4	5	
home	0,97	0,91	0,63	1,02	0,9	0,886
Calculate	0,81	0,79	0,74	0,72	0,6	0,732
Show Data	0,47	0,34	0,32	0,47	0,37	0,394
Grafik	0,39	0,58	0,75	0,75	0,73	0,640
About	0,98	0,55	0,56	0,59	0,38	0,612

Tabel 4.3 menunjukkan hasil pengukuran *delay* pada button aplikasi, berdasarkan hasil pengujian dapat disimpulkan hasil rata-rata dari setiap button tidak mencapai satu detik.

Tabel 4. 4 pengujian *delay* pada LCD

Pengukuran <i>delay</i> pada LCD						
No	Delay (s)					Rata-Rata
	1	2	3	4	5	
Tinggi badan	10,9	10,56	9,91	10,12	10,54	10,406
Tinggi Lompatan	1	1,54	0,55	0,65	1,49	1,046

Tabel 4.4 menunjukkan hasil pengukuran *delay* pada LCD alat *microcontroller*, berdasarkan hasil pengujian dapat disimpulkan hasil rata-rata dari setiap tampilan tinggi badan di bawah 10,5 detik dan rata-rata tinggi lompatan di bawah 1,5 detik.

5. PENUTUP

5.1 Kesimpulan

Tugas Akhir ini berhasil melakukan membuat aplikasi dan alat untuk pengukuran *Vertical Jump*. Berdasarkan hasil perancangan, pengujian dan analisa yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Proyek Akhir ini telah berhasil membuat pengukuran *Vertical Jump* secara *Real Time*, di Gedung KONI, Kota Bandung.
2. Berdasarkan hasil pengujian fungsionalitas terhadap fitur-fitur yang ada pada aplikasi android yang telah dibuat, dapat disimpulkan bahwa semua fitur dapat berjalan dengan baik.
3. Dari hasil pengujian alat *Microcontroller* dan aplikasi tingkat akurasi aplikasi jika dibandingkan dengan pengukuran manual adalah 80%.
4. Dari hasil pengujian aplikasi tingkat *delay* atau waktu tunggu yang terdapat pada aplikasi pengukuran tinggi lompatan *vertical* adalah di bawah 1 detik, dapat disimpulkan waktu *delay* aplikasi sudah baik secara fungsional.
5. Dari hasil pengujian alat *microcontroller* tingkat *delay* atau waktu tunggu yang terdapat pada aplikasi pengukuran tinggi lompatan *vertical* pada tampilan tinggi badan adalah di bawah 10,5 detik dan tampilan pada tinggi lompatan *vertical* dibawah 1,5 detik, dapat disimpulkan waktu *delay* alat ini sudah baik secara fungsional.

5.2 Saran

Berdasarkan hasil pembangunan Proyek Akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya yaitu :

1. Membuat tampilan dalam aplikasi menjadi lebih menarik dan lebih baik lagi.
2. Menambahkan fitur penyimpanan data pengukuran perhari,perbulan,pertahun.
3. Menggunakan sensor maxbotix.

REFRENSI

- [1] Poole, Cellular Communication Explained F[1] D. Darlis, S. Si, and R. A. P, "SISTEM PENGATUR JARAK MENGGUNAKAN SENSOR ULTRASONIC PADA PURWARUPA TRUCK PLATOONING BERBASIS VLC Distance Control System Using Ultrasonic Sensor On VLC-Based Truck Platooning Prototype," vol. 6, no. 1, pp. 737–750, 2020.
- [2] M. R. Hidayat, C. Christiono, and B. S. Sapudin, "PERANCANGAN SISTEM KEAMANAN RUMAH BERBASIS IoT DENGAN NodeMCU ESP8266 MENGGUNAKAN SENSOR PIR HC-SR501 DAN SENSOR SMOKE DETECTOR," *Kilat*, vol. 7, no. 2, pp. 139–148, 2018, doi: 10.33322/kilat.v7i2.357.
- [3] A. Škraba, A. Koložvari, D. Kofjač, R. Stojanović, V. Stanovov, and E. Semenkin, "Prototype of group heart rate monitoring with NODEMCU ESP8266," *2017 6th Mediterr. Conf. Embed. Comput. MECO 2017 - Incl. ECYPS 2017, Proc.*, no. November 2018, 2017, doi: 10.1109/MECO.2017.7977151.
- [4] L. Goswami, M. K. Kaushik, R. Sikka, V. Anand, K. Prasad Sharma, and M. Singh Solanki, "IOT Based Fault Detection of Underground Cables through Node MCU Module," *2020 Int. Conf. Comput. Sci. Eng. Appl. ICCSEA 2020*, 2020, doi: 10.1109/ICCSEA49143.2020.9132893.
- [5] ESP8266 Datasheet, "ESP8266EX Datasheet," *Espr. Syst. Datasheet*, pp. 1–31, 2015.
- [6] M. Kaur, J. Pal, and M. Tech Student, "Distance Measurement of Object by Ultrasonic Sensor HC-SR04," *IJSRD-International J. Sci. Res. Dev.*, vol. 3, no. 05, pp. 2321–0613, 2015.
- [7] Indoware, "Ultrasonic Ranging Module HC - SR04," *Datasheet*, pp. 1–4, 2013.
- [8] E. J. Morgan, "HC SR04 Ultrasonic Ranging Sensor Module," *Eval. Tec. Sens.*, p. Nov. 16 2014, 2014.
- [9] T. Specification, "SRF04 - Ultra-Sonic Ranger," *SRF04 Ultrason. Ranger*, pp. 1–6, 2012.
- [10] D. H. Trihantoro, D. Darlis, and H. Putri, "Implementasi Visible Light Communication (Vlc)

Untuk Pengiriman Teks,” no. VIc, pp. 1–5, 2017, doi: 10.31227/osf.io/78rpn.

[11] “About Us.” [Online]. Available: <http://appinventor.mit.edu/about-us>. [Accessed: 23-Jan-2021].

