



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**RANCANG BANGUN PROTOTYPE SISTEM PERANGKAP  
SATWA LIAR DI PEMUKIMAN PENDUDUK BERBASIS  
ANDROID**

*“Rancang Bangun Aplikasi Android Sistem Perangkap Satwa Liar  
di Pemukiman Penduduk”*

**TUGAS AKHIR**

**POLITEKNIK  
NEGERI  
JAKARTA**  
**RIZKA DWI SAKHA**  
**2103332090**

**PROGRAM STUDI TELEKOMUNIKASI**

**JURUSAN TEKNIK ELEKTRO**

**POLITEKNIK NEGERI JAKARTA**

**2024**



## HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir ini adalah hasil karya saya sendiri dan semua sumber baik yang kutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Rizka Dwi Sakha

NIM : 2103332090

Tanda Tangan :



RIZKA

Tanggal : Jumat, 30 Agustus 2023



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

HALAMAN PENGESAHAN  
TUGAS AKHIR

LEMBAR PENGESAHAN  
TUGAS AKHIR

Tugas Akhir diajukan oleh :

Nama : Rizka Dwi Sakha  
NIM : 2103132090  
Program Studi : Telekomunikasi  
Judul Tugas Akhir : Rancang Bangun Aplikasi Android Sistem Perangkap Satwa Liar di Pemukiman Penduduk

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada Jumat, 9 Agustus 2024 dan dinyatakan LULUS

Pembimbing I :

Rifqi Fuadi Hasani, S.T., M.T.  
NIP. 19920818 201903 1 015

POLITEKNIK  
NEGERI  
JAKARTA

Depok, .....

Disahkan Oleh

Ketua Jurusan Teknik Elektro



Dr. Nurul Dwiyanti, S.T., M.T.  
NIP. 19780531 200312 2 002

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan satu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





## KATA PENGANTAR

Puji Syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena berkat dan Rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir ini. Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Diploma Tiga Politeknik.

Tugas Akhir ini berjudul “Rancang Bangun Prototype Sistem Perangkat Satwa Liar di Pemukiman Penduduk Berbasis Arduino” guna membantu para penduduk mengamankan Kawasan pemukimannya atau pun perkebunannya yang sering kali di rusak oleh satwa liar.

Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tugas akhir ini, sangatlah sulit bagi penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Rifqi Fuadi Hasani., S.T., M.T., selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan tugas akhir ini;
2. Benny Nixon, S.T., M.T selaku Kepala Program Studi Telekomunikasi;
3. Orang tua dan keluarga penulis yang telah memberikan bantuan dukungan material dan moral;
4. Nugrahani Wanda Oktaviana yang telah menjadi rekan penulis serta membantu menyelesaikan penyusunan Tugas Akhir; dan
5. Sahabat yang telah banyak membantu penulis dalam menyelesaikan tugas akhir ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga Tugas Akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, Agustus 2024

Rizka Dwi Sakha



## “Rancang Bangun Aplikasi Android Sistem Perangkap Satwa Liar di Pemukiman Penduduk”

### Abstrak

*Perangkap satwa merupakan sebuah perangkat yang dirancang secara khusus untuk menangkap satwa liar yang memasuki pemukiman penduduk, agar tidak terjadi konflik antara manusia dengan satwa liar. Perangkat satwa pada umumnya masih mempunyai kekurangan dalam kendali dan pemantauan secara real-time. Maka dari itu sistem kendali dan pemantauan satwa dirancang dengan memanfaatkan internet. Metode yang digunakan untuk mengirimkan perintah buka atau tutup perangkat adalah real-time dan pemantauan satwa menggunakan sensor ultrasonik yang terhubung langsung pada perangkat jaringan dengan menggunakan aplikasi Kodular untuk menampilkan sistem pemantauan dan kendali jarak jauh. Hasil pengujian menunjukkan jika terdapat satwa yang terperangkap maka status pemantauan akan tertera di menu pemantauan dan kendali. Pengukuran Quality of Service dilakukan pada tiga waktu berbeda: pagi, siang, dan malam hari. Hasil pengukuran menunjukkan performa optimal pada hari pertama pagi hari dengan throughput 1,474 Kbps, packet loss 0%, dan delay 5.75 ms. Pada siang hari, performa jaringan menurun dengan throughput 170 Kbps, packet loss tetap 0%, dan delay meningkat menjadi 32.11 ms. Pada malam hari, throughput mencapai titik terendah 50 Kbps, packet loss tetap 0%, dan delay meningkat hingga 69.12 ms. Selain itu, pengujian menggunakan aplikasi Speedtest menunjukkan kecepatan download sebesar 26 Mbps dan upload sebesar 16.3 Mbps.*

**Kata kunci:** Aplikasi Android, Kodular, Firebase, Satwa Liar, Mikrokontroler

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

*“Android App Design of Wildlife Trapping System in Residential Areas”*

**Abstract**

*A wildlife trap is a trap specifically designed to capture wildlife that enters a residential area, to prevent conflict between humans and wildlife. Animal traps in general still have shortcomings in real-time control and monitoring. Therefore, the animal control and monitoring system is designed by utilizing the internet. The method used to send commands to open or close the trap is real-time and animal monitoring uses ultrasonic sensors that are directly connected to network devices using Kodular applications to display remote monitoring and control systems. The test results show that if there are trapped animals, the monitoring status will be listed in the monitoring and control menu. Quality of Service measurements were taken at three different times: morning, afternoon, and evening. The measurement results show optimal performance on the morning with a throughput of 1,474 Kbps, packet loss of 0%, and delay of 5.75 ms. During the day, the network performance decreased with a throughput of 170 Kbps, packet loss remained 0%, and delay increased to 32.11 ms. At night, throughput reached its lowest point of 50 Kbps, packet loss remained 0%, and delay increased to 69.12 ms. In addition, testing using the Speedtest application application showed a download speed of 26 Mbps and an upload speed of 16.3 Mbps.*

**Keywords:** *Android Application, Kodular, Firebase, Wildlife, Microcontroller*



## DAFTAR ISI

HALAMAN SAMPUL.....	1
HALAMAN JUDUL .....	ii
HALAMAN PERNYATAAN ORISINALITAS.....	iii
HALAMAN PENGESAHAN TUGAS AKHIR.....	iv
KATA PENGANTAR.....	v
ABSTRAK .....	vi
ABSTRACT .....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL .....	x
DAFTAR GAMBAR.....	xi
DAFTAR LAMPIRAN .....	xiii
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan .....	2
1.4 Luaran .....	2
<b>BAB II TINJAU PUSTAKA.....</b>	<b>3</b>
2.1 <i>Internet of Things</i> .....	3
2.2 Internet .....	4
2.3 Android .....	5
2.4 Kodular .....	5
2.5 Firebase .....	11
2.6 Wireshark .....	12
2.7 <i>Hypertext Transfer Protocol (HTTP)</i> .....	12
2.8 <i>Quality of Service</i> .....	12
<b>BAB III PERENCANAAN DAN REALISASI.....</b>	<b>15</b>
3.1 Rancang Alat.....	15
3.1.1 Deskripsi Alat .....	15
3.1.2 Cara Kerja Alat .....	16
3.1.3 Spesifikasi Alat.....	17
3.1.4 Diagram Blok .....	17
3.2 Realisasi Alat .....	18
3.2.1. Realisasi Pembuatan <i>Database</i> .....	18
3.2.2. Penyambungan <i>Database</i> ke Program Aplikasi .....	22
3.2.3. Perancangan Aplikasi Android .....	25
3.2.4. Realisasi Program Aplikasi Android .....	26
3.2.5. Penyambungan Alat ke Database Firebase.....	38
<b>BAB IV PEMBAHASAN.....</b>	<b>41</b>
4.1 Pengujian Aplikasi Android .....	41
4.1.1. Deskripsi Pengujian Aplikasi Android .....	41
4.1.2. Prosedur Pengujian .....	42
4.1.3. Analisa Data .....	47
4.2 Pengujian <i>Quality of Service</i> .....	47
4.2.1. Deskripsi Pengujian.....	47
4.2.2. Prosedur Pengujian .....	48

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2.3. Hasil Pengujian <i>Quality of Service</i> .....	49
4.2.4. Analisa Data .....	67
4.3 Pengujian Internet Menggunakan Speedtest.....	68
4.3.1 Deskripsi Pengujian .....	68
4.3.3 Prosedur Pengujian .....	69
4.3.4 Data Hasil Pengujian .....	70
4.3.5 Analisa Data.....	70
4.4 Analisa Sistem .....	71
<b>BAB V PENUTUP .....</b>	<b>72</b>
5.1 Kesimpulan .....	72
5.2 Saran .....	72
<b>DAFTAR PUSTAKA.....</b>	<b>73</b>
<b>DAFTAR RIWAYAT HIDUP .....</b>	<b>74</b>





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 2. 1 Kategori <i>Throughput</i> .....	13
Tabel 2. 2 Kategori <i>Degradasi</i> .....	14
Tabel 2. 3 Kategori <i>Latency</i> .....	14
Tabel 3. 1 Alat yang digunakan.....	17
Tabel 3.2 Penggunaan pin Transmisi Lora pada pengirim dan penerima .....	38
Tabel 4.1 Parameter Hasil Pengukuran Pagi Hari Pertama.....	51
Tabel 4. 2 Parameter Hasil Pengukuran Siang Hari Pertama.....	53
Tabel 4. 3 Parameter Hasil Pengukuran Malam Hari Pertama.....	55
Tabel 4. 4 Parameter Hasil Pengukuran Pagi Hari kedua .....	57
Tabel 4. 5 Parameter Hasil Pengukuran Siang Hari kedua .....	59
Tabel 4. 6 Parameter Hasil Pengukuran Malam Hari kedua .....	61
Tabel 4. 7 Parameter Hasil Pengukuran Pagi Hari ketiga .....	62
Tabel 4. 8 Parameter Hasil Pengukuran Siang Hari ketiga .....	64
Tabel 4. 9 Parameter Hasil Pengukuran Malam Hari ketiga.....	66



POLITEKNIK  
NEGERI  
JAKARTA



## DAFTAR GAMBAR

Gambar 2. 1 Hirerarki IoT .....	3
Gambar 2. 2 Kodular.....	6
Gambar 2. 3 Halaman Design .....	7
Gambar 2. 4 Halaman Block.....	8
Gambar 2. 5 Code Block Control.....	8
Gambar 2. 6 Code Block Logic .....	9
Gambar 2. 7 Code Block Text.....	9
Gambar 2. 8 Code Block Variables .....	10
Gambar 2. 9 LogoFirebase.....	11
Gambar 3. 1 Ilustrasi Sistem Perangkat Satwa Liar.....	15
Gambar 3. 2 Flowchart Cara Kerja Aplikasi.....	16
Gambar 3. 3 Diagram Blok Sistem .....	17
Gambar 3. 4 Flowchart Pembuatan Firebase .....	19
Gambar 3. 5 Tampilan Awal Firebase .....	20
Gambar 3. 6 Halaman Real-time Database .....	21
Gambar 3. 7 Tampilan Auhtentication .....	21
Gambar 3. 8 Project Setting .....	22
Gambar 3. 9 Tampilan Register Aplikasi .....	23
Gambar 3. 10 Firebase Database Properties .....	24
Gambar 3. 11 Real-time Database.....	24
Gambar 3. 12 Tampilan Project Setting .....	25
Gambar 3. 13 Flowchart Perancangan Aplikasi.....	26
Gambar 3. 14 Splash Screen Aplikasi .....	27
Gambar 3. 15 Sketch Splash Screen .....	28
Gambar 3. 16 Pallette Clock 1 dan Clock 2 .....	29
Gambar 3. 17 Tampilan Masuk Akun .....	30
Gambar 3. 18 Skecth Masuk Akun .....	31
Gambar 3. 19 Tampilan Daftar Akun.....	32
Gambar 3. 20 Sketch Daftar Akun.....	33
Gambar 3. 21 (a) Tampilan Selamat Datang (b) Pemberitahuan Tentang Satwa Liar (c) Pemberitahuan Mengenai Aplikasi Peduli Satwa (d) Cara Penggunaan Aplikasi .....	34
Gambar 3. 22 Sketch Tentang Aplikasi.....	35
Gambar 3. 23 (a) Kondisi Perangkat Terbuka (b) Kondisi Perangkat Tertutup .....	36
Gambar 3. 24 Sketch Kendali dan Pemantauan .....	37
Gambar 3. 25 Menyambungkan ESP32 dengan Internet.....	39
Gambar 3. 26 Menyambungkan ESP32 dengan Firebase.....	39
Gambar 4. 1 Menghubungkan Android degan Jaringan Internet .....	42
Gambar 4. 2 Scan Barcode Menggunakan Smartphone.....	42
Gambar 4. 3 Tahapan Membuat Akun Baru.....	43
Gambar 4. 4 Tahapan Masuk Akun.....	43
Gambar 4. 5 Pengujian Pembuatan Akun (a) Password Tidak Sama (b) Username Sudah Terdaftar! Silahkan Login (c) Username Salah (d) Password Salah (e) Login Berhasil... ..	44
Gambar 4. 6 Tampilan Pemantauan Aplikasi (a) Kondisi Tidak Terdapat Satwa (b) Kondisi Terdapat Satwa .....	45
Gambar 4. 7 Gambar Real-time Database Pemantauan (a) Status “0” (b) Status “1” .....	45

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Gambar 4. 8 Kendali Perangkat Satwa (a) Kondisi Perangkat dibuka (b) Kondisi Perangkat ditutup .....	46
Gambar 4. 9 Kendali Perangkat Satwa (a) Status Perangkat “0” (b) Status Perangkat (1) .....	47
Gambar 4. 10 Set Up Pengujian Aplikasi Android .....	48
Gambar 4. 11 Website Firebase dan Aplikasi Wireshark .....	48
Gambar 4. 12 Statistic > Capture File Properties.....	49
Gambar 4. 13 Capture File Properties.....	49
Gambar 4. 14 HTTP Wireshark Pengukuran Jaringan Pagi Hari Pertama.....	50
Gambar 4. 15 TCP Wireshark Pengukuran Jaringan Pagi Hari Pertama .....	51
Gambar 4. 16 Capture File Properties Pengukuran Jaringan Pagi Hari Pertama .....	51
Gambar 4. 17 HTTP Wireshark Pengukuran Jaringan Siang Hari Pertama.....	52
Gambar 4. 18 TCP Wireshark Pengukuran Jaringan Siang Hari Pertama .....	52
Gambar 4. 19 Capture File Properties Pengukuran Jaringan Siang Hari Pertama .....	53
Gambar 4. 20 HTTP Wireshark Pengukuran Jaringan Malam Hari Pertama.....	54
Gambar 4. 21 TCP Wireshark Pengukuran Jaringan Malam Hari Pertama .....	54
Gambar 4. 22 Capture File Properties Pengukuran Jaringan Malam Hari Pertama.....	55
Gambar 4. 23 HTTP Wireshark Pengukuran Jaringan Pagi Hari kedua .....	56
Gambar 4. 24 TCP Wireshark Pengukuran Jaringan Pagi Hari kedua .....	56
Gambar 4. 25 Capture File Properties Pengukuran Jaringan Pagi Hari kedua.....	57
Gambar 4. 26 HTTP Wireshark Pengukuran Jaringan Siang Hari kedua .....	58
Gambar 4. 27 TCP Wireshark Pengukuran Jaringan Siang Hari kedua .....	58
Gambar 4. 28 Capture File Properties Pengukuran Jaringan Siang Hari kedua.....	58
Gambar 4. 29 HTTP Wireshark Pengukuran Jaringan Malam Hari kedua .....	59
Gambar 4. 30 TCP Wireshark Pengukuran Jaringan Malam Hari kedua.....	60
Gambar 4. 31 Capture File Properties Pengukuran Jaringan Malam Hari kedua .....	60
Gambar 4. 32 HTTP Wireshark Pengukuran Jaringan Pagi Hari ketiga .....	61
Gambar 4. 33 TCP Wireshark Pengukuran Jaringan Pagi Hari ketiga.....	62
Gambar 4. 34 Capture File Properties Pengukuran Jaringan Pagi Hari ketiga .....	62
Gambar 4. 35 HTTP Wireshark Pengukuran Jaringan Siang Hari ketiga .....	63
Gambar 4. 36 TCP Wireshark Pengukuran Jaringan Siang Hari ketiga.....	64
Gambar 4. 37 Capture File Properties Pengukuran Jaringan Siang Hari ketiga .....	64
Gambar 4. 38 Gambar 4. 39 HTTP Wireshark Pengukuran Jaringan Malam Hari ketiga	65
Gambar 4. 40 TCP Wireshark Pengukuran Jaringan Malam Hari ketiga.....	65
Gambar 4. 41 Capture File Properties Pengukuran Jaringan Malam Hari ketiga .....	66
Gambar 4. 42 Ilustrasi Memulai Pengujian Speedtest .....	69
Gambar 4. 43 Hasil Pengujian Speedtest.....	69
Gambar 4. 44 Hasil Pengukuran Speedtest .....	70



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**DAFTAR LAMPIRAN**

L- 1 Tampilan Aplikasi Android .....	75
L- 2 Coding Block Aplikasi .....	77





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Satwa liar yang dilindungi sering kali turun ke permukiman warga karena habitat asli mereka rusak. Hal ini menyebabkan mereka mencari makan di sekitar hutan atau permukiman penduduk. Untuk mengatasi masalah ini, diperlukan pemahaman yang lebih baik tentang kehidupan satwa liar, sehingga tindakan penanganan dan pencegahan dapat dilakukan dengan lebih optimal.

Diperlukan alat pengawasan dan pencegahan menggunakan teknologi *Internet of Things* (IoT), penggunaan IoT dapat melakukan pemantauan dan kendali jarak jauh yang memungkinkan pengawas lingkungan untuk memantau satwa liar melalui *smartphone*. Sistem ini menggabungkan teknologi Firebase dan aplikasi Android dengan komunikasi *Long Range* (LoRa) untuk memungkinkan transmisi data di wilayah dengan keterbatasan jaringan internet. Penggunaan aplikasi Android memiliki kelebihan untuk pemantauan satwa liar dari jarak jauh yang dapat memudahkan setiap penggunanya melakukan pengawasan dimanapun dan kapanpun. Sementara itu, *Long Range* dimanfaatkan karena memiliki kelebihan komunikasi nirkabel yang handal. Sistem pemantauan dan kendali satwa liar diterapkan di sekitar hutan atau permukiman warga. Ini memungkinkan pengawas lingkungan untuk memantau dan mencegah satwa liar memasuki kawasan penduduk.

Berdasarkan penjelasan di atas, dirancanglah sebuah aplikasi Android untuk sistem perangkat satwa liar yang dapat menjadi solusi efektif untuk pengawasan satwa di lingkungan permukiman penduduk. Aplikasi ini menampilkan fitur kondisi satwa liar yang terperangkap dan sistem kendali perangkat. Untuk meningkatkan efektivitas pengawasan satwa liar dan meminimalkan terjadinya hal-hal yang tidak diinginkan, baik pada satwa maupun manusia, dirancanglah alat dalam tugas akhir ini yang berjudul "Rancang Bangun Prototype Sistem Perangkat Satwa Liar di Permukiman Penduduk Berbasis Android."



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, adapun beberapa masalah yang perlu dirumuskan, ialah:

1. Bagaimana merancang aplikasi Android untuk perangkat satwa liar di pemukiman penduduk?
2. Bagaimana merealisasikan aplikasi Android perangkat satwa liar di pemukiman penduduk dengan alat mikrokontroler?
3. Bagaimana proses komunikasi data dari aplikasi Android ke Firebase?
4. Bagaimana melakukan pengujian untuk mengukur kinerja proses komunikasi data dari Firebase ke aplikasi Android?

### 1.3 Tujuan

Tujuan dari penulisan laporan tugas akhir ini adalah:

1. Dapat merancang aplikasi sistem perangkat satwa liar di pemukiman penduduk.
2. Dapat membuat aplikasi sistem perangkat satwa liar di pemukiman penduduk dengan alat mikrokontroler.
3. Dapat menjalankan proses komunikasi data dari aplikasi Android ke Firebase
4. Menguji performasi aplikasi sistem perangkat satwa liar di pemukiman penduduk.

### 1.4 Luaran

Adapun luaran dari tugas akhir ini adalah:

1. Perancangan aplikasi Android perangkat satwa liar di pemukiman penduduk.
2. Laporan Tugas Akhir.
3. Artikel ilmiah.
4. Poster.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

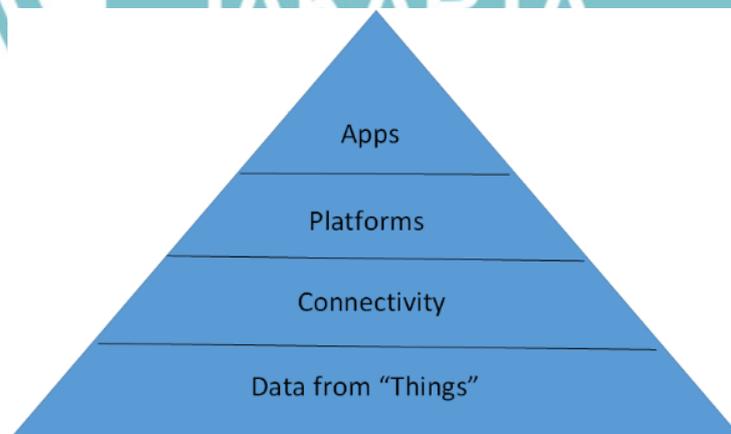
## BAB II TINJAU PUSTAKA

### 2.1 *Internet of Things*

*Internet of Things* (IoT) adalah sebuah teknologi canggih memiliki konsep yang bertujuan untuk memperluas dan mengembangkan manfaat dari konektivitas internet yang tersambung terus menerus. menghubungkan benda benda di sekitar agar aktivitas sehari hari menjadi lebih mudah dan efisien yang sangat membantu segala pekerjaan manusia. (Selay, 2022)

IoT menghubungkan perangkat elektronik berupa sensor atau aktuator ke internet untuk mengumpulkan dan berbagi data secara otomatis yang dimanfaatkan untuk analisa lebih lanjut. Salah satu aspek penting dari IoT adalah *Cyber Physical System* (CPS), yang menggabungkan komputasi, jaringan, dan proses fisik untuk menciptakan sistem yang saling terhubung dan berinteraksi, sehingga memungkinkan pengguna untuk mengontrol dan memantau perangkat dari jarak jauh. (Triawan, 2023)

IoT terdiri dari beberapa komponen utama yang memungkinkan interkoneksi dan pertukaran data antara perangkat. Komponen-komponen ini meliputi data dari “*Things*”, konektivitas atau perangkat yang terhubung, platform sebagai lapisan perangkat lunak untuk mengelola dan menganalisis data, serta aplikasi yang memberikan layanan dan informasi berdasarkan data yang dikumpulkan dari perangkat dan platform IoT. Gambar 2.1 di bawah ini merupakan piramida dari hierarki IoT.



Gambar 2. 1 Hierarchy IoT

Sumber: Tito Yuwono, 2024



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Berikut beberapa komponen dari IoT:

#### 1. *Things*

Hierarki yang paling bawah (pertama) adalah data dari *Things*. Data bisa berupa koordinat lokasi, data pengukuran sensor, citra maupun video. Jenis data tergantung jenis dan tujuan IoT itu di bangun.

#### 2. Konektivitas

Perangkat terhubung adalah perangkat fisik yang terhubung ke internet dan dilengkapi dengan kemampuan komputasi dan komunikasi. Perangkat ini berfungsi sebagai otak dari sistem IoT, di mana data dari sensor diterima, diproses, dan digunakan untuk mengambil keputusan atau melakukan tindakan tertentu secara lokal sebelum data dikirim ke platform atau *cloud* untuk analisis lebih lanjut.

#### 3. Platform

Lapisan perangkat lunak yang mengintegrasikan dan mengelola data dari berbagai peran terhubung. Platform ini berfungsi untuk mengorganisasi, menyimpan, dan menganalisis data, serta menyediakan antar muka untuk penggunaan dan pengelolaan aplikasi IoT. Platform mendukung pengembangan aplikasi khusus yang dapat memberikan layanan dan solusi yang sesuai dengan kebutuhan pengguna.

#### 4. Aplikasi

Bagian yang terlihat oleh pengguna akhir dan memberikan layanan atau informasi berdasarkan data yang dikumpulkan dari perangkat atau informasi berdasarkan data yang dikumpulkan dari perangkat dan platform IoT. Aplikasi IoT bertindak sebagai antarmuka yang memungkinkan interaksi dan kontrol terhadap sistem IoT secara intuitif dan efisien.

Melalui hierarki IoT ini, data dari berbagai perangkat dan sensor dihimpun, diproses, dan diintegrasikan untuk memberikan Solusi yang berharga dan layanan berbasis IoT kepada pengguna. (Erwin, 2023)

## 2.2 Internet

Internet (*Inter-Network*) adalah sebutan untuk sekumpulan jaringan komputer yang menghubungkan situs akademik, pemerintahan, komersial, organisasi, maupun perorangan. Internet menyediakan akses untuk layanan telekomunikasi dan sumber daya informasi untuk jutaan pemakainya yang tersebar di seluruh dunia.



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Jaringan yang membentuk internet bekerja berdasarkan suatu set protokol standar yang digunakan untuk menghubungkan jaringan komputer dan mengalami lalu lintas dalam jaringan. Protokol ini mengatur format data yang diijinkan, penanganan kesalahan (*error handling*), lalu lintas pesan, dan standar komunikasi lainnya. Protokol standar pada internet dikenal sebagai TCP/IP (*Transmission Control Protocol/Internet Protocol*). Protokol ini memiliki kemampuan untuk bekerja diatas segala jenis komputer, tanpa terpengaruh oleh perbedaan perangkat keras maupun sistem operasi yang digunakan. Sebuah sistem komputer yang terhubung secara langsung ke jaringan memiliki nama domain dan alamat IP (*Internet Protocol*) dalam bentuk numerik dengan format tertentu sebagai pengenal. Internet juga memiliki *gateway* ke jaringan dan layanan yang berbasis protokol lainnya. (Rohaya, 2008)

### 2.3 Android

Android adalah sebuah sistem operasi perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google inc memberi Android inc merupakan pendatang baru yang membuat perangkat lunak untuk ponsel atau *smartphone*.

Android merupakan sebuah sistem operasi seluler yang didasarkan pada versi modifikasi dari kernel linux dan perangkat sumber terbuka lainnya. Android merupakan sebuah sistem operasi seluler yang didasarkan pada versi modifikasi dari kernel linux dan perangkat sumber terbuka lainnya. (Ratrie Lara Ditha, 2023)

### 2.4 Kodular

Kodular adalah situs web yang menawarkan alat untuk menggunakan blok program untuk membuat aplikasi Android. Dengan kata lain, peneliti tidak perlu membangun aplikasi android dengan mengetik kode program secara manual. Kelebihan Kodular *Store* dan Kodular *Extension* IDE memudahkan pembuat aplikasi android untuk diunggah atau diupload ke dalam Kodular *Store*. Ini dapat dilakukan saat membuat blok program *Extension* IDE sesuai keinginan pembuat. (Lutvi Aditya Rahmadan, 2023)

Pemrograman blok adalah fitur inti dari kodular, dengan fitur ini kita tidak perlu lagi memasukkan kode program secara manual untuk membuat aplikasi

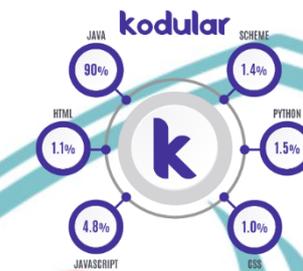


## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Android. Kodular juga menyediakan *dBase mini* dan fungsi penyimpanan sehingga kita dapat menyimpan dan mengunduh data sesuai keinginan. Dari segi antarmuka/GUI, kode dapat disesuaikan dengan tema untuk membuat aplikasi yang kita buat lebih modern dan professional. (Muyasir, 2022)



Gambar 2. 2 Kodular

Sumber: ZekkelAR, 2022

Kodular memiliki berbagai halaman dengan fungsinya masing-masing yang saling terhubung. Pada halaman utama Kodular, terdapat halaman desain yang menyediakan komponen, formulir, dan properti yang dibutuhkan oleh pengembang dalam membangun sistem. Selanjutnya, terdapat halaman blok yang digunakan untuk membuat pemrograman dari proyek yang sedang dikembangkan. Berikut ini adalah penjelasan mengenai halaman-halaman yang terdapat di Kodular:

#### a. Halaman *designer*

Halaman *designer* menyediakan ruang untuk merancang sistem yang akan dibangun dengan menyajikan berbagai komponen di bagian *Palette*. Di bagian *Palette* ini, terdapat berbagai macam komponen yang dapat digunakan oleh pengembang untuk sistemnya, antara lain *Button*, *Label*, *Image*, *Notifier*, dan sebagainya. Selain itu, terdapat tampilan yang memuat semua komponen yang digunakan, sehingga pengembang dapat dengan mudah memodifikasi sistem yang dibuat.

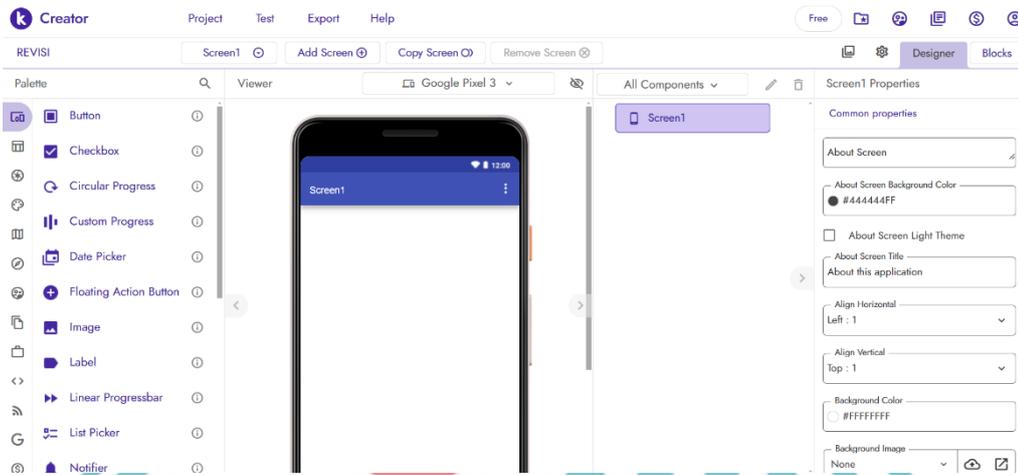
Menu *design* Kodular digunakan untuk membuat dan mengatur tampilan aplikasi dengan cara *drag-and-drop* berbagai komponen yang terdapat pada bagian *pallette* seperti yang dapat dilihat pada Gambar 2.3 *Viewer* pada bagian tengah layar yang merupakan tampilan prangkat Android yang digunakan sebagai kanvas tempat pengguna bisa menambahkan dan mengatur komponen dari *pallette*. Pada bagian



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

atas pada *viewer*, terdapat *dropdown* yang digunakan untuk memilih jenis perangkat dimana digunakan untuk melihat pratinjau aplikasi.



Gambar 2. 3 Halaman *Design*

Sumber: Kodular

**b. Halaman *block***

Dalam pemrograman komputer, sebuah blok atau blok kode adalah bagian dari kode program yang dikelompokkan bersama-sama, terdiri dari satu atau lebih pernyataan dan deklarasi. Blok ini berfungsi sebagai unit dasar dalam pemrograman terstruktur, di mana struktur kontrol dalam program terbentuk dari blok-blok tersebut.

Blok kode memungkinkan sekelompok pernyataan atau instruksi diperlakukan seolah-olah mereka adalah satu kesatuan. Hal ini memudahkan programmer untuk mengelola logika program secara terorganisir dan terstruktur. Selain itu, blok juga memiliki peran penting dalam mempersempit ruang lingkup leksikal dari variabel, prosedur, dan fungsi yang dideklarasikan di dalamnya. Dengan demikian, variabel atau fungsi yang dideklarasikan dalam sebuah blok tidak akan bertentangan dengan variabel atau fungsi yang memiliki nama yang sama yang digunakan di bagian lain dari program untuk tujuan yang berbeda.

Bahasa pemrograman yang mendukung penciptaan blok, termasuk blok yang dapat bersarang di dalam blok lain, disebut sebagai bahasa pemrograman berstruktur blok (*block-structured programming language*). Dalam bahasa pemrograman berstruktur blok, nama-nama variabel dan elemen lain seperti prosedur yang dideklarasikan di dalam blok luar tetap dapat diakses oleh blok-blok

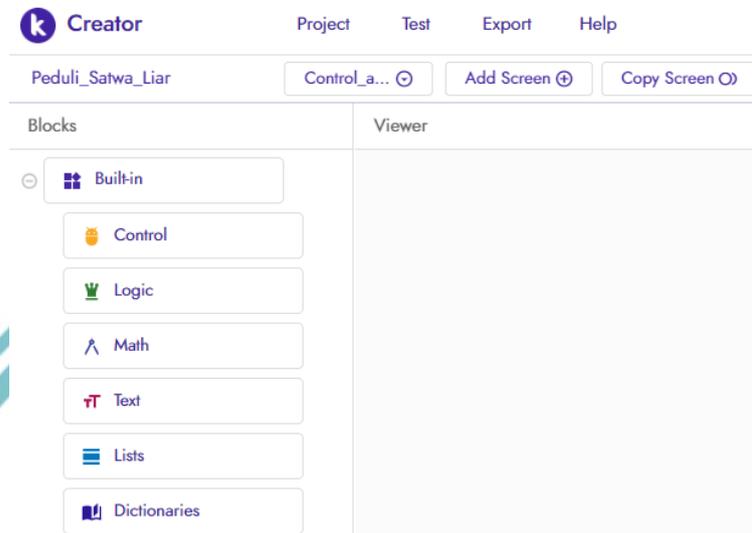


## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

yang berada di dalamnya. Namun, blok yang lebih dalam memiliki akses terbatas hanya pada deklarasi yang berada dalam blok yang sama atau di blok luar yang menaunginya, sehingga menjaga struktur dan kejelasan kode program secara keseluruhan.



Gambar 2. 4 Halaman *Block*

Sumber: Kodular

Blok adalah elemen pemrograman yang terdiri dari berbagai *codeblock* dengan fungsinya masing-masing. *Codeblock* ini membantu pengembang membuat aplikasi yang fungsional, interaktif, dan dinamis tanpa perlu menulis kode secara manual. Pada Blok terdapat beberapa komponen seperti *Control*, *Logic*, *Math*, *Text*, *Lists*, *Dictionaries*, *Colors*, *Variables*, dan *Procedures*. Berikut ini adalah fungsi dari masing-masing *codeblock*:

#### 1. *Control*

Kontrol memiliki fungsi untuk mengatur alur program dengan blok kontrol seperti *while*, *for*, dan pengaturan waktu (*timers*). Pada Gambar 2.5 merupakan kode blok yang sering digunakan.



Gambar 2. 5 *Code Block Control*

Sumber: Kodular



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada Gambar 2.4 merupakan contoh kode blok yang sering digunakan pada Kodular. Blok “*if, then, else*” dan “*Open another screen*” memiliki fungsi dalam pengembangan alur logika dan navigasi antar layar aplikasi.

### 2. Logic

*Logic* memungkinkan pengembang menambahkan logika kondisi, seperti pernyataan *if* atau *else* dan *loop*, ke dalam aplikasi.



Gambar 2. 6 Code Block Logic

Sumber: Kodular

Blok-blok pada *logic* memiliki fungsi alur logika. Pada blok “*and*” dimana blok tersebut digunakan untuk mengevaluasi dua kondisi. Blok “*or*” juga digunakan untuk mengevaluasi dua kondisi tetapi menghasilkan nilai ‘*true*’ jika salah satu atau keduanya bernilai ‘*true*’. Blok “*not*” digunakan untuk membalik nilai Boolean. Pada blok “*true*” dan “*false*” masing-masing menghasilkan nilai Boolean ‘*true*’ dan ‘*false*’ jika berbeda.

### 3. Math

Memfasilitasi operasi matematika, termasuk operasi dasar (penjumlahan, pengurangan, perkalian, pembagian) dan fungsi yang lebih kompleks seperti trigonometri.

### 4. Text

Mengelola dan memanipulasi teks dengan blok untuk penggabungan, pemisahan, penggantian, dan pemformatan teks. Gambar 2.7 menunjukkan contoh *code block text*. *Code block text* merupakan pernyataan *text* yang digunakan untuk banyak hal, seperti memasukan link, teks yang ingin dimasukkan untuk sebuah kondisi, dan yang lain sebagainya.



Gambar 2. 7 Code Block Text

Sumber: Kodular



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 5. Lists

Pengelolaan daftar (array) dalam pemrograman sangat penting untuk menyimpan dan memanipulasi sekumpulan data yang terkait. Dengan menggunakan blok kode, kita dapat membuat, menambah, menghapus, dan mengakses elemen-elemen dalam sebuah array secara terstruktur.

### 6. Colors

Mengatur dan memanipulasi warna adalah aspek penting dalam desain antarmuka pengguna (UI) dan pengembangan aplikasi. Warna dapat digunakan untuk memperkuat estetika, meningkatkan pengalaman pengguna, dan membantu dalam navigasi.

### 7. Variables

Membuat dan mengelola variabel untuk menyimpan data yang dapat digunakan dan diubah selama eksekusi program.



Gambar 2. 8 Code Block Variables

Sumber: Kodular

Blok-blok pada Gambar 2.8 memiliki kegunaan untuk menginisialisasi dan memanipulasi variabel *local* dan *global*. Blok “*initialize local name to*” digunakan untuk mendeklarasi dan menginisialisasi variabel lokal dalam lingkup tertentu, sehingga variabel ini hanya dapat diakses di dalam lingkup tersebut. Untuk blok “*initialize global name to*” digunakan untuk mendeklarasi dan menginisialisasi variabel global yang dapat diakses dari mana saja dalam aplikasi. Blok “*get*” merupakan blok yang digunakan untuk mengambil data dari variabel yang sudah ada dan blok “*set*” digunakan untuk menetapkan atau mengubah nilai dari variabel yang sudah ada.

### 8. Procedures

Memungkinkan pengembang untuk membuat prosedur atau fungsi, yang membantu dalam modularisasi kode dan meningkatkan keterbacaan serta pemeliharaan kode.



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## 2.5 Firebase

Firebase adalah penyedia layanan *cloud* dengan *back-end* sebagai servis yang berbasis di San Fransisco, California. Firebase membuat sejumlah produk untuk pengembangan aplikasi *Mobile* ataupun *web*. Firebase didirikan oleh Andrew Lee dan James Tamplin pada tahun 2011 dan diluncurkan dengan *cloud database* secara *real-time* di tahun 2012.

Produk utama dari Firebase yakni suatu *database* yang menyediakan API untuk memungkinkan pengembang menyimpan dan mensinkronisasi data lewat *multipleclient*. Perusahaan ini diakusisi oleh Google pada Oktober 2014.

Firebase adalah penyedia layanan *real-time database* dan *backend* sebagai layanan. Suatu aplikasi yang memungkinkan pengembang membuat API untuk disinkronisasikan untuk *client* yang berbeda-beda dan disimpan pada *cloud*-nya Firebase. *Database* Firebase juga bersifat bisa diakses lewat REST API. REST API tersebut menggunakan protokol *Server-Sent Event* dengan membuat koneksi HTTP untuk menerima *push notification* dari server. Pengembang menggunakan REST API untuk *post* data yang selanjutnya Firebase *client library* yang sudah diterapkan pada aplikasi yang dibangun yang akan mengambil data secara *real-time*. (Anisyia Sonita, 2018)



Gambar 2. 9 Logo Firebase

Sumber: Angga Abi Pratama, 2022

Firebase adalah platform yang dikembangkan oleh Google untuk membantu pengembang dalam membuat aplikasi web dan aplikasi seluler yang berkualitas tinggi, meningkatkan basis pengguna, dan memperoleh lebih banyak pendapatan. Firebase menawarkan berbagai layanan dan library yang mendukung integrasi dengan berbagai platform, termasuk Android, iOS, JavaScript, Java, Objective-C, dan Node.js. Salah satu fitur utama dari Firebase adalah *Real-time Database*.



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## 2.6 Wireshark

Wireshark adalah salah satu analisis paket bebas serta sumber terbuka. Perangkat ini untuk digunakan sebagai pemecah suatu permasalahan jaringan, analisis, perangkat lunak dan serta mengembangkan protokol komunikasi, dan juga pendidikan, dari sekian banyak aplikasi *Network Analyzer* yang banyak digunakan oleh *Network Administrator* untuk menganalisa kinerja jaringannya dan mengontrol lalu lintas data di jaringan yang di kelola Wireshark. Wireshark mampu menangkap paket-paket data yang ada pada jaringan tersebut. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa. (Romasella Tri Novita, 2021)

## 2.7 Hypertext Transfer Protocol (HTTP)

HTTP adalah sebuah protokol meminta atau menjawab antara *client* dan *server*. Sebuah *client* HTTP seperti *web browser*, biasanya memulai permintaan dengan membuat hubungan TCP/IP ke *port* tertentu di tuan rumah yang jauh (biasanya *port* 80). Sebuah *server* HTTP yang mendengarkan di *port* tersebut menunggu *client* mengirim kode permintaan (*request*), seperti "GET / HTTP/1.1" (yang akan meminta halaman yang sudah ditentukan), diikuti dengan pesan MIME yang memiliki beberapa informasi kode kepala yang menjelaskan aspek dari permintaan tersebut, diikuti dengan badan dari data tertentu. Beberapa kepala (*header*) juga bebas ditulis atau tidak, sementara lainnya (seperti tuan rumah) diperlukan oleh protokol HTTP/1.1. Begitu menerima kode permintaan (dan pesan, bila ada), server mengirim kembali kode jawaban, seperti "200 OK", dan sebuah pesan yang diminta, atau sebuah pesan *error* atau pesan lainnya. Pengembangan HTTP dikoordinasi oleh Konsorsium *World Wide Web* (W3C) dan grup bekerja *Internet Engineering Task Force* (IETF), bekerja dalam publikasi satu seri RFC, yang paling terkenal RFC 2616, yang menjelaskan HTTP/1.1, versi HTTP yang digunakan umum sekarang ini. (Adzan Abdul Zabar, 2015)

## 2.8 Quality of Service

*Quality of Service* (QoS) merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari satu servis. QoS digunakan untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu servis. Parameter yang



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

digunakan pada QoS terdiri dari *throughput*, *packet loss*, dan *delay*. (Wulandari, 2016)

Parameter *Quality of Service* terdiri dari:

1. *Throughput*

*Throughput* yaitu, kecepatan (*rate*) transfer data efektif yang diukur dalam bps (*bit per second*). *Throughput* adalah jumlah total data atau paket yang berhasil diterima di tujuan dalam interval waktu tertentu, dibagi dengan durasi interval waktu tersebut. Ini berarti *throughput* merupakan ukuran kinerja jaringan atau sistem dalam mentransfer data. Perhitungan *throughput* ditentukan pada persamaan 2.1.

$$\text{Throughput} = \frac{\text{Paket data diterima}}{\text{Lama pengamatan}} \quad (2.1)$$

Untuk menentukan besar *throughput* yang diukur dapat menyesuaikan data hasil pengukuran *throughput* dengan kategori pada Tabel 2.1.

Tabel 2. 1 Kategori *Throughput*

Kategori <i>Throughput</i>	<i>Throughput</i> (bps)	Indeks
Sangat bagus	100	4
Bagus	75	3
Sedang	50	2
Jelek	<25	1

Sumber: THIPON

2. *Packet Loss*

*Packet loss* terjadi ketika paket data yang dikirim dari pengirim (*source*) tidak sampai ke penerima (*destination*) dengan sukses. Paket-paket ini hilang di sepanjang jalur dari pengirim ke penerima karena berbagai alasan seperti paket yang hilang dapat terjadi karena *collision* dan *congestion* pada jaringan. Perhitung *packet loss* ditentukan pada persamaan 2.2.

$$\text{Packet Loss} = \frac{(\text{Paket data dikirim} - \text{Paket data diterima}) \times 100\%}{\text{Paket data yang dikirim}} \quad (2.2)$$

Untuk menentukan suatu jaringan sudah baik atau tidaknya dapat menyesuaikan data *packet loss* yang diukur dengan kategori pada Tabel 2.2.

Tabel 2. 2 Kategori *Degradasi*

Kategori Degradasi	<i>Packet Loss</i> (%)	Indeks
Sangat bagus	0	4
Bagus	3	3
Sedang	15	2
Jelek	25	1

Sumber: THIPON

3. *Delay (Latency)*

*Delay (Latency)* merupakan waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, *congesti*, atau juga waktu proses yang lama. Perhitungan *delay* ditentukan pada persamaan

2.3.

$$Delay = \frac{Packet\ length}{Link\ Bandwidth} \quad (2.3)$$

Untuk menentukan besar *delay* suatu jaringan dapat menyesuaikan data *delay* yang diukur dengan kategori pada Tabel 2.3.

Tabel 2. 3 Kategori *Latency*

Kategori <i>Latency</i>	Besar <i>Delay</i> (ms)	Indeks
Sangat bagus	<150 ms	4
Bagus	150 ms – 300 ms	3
Sedang	300 ms - 450 ms	2
Jelek	>450ms	1

Sumber: THIPON



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

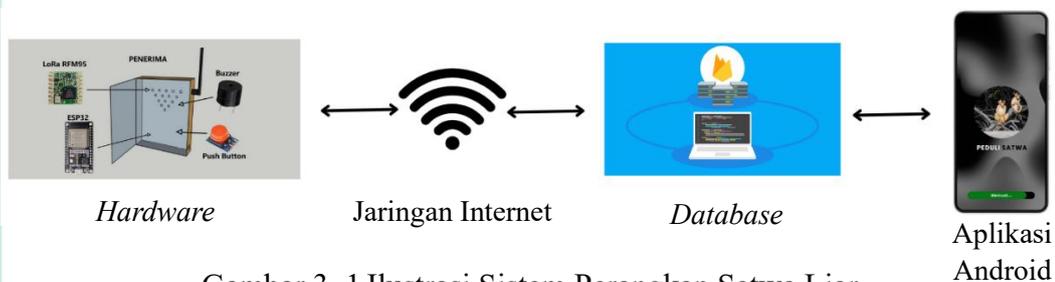
## BAB III PERENCANAAN DAN REALISASI

### 3.1 Rancang Alat

Pada tugas akhir ini, dirancang dan dibangun sebuah "Rancang Bangun Prototipe Sistem Perangkat Satwa Liar di Pemukiman Penduduk Berbasis Android" yang bertujuan untuk memudahkan pengguna, terutama pengawas satwa, dalam memantau satwa liar yang memasuki pemukiman. Sistem ini terhubung secara *real-time* dengan Firebase dan diintegrasikan dengan aplikasi Android Perangkat Satwa. Bagian yang dibahas meliputi perencanaan program untuk sistem transmisi dan pemrograman aplikasi Android, dengan sensor yang digunakan, yaitu sensor ultrasonik, kemudian mengirimkan status kondisi satwa dengan ESP32 dan LoRa sebagai sistem transmisinya. Untuk kendali perangkat menggunakan servo yang mengirimkan perintah dari aplikasi.

#### 3.1.1 Deskripsi Alat

Alat ini berfungsi mendeteksi keberadaan satwa di dalam perangkat yang dipasang di sekitar pemukiman penduduk. Deteksi dilakukan menggunakan sensor ultrasonik, dan ketika satwa terdeteksi, satwa tersebut akan langsung terperangkap. Status kondisi perangkat satwa dikirimkan oleh LoRa pengirim dan diterima oleh LoRa penerima. Data yang diperoleh kemudian dikirimkan ke *database* untuk disimpan dan ditampilkan pada aplikasi Android. Aplikasi ini dirancang menggunakan platform Kodular dan memanfaatkan Firebase sebagai *database*, sehingga menciptakan sistem yang efisien dan terhubung dengan internet. Pengendalian jarak jauh dilakukan melalui aplikasi, yang mengirimkan perintah yang tersambung ke Firebase, memungkinkan alat untuk membuka dan menutup perangkat secara jarak jauh. Gambar 3.1 menunjukkan ilustrasi dari sistem perangkat satwa liar.



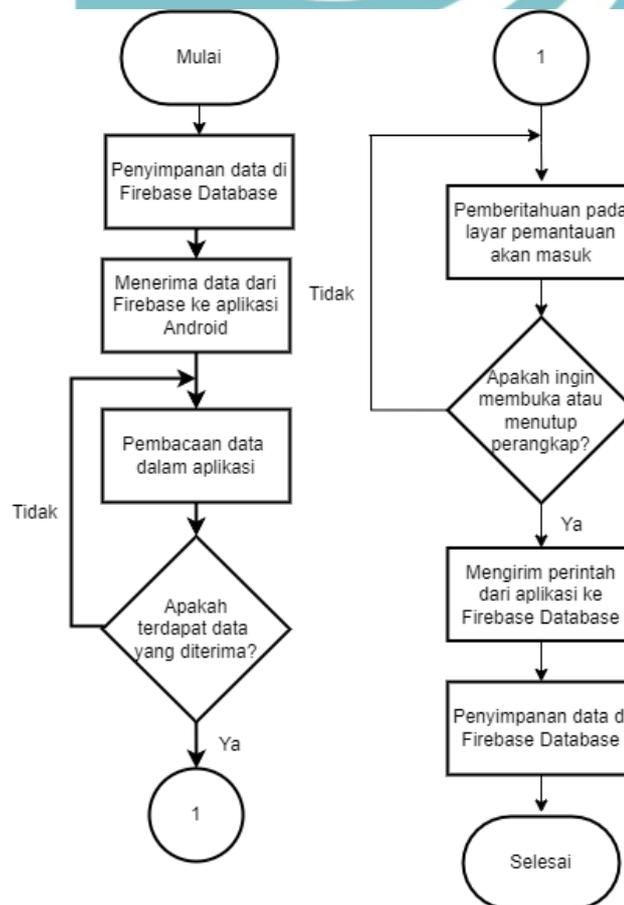
Gambar 3. 1 Ilustrasi Sistem Perangkat Satwa Liar



## © Hak Cipta milik Politeknik Negeri Jakarta

### 3.1.2 Cara Kerja Alat

Sistem yang dibuat bertujuan untuk mendeteksi keberadaan satwa liar yang akan memasuki pemukiman penduduk serta berfungsi sebagai pengendali perangkat satwa. Aplikasi Android bernama Perangkat Satwa menyediakan sistem pemantauan dan kendali perangkat secara jarak jauh dalam satu menu atau tampilan aplikasi. Pada sistem pemantauan, indikator yang digunakan adalah sensor ultrasonik yang mendeteksi keberadaan satwa. Data hasil deteksi sensor tersebut kemudian dikirimkan dalam bentuk status menggunakan LoRa, yang selanjutnya diterima oleh LoRa penerima dan dibaca oleh ESP32. Data dari ESP32 dikirimkan ke Firebase *database*. Untuk kendali, data dikirimkan melalui aplikasi yang dirancang khusus untuk mengirim perintah, yang kemudian diterima oleh Firebase *database* secara *real-time* dan langsung terhubung dengan perangkat atau alat. Seluruh data dari sistem pemantauan dan kendali tersimpan secara *real-time* di Firebase *database* dan ditampilkan pada aplikasi Android. Alur kerja aplikasi Perangkat Satwa dapat dilihat pada Gambar 3.2.



Gambar 3.2 Flowchart Cara Kerja Aplikasi

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 3.1.3 Spesifikasi Alat

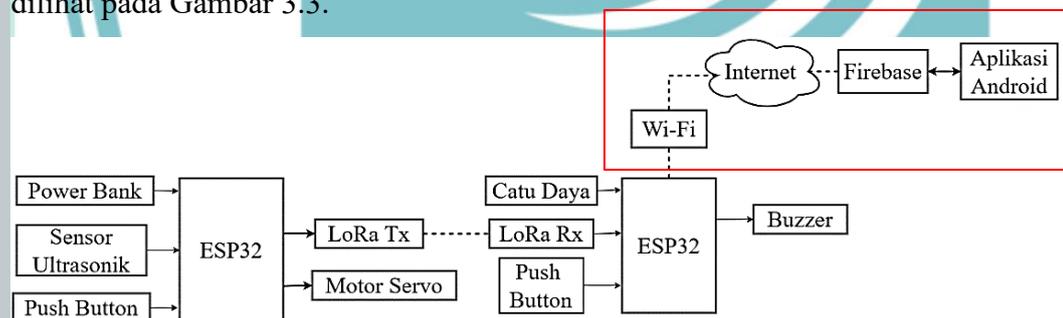
Perangkat keras maupun lunak yang digunakan pada tugas akhir memiliki spesifikasi tertentu. Pada Tabel 3.1 menunjukkan spesifikasi perangkat untuk merancang aplikasi Peduli Satwa.

Tabel 3. 1 Alat yang digunakan

Nama Alat	Spesifikasi
Smartphone	ITEL P55 8/128 GB
Laptop	OS Windows
Kodular	1.5C.3 Fenix
Firebase	Wireshark 4.3.6
Wireshark	Firebase Android BoM 33.1.1

### 3.1.4 Diagram Blok

Diagram blok terdapat *Transmitter* dan *Receiver*. Diagram blok sistem dapat dilihat pada Gambar 3.3.



Gambar 3. 3 Diagram Blok Sistem

Pada gambar menunjukkan diagram blok yang terdiri dari tiga bagian, yaitu bagian *input*, *proses*, dan *output*. Bagian *input* terdiri sensor ultrasonik dan baterai 5V ke ESP32 pada masing-masing alat. Pada bagian *proses* terdiri dari bagian *proses* pada *transmitter* dan juga *proses* pada *receiver*. Pada bagian *proses* di *transmitter* terdapat ESP32 yang mendapatkan *power* dari *catu daya*, kemudian di bagian *proses* pada *receiver* terdapat LoRa penerima, ESP32, dan juga buzzer. Pada bagian *output* terdapat *database* Firebase sebagai keluaran data yang akan disimpan dan disambungkan pada aplikasi Android yang berfungsi untuk menampilkan hasil kondisi keberadaan satwa secara *real-time* dari sensor ultrasonik.



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 3.2 Realisasi Alat

Pada realisasi alat akan dibahas secara rinci mengenai realisasi tugas akhir yang meliputi beberapa langkah utama. Pertama, pembuatan Firebase yang mencakup konfigurasi dan pengaturan *database* untuk mendukung aplikasi. Selanjutnya, proses menyambungkan Firebase ke aplikasi Android dengan menggunakan platform pengembangan Kodular, meliputi pengaturan koneksi dan integrasi layanan Firebase ke dalam aplikasi. Terakhir, realisasi program aplikasi Android akan dijelaskan, mencakup implementasi kode, pengujian, dan optimasi aplikasi agar berfungsi dengan baik sesuai tujuan yang diharapkan. Keseluruhan langkah ini merupakan bagian dari realisasi alat dalam tugas akhir yang berjudul "Rancang Bangun Aplikasi Android Sistem Perangkap Satwa Liar di Pemukiman Penduduk".

#### 3.2.1. Realisasi Pembuatan *Database*

Dalam pembuatan sistem pengendalian dan pemantauan satwa secara *real-time*, Firebase *real-time database* digunakan sebagai media transmisi data. Setelah sensor ultrasonik mendeteksi keberadaan satwa, data status hasil deteksi dikirimkan oleh LoRa pengirim ke LoRa penerima. Selanjutnya, ESP32 akan mengirimkan data tersebut ke internet, dan data ini kemudian diterima serta diperbarui secara otomatis melalui Firebase *real-time database*. Firebase *real-time database* memungkinkan data untuk terus diperbarui sesuai dengan kondisi terkini dari hasil deteksi keberadaan satwa, sehingga pemantauan dan kendali dapat dilakukan secara *real-time*.

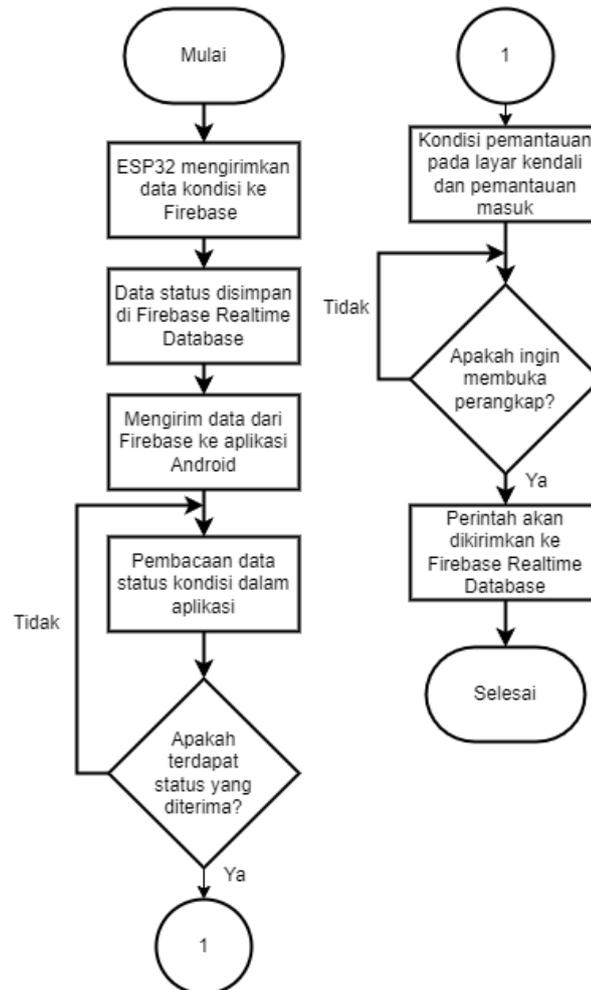
Selain itu, kendali sistem juga dapat dilakukan melalui aplikasi yang telah dirancang untuk mengirimkan perintah ke Firebase *real-time database*. Perintah yang dikirimkan melalui aplikasi ini akan diterima oleh Firebase dan secara otomatis diperbarui dalam *database*. Kemudian, perintah tersebut langsung diteruskan ke perangkat atau alat yang terhubung, sehingga memungkinkan pengendalian perangkap satwa secara jarak jauh. Gambar 3.4 menunjukkan *flowchart* perancangan *database* Firebase.



## Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 3. 4 *Flowchart* Pembuatan Firebase

Pada Gambar 3.4 ditunjukkan *flowchart* perancangan *database* pada Firebase. Langkah pertama dalam pembuatan *database* Firebase adalah membuat proyek baru. Setelah itu, buatlah *real-time database* dan variabel untuk data yang akan disimpan. Variabel yang digunakan harus sesuai dengan yang dibuat pada bagian pengirim. Setelah *database* dan variabel berhasil dibuat, *database* dapat menerima data dari pengirim. Sebaliknya, jika *database* dan variabel tidak berhasil dibuat, maka *database* tidak akan dapat menerima data. Proses pembuatan *database* Firebase dilakukan dengan langkah-langkah berikut:

1. Membuat proyek baru

Membuat proyek baru *database* menggunakan Firebase dapat dilakukan dengan cara sebagai berikut:

- a. Langkah awal membuat *database* pada Firebase adalah membuka laman web Firebase.

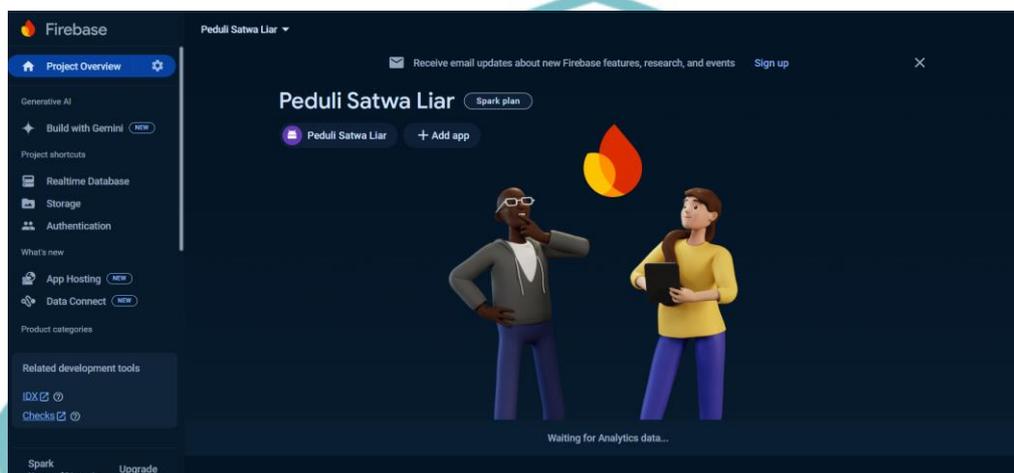


## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- b. Saat awal membuka laman Firebase klik “*Get Started*”.
- c. Ketika ingin membuat projek baru, maka klik “*Create a project*”.
- d. Kemudian menamakan projek dan ikuti langkah-langkah yang diberikan. Tahap selanjutnya adalah menyesuaikan kebutuhan projek yang akan dibuat. Dapat dilihat pada Gambar 3.5 halaman pada projek yang akan dibuat.



Gambar 3. 5 Tampilan Awal Firebase

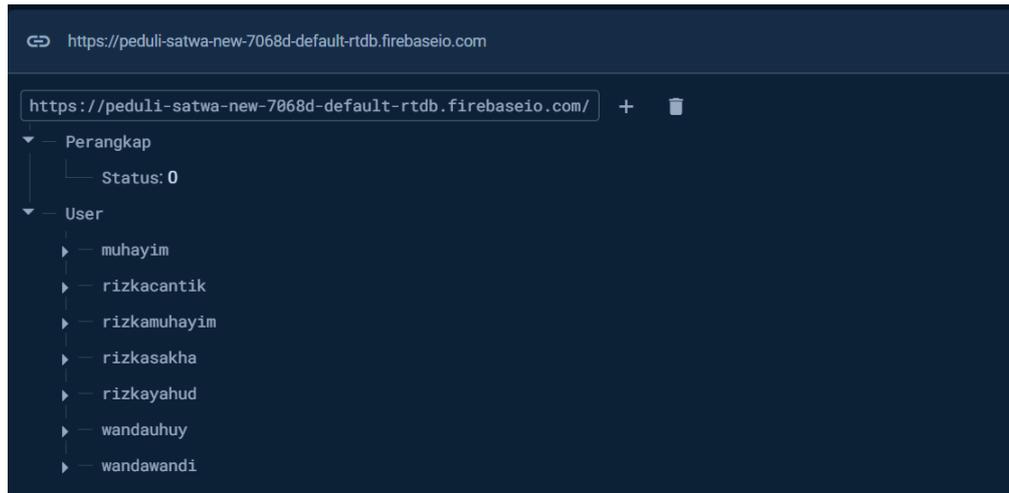
### 2. Membuat *Real-time Database*

*Database real-time* adalah jenis *database* yang dirancang untuk menyinkronkan dan menyimpan data antara pengguna secara langsung dan terus menerus, sehingga setiap perubahan yang terjadi pada data akan segera diperbarui dan diterapkan ke semua klien yang terhubung. Hal ini memastikan bahwa semua pengguna memiliki akses ke data yang selalu mutakhir dan konsisten, tanpa harus melakukan refresh manual. *Database real-time* sering digunakan dalam aplikasi yang memerlukan interaksi langsung dan pembaruan data yang cepat, seperti aplikasi pesan instan, kolaborasi dokumen, dan pelacakan lokasi. Selain itu, database ini juga mampu menyimpan data sementara yang nantinya akan ditampilkan pada aplikasi, misalnya pada aplikasi Android. Cara kerja *database real-time* mencakup manajemen dan eksekusi data serta instruksi secara langsung dalam aplikasi, memastikan kinerja yang responsif dan efisien. Firebase adalah salah satu contoh platform yang menyediakan layanan *database real-time*, memungkinkan pengembang untuk dengan mudah mengimplementasikan sinkronisasi data secara langsung dalam aplikasi mereka.



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritis atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

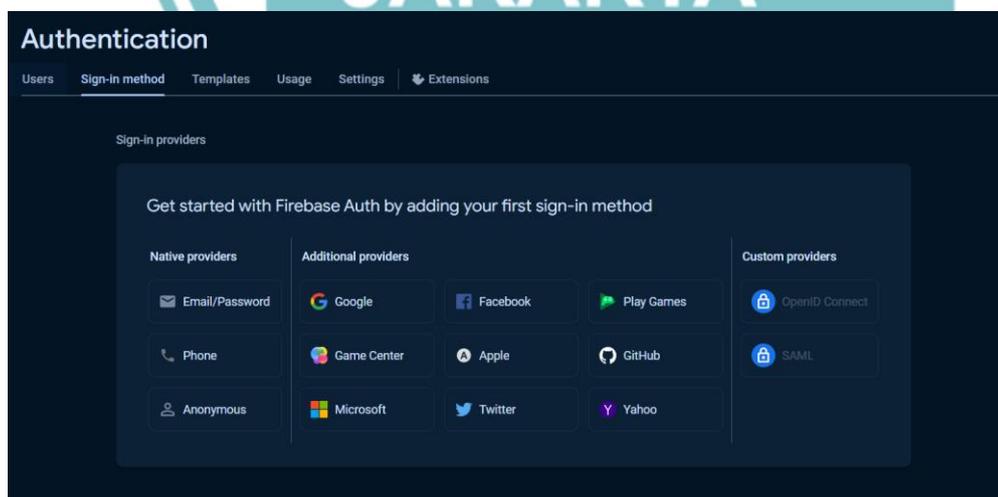


Gambar 3. 6 Halaman *Real-time Database*

Pada Gambar 3.6 menunjukkan bagaimana tampilan pada *real-time database*. *Real-time database* yang digunakan adalah *test mode*. *Test mode* dapat mengakses data dan melakukan perubahan data tanpa *private*.

### 3. Membuat *Authentication*

Firestore *authentication* menyediakan Web API Key yang digunakan untuk mengautentikasi permintaan API dari aplikasi yang dirancang. Web API Key adalah kunci yang memungkinkan aplikasi untuk berkomunikasi dengan layanan Firebase dan memastikan bahwa hanya aplikasi yang dibangun yang dapat mengakses data dan layanan Firebase yang diatur. Jadi penyambungan ke aplikasi dengan menggunakan platform Kodular menggunakan API Key dimana jika sudah masuk ke dalam halaman *authentication*, maka mengaktifkannya dengan cara mengklik “*Get started*”. Pada Gambar 3.7 menunjukkan tampilan pada *authentication*.



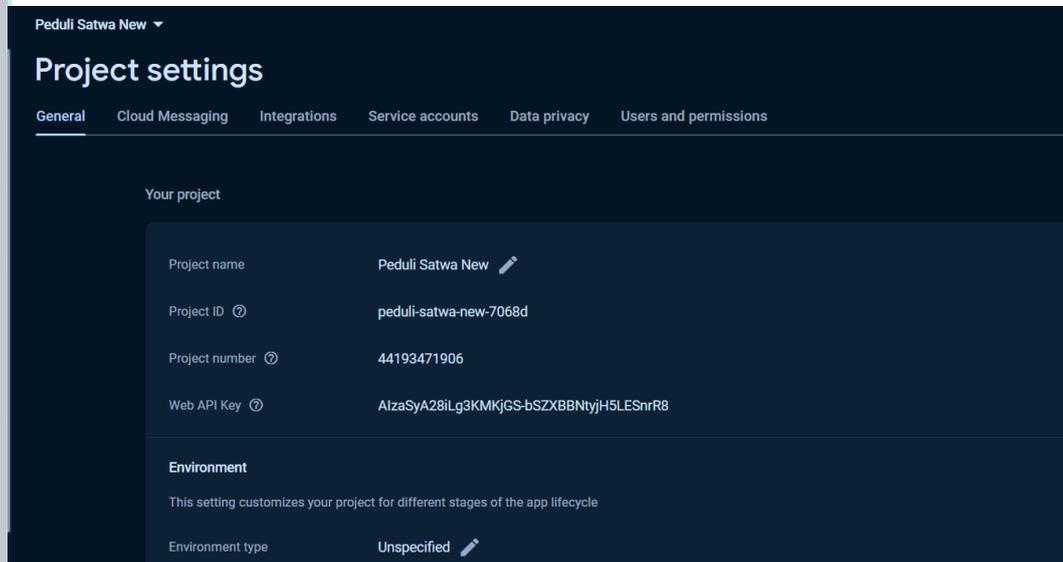
Gambar 3. 7 Tampilan *Auhtentication*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada Gambar 3.8 menunjukkan tampilan *authentication* jika sudah diaktifkan, maka Web API Key akan terlihat di *project setting* pada bagian *general* yang dapat dilihat pada Gambar 3.8.



Gambar 3. 8 *Project Setting*

Gambar 3.8 menunjukkan tampilan pada menu *general* di *project setting* yang menampilkan *code* Web API Key yang dapat digunakan.

### 3.2.2. Penyambungan *Database* ke Program Aplikasi

Menyambungkan *database* ke program aplikasi Android adalah langkah penting untuk memungkinkan aplikasi menyimpan, mengakses, dan mengelola data secara efisien.

#### 1. Menyambungkan Firebase ke aplikasi Android

Pada tampilan awal Firebase, pengguna akan disajikan dengan berbagai pilihan layanan dan fitur. Untuk menyambungkan Firebase dengan aplikasi Android, langkah pertama adalah memilih ikon Android yang ditujukan untuk pendaftaran aplikasi Android. Proses ini akan membawa pengguna ke halaman pendaftaran aplikasi, di mana pengguna harus mengisi informasi detail tentang aplikasi Android yang ingin didaftarkan. Informasi yang diminta meliputi nama paket Android (*Android package name*), yang harus unik dan konsisten dengan nama paket aplikasi yang sudah ada di proyek Android. Selain itu, pengguna juga dapat





## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

projek di Firebase. Gambar 3.10 menunjukkan tampilan properti pada *pallet* Firebase di Kodular.

Gambar 3. 10 Firebase *Database Properties*

Gambar 3.10 menunjukkan tampilan proses penyambungan Firebase ke Kodular. Pada properti *pallet* Firebase terdapat Firebase Token dan Firebase URL yang diperlukan untuk membaca projek Firebase yang akan dihubungkan. Firebase URL dapat ditemukan di tampilan *real-time database*, sedangkan Firebase Token dapat dilihat pada Web API Key. Gambar 3.11 memperlihatkan tampilan *real-time database* di Firebase.



Gambar 3. 11 *Real-time Database*

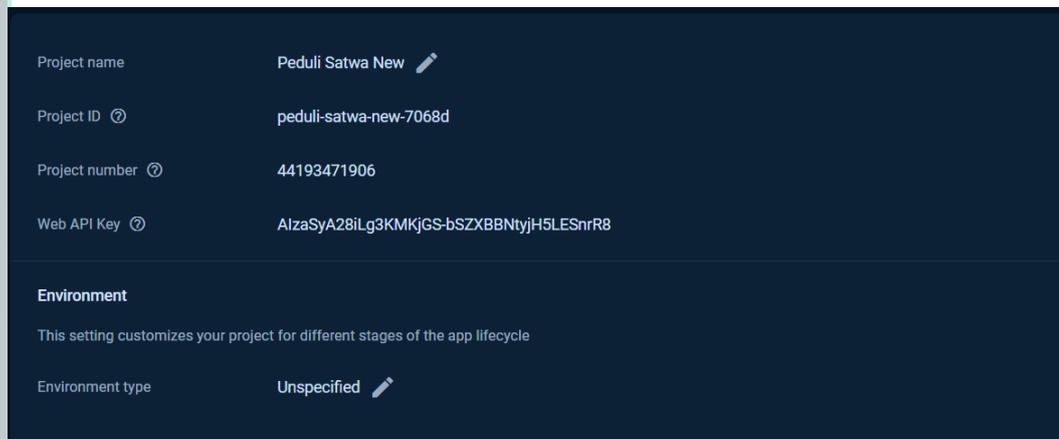
Gambar 3.11 menunjukkan tampilan pada *real-time database* pada Firebase. Pada bagian atas merupakan Firebase URL. Dimana Firebase URL akan diisi pada *properties pallette* Firebase di Kodular sebagai alamat untuk menghubungkan



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Firestore *real-time database* dengan aplikasi Android. Pada Gambar 3.12 menunjukkan tampilan *project setting* pada Firebase



Gambar 3. 12 Tampilan *Project Setting*

Gambar 3.12 menunjukkan tampilan pengaturan proyek di Firebase, yang mencakup Web API Key yang digunakan sebagai kode token Firebase. Token Firebase ini merupakan informasi penting yang diperlukan untuk menghubungkan aplikasi Kodular dengan proyek Firebase Anda. Token ini perlu diisi pada bagian properti Firebase di Kodular untuk memastikan bahwa aplikasi Anda dapat berkomunikasi dengan Firebase.

Untuk mendapatkan Web API Key ini, Anda harus membuat sistem autentikasi terlebih dahulu di Firebase. Proses autentikasi ini penting untuk keamanan dan memastikan bahwa hanya aplikasi yang diizinkan yang dapat mengakses proyek Firebase Anda. Setelah autentikasi dibuat, Web API Key akan muncul di pengaturan proyek Firebase di bawah bagian "*Project Settings*".

### 3.2.3. Perancangan Aplikasi Android

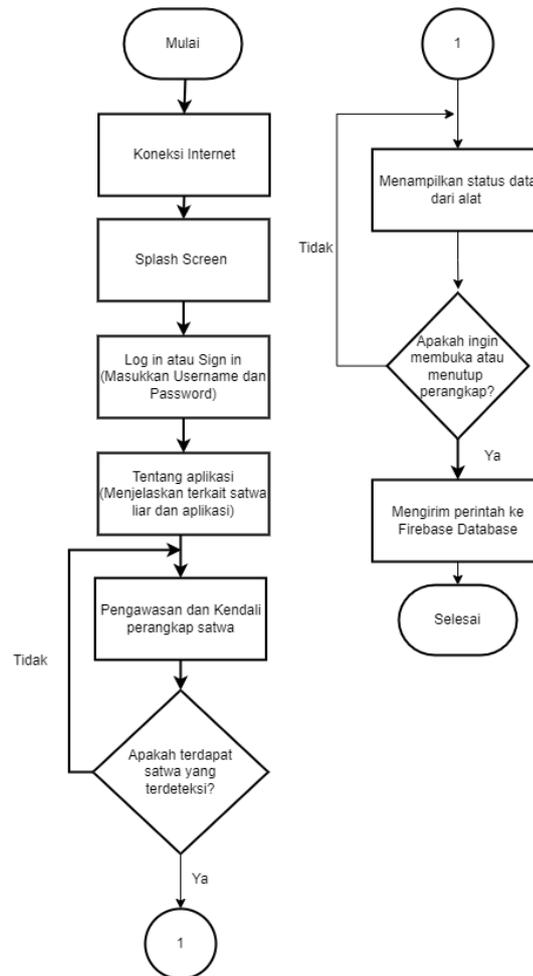
Aplikasi yang dibangun dengan menggunakan platform Kodular dan aplikasi yang dibangun ini diberi nama "Perangkap Satwa". Aplikasi ini difungsikan untuk pemantauan dan kendali perangkap satwa. Pada saat terdapat satwa yang memasuki perangkap, maka pemberitahuan di layar pemantauan akan muncul sebagai kondisi keberadaan satwa. Aplikasi yang dirancang akan digunakan untuk menampilkan hasil kondisi keberadaan saat satwa memasuki perangkap agar mengetahuinya secara *real-time*. *Smartphone* yang sudah mengunduh aplikasi harus terhubung dengan internet agar dapat menerima data berupa status kondisi



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

perangkat dari Firebase dan mengirimkan data berupa perintah ke Firebase. Pada Gambar 3.13 yang merupakan *flowchart* perancangan aplikasi Android.



Gambar 3. 13 *Flowchart* Perancangan Aplikasi

Pada Gambar 3.13 merupakan *flowchart* perancangan aplikasi “Perangkap Satwa” menggunakan Kodular. Aplikasi ini dapat menampilkan hasil status keberadaan satwa dari ESP32 pada penerima yang tersimpan dalam Firebase.

### 3.2.4. Realisasi Program Aplikasi Android

Program aplikasi Perangkap Satwa dirancang dengan menggunakan platform Kodular. Aplikasi ini diberi nama ‘Perangkap Satwa’ karena perancangan aplikasi ini bertujuan untuk memantau satwa liar dengan melihat status keberadaan satwa tersebut di aplikasi dan juga dapat mengendalikan perangkap secara jarak jauh melalui aplikasi. Untuk dapat mengakses aplikasi ‘Perangkap Satwa’ *Smartphone* harus terhubung ke jaringan internet agar dapat memantau keberadaan satwa.



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritis atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 1. Membuat tampilan *Splash Screen*

*Splash screen* adalah tampilan awal yang muncul ketika sebuah aplikasi atau perangkat lunak dijalankan untuk pertama kalinya. Tampilan *splash screen* hanya muncul selama beberapa detik, cukup lama untuk menampilkan logo, nama aplikasi, atau elemen grafis lainnya sambil menunggu proses pemuatan aplikasi selesai. Fungsi utama dari *splash screen* adalah untuk memberikan kesan pertama yang baik dan menarik bagi pengguna, sekaligus memberikan waktu bagi aplikasi untuk melakukan inisialisasi. *Splash screen* tidak hanya berfungsi untuk tampilan pengenalan saja, tetapi untuk memberi waktu bagi aplikasi untuk memuat semua yang diperlukan di latar belakang. Gambar 3.14 merupakan *Splash Screen* dari aplikasi 'Perangkap Satwa'.



Gambar 3. 14 *Splash Screen* Aplikasi

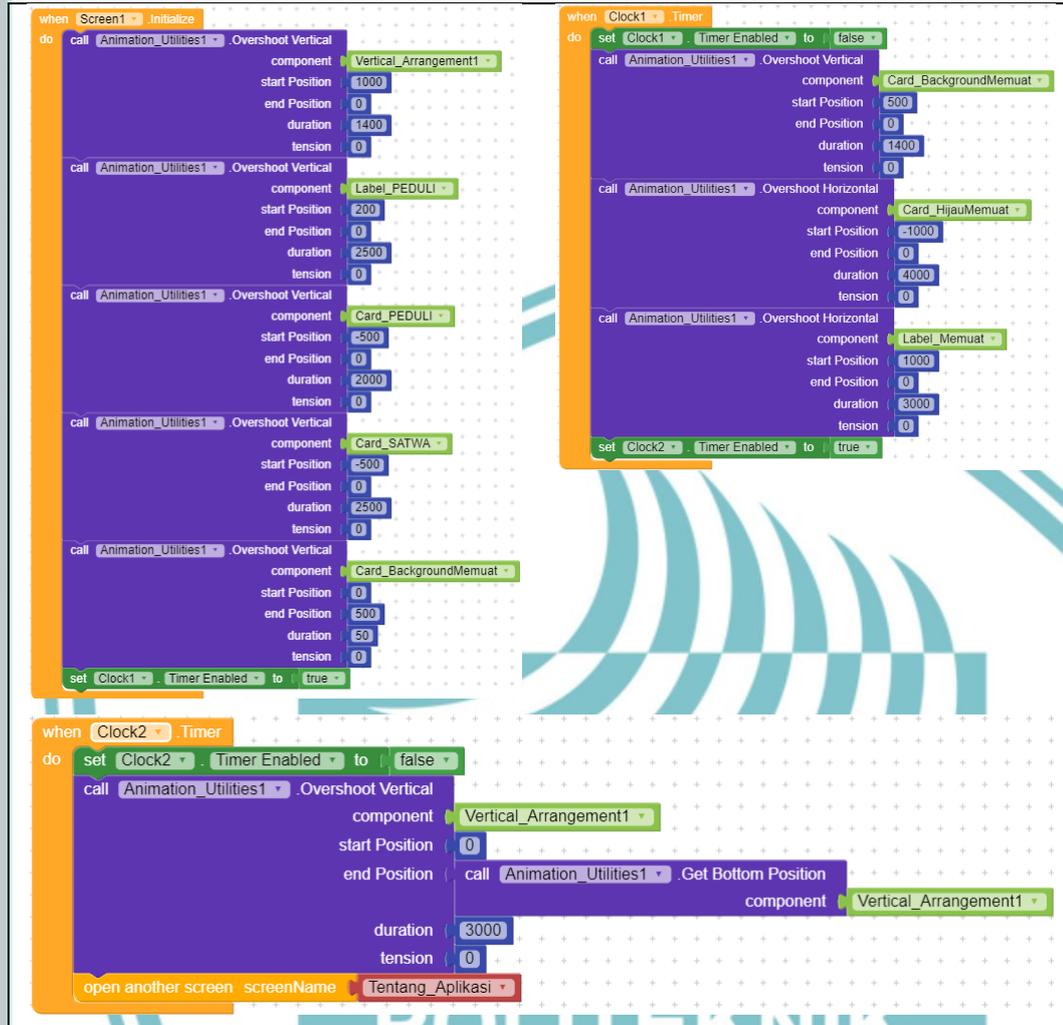
Berdasarkan Gambar 3.14 terlihat tampilan *splash screen* yang dibuat dengan menggunakan beberapa *pallet*. *Splash screen* adalah layar pembuka yang ditampilkan ketika aplikasi pertama kali dijalankan, berfungsi untuk memberikan kesan awal yang menarik kepada pengguna. Pemilihan *pallet* untuk tampilan ini dilakukan dengan memilih opsi *pallet* pada bagian *designer* di halaman perancangan antarmuka aplikasi.



## Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 3. 15 Sketch Splash Screen

Berdasarkan *sketch* di atas mengatur tampilan *splash screen* yang digunakan untuk membuat animasi pada komponen UI saat *splash screen* diinisialisasi. Saat layar diinisialisasi, serangkaian animasi akan dijalankan pada beberapa komponen menggunakan metode ‘*OvershootVertical*’ dari ‘*Animation\_Uilities1*’. Pertama, *Vertical\_Arrangement1* dipindahkan dari posisi 1000 ke 1400 secara instan. Kemudian, komponen PEDULI1 bergerak dari posisi 200 ke 0 selama 2,5 detik, diikuti oleh *Card\_View1* yang bergerak dari posisi -500 ke 0 dalam 2 detik. Selanjutnya, *Card\_View2* juga bergerak dari posisi -500 ke 0 tetapi dalam 2,5 detik. Terakhir, *Card\_View3* dipindahkan dari posisi 0 ke 500 dalam waktu yang sangat cepat, yaitu 50 ms. Setelah semua animasi ini diatur, *Clock1* diaktifkan yang mungkin akan digunakan untuk memicu tindakan lebih lanjut setelah animasi selesai.

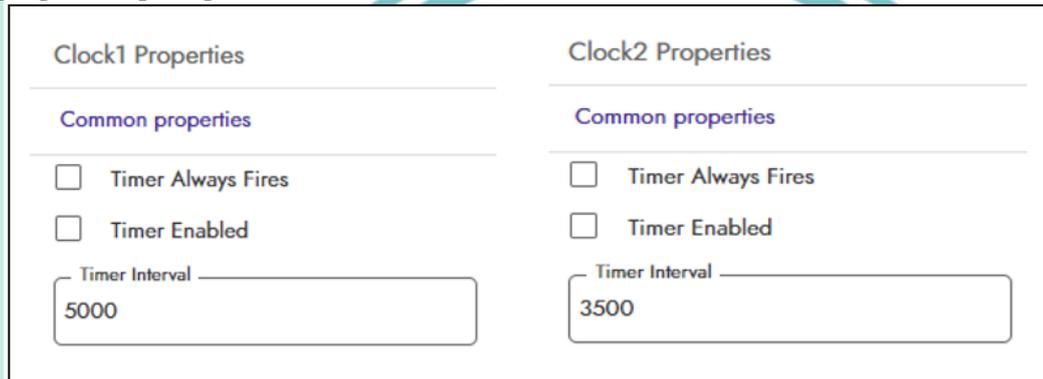


## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada blok kode selanjutnya ini digunakan untuk mengatur serangkaian animasi pada beberapa komponen UI setelah *timer* diaktifkan. Saat *timer* 'Clock1' berjalan, beberapa komponen akan dipindahkan ke posisi baru dengan durasi dan arah yang berbeda. Setelah semua animasi selesai, *timer* 'Clock2' diaktifkan untuk menjalankan animasi tambahan pada 'Vertical\_Arrangement1' sebelum akhirnya membuka layar baru bernama 'About\_Aplication'. Pada Gambar 3.16 menunjukkan *properties* pada *pallette* Clock1 dan Clock2.



Gambar 3. 16 *Pallette* Clock 1 dan Clock 2

Gambar 3.16 menunjukkan pengaturan properti pada dua elemen jam (Clock1 dan Clock2) yang digunakan dalam aplikasi "Perangkap Satwa". Properti yang paling penting untuk diperhatikan adalah Time Interval, yang harus disesuaikan dengan durasi yang diperlukan pada kode block untuk menjalankan berbagai proses.

Pada *Clock1*, *Time* Interval diatur menjadi 5 detik. Interval ini dipilih untuk mencocokkan durasi animasi *loading* pada *splash screen*, memberikan waktu yang cukup bagi pengguna untuk melihat tampilan awal aplikasi sebelum diarahkan ke layar berikutnya. *Clock1* bertanggung jawab memastikan *splash screen* ditampilkan dengan baik selama waktu yang ditentukan sebelum transisi ke menu masuk.

Sementara itu, *Clock2* memiliki *Time* Interval yang diatur menjadi 3,5 detik. Interval ini digunakan untuk menunda pembukaan layar berikutnya setelah *splash screen*. Setelah *Clock2* mencapai interval waktu yang ditentukan, layar selanjutnya akan dibuka, memungkinkan transisi yang mulus antara *splash screen* dan layar masuk atau layar utama aplikasi.



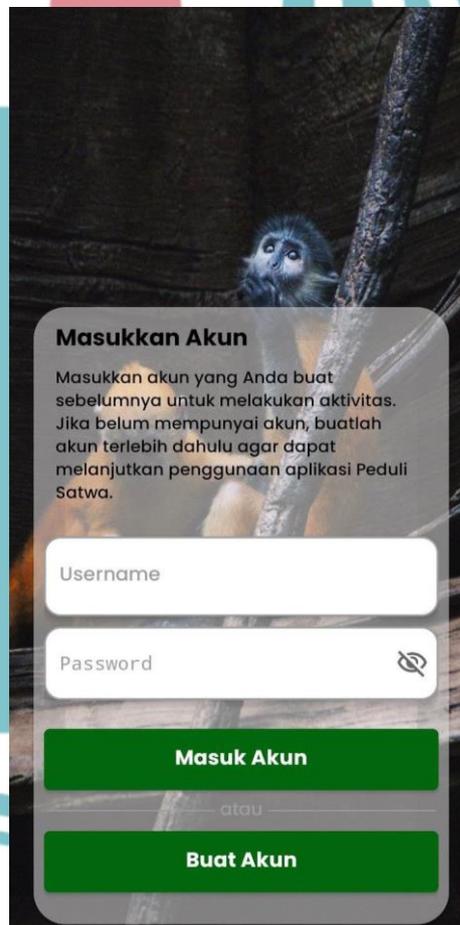
## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## 2. Membuat tampilan menu masuk

Setelah splash screen selesai ditampilkan, aplikasi "Perangkap Satwa" akan menampilkan layar masuk sebagai langkah awal interaksi pengguna dengan aplikasi. Layar masuk ini berfungsi sebagai gerbang utama bagi pengguna untuk mengakses fitur-fitur aplikasi lebih lanjut. Untuk mewujudkan tampilan ini, diperlukan pembuatan layar baru yang khusus didedikasikan sebagai menu masuk. Pada layar ini, pengguna akan disajikan dengan opsi untuk masuk atau mendaftar, serta elemen-elemen lain yang mendukung proses autentikasi, seperti input nama pengguna dan kata sandi. Gambar 3.17 menunjukkan tampilan menu masuk pada aplikasi.



Gambar 3. 17 Tampilan Masuk Akun

Berdasarkan Gambar 3.17 menunjukkan tampilan menu masuk pada aplikasi. Terdapat *pallette text box* untuk memasukkan *username* dan *password*, kemudian tombol 'Masuk Akun' beserta 'Buat Akun'. Tombol masuk jika sudah memiliki akun sehingga bisa langsung masuk ke *screen* selanjutnya. Untuk tombol Buat

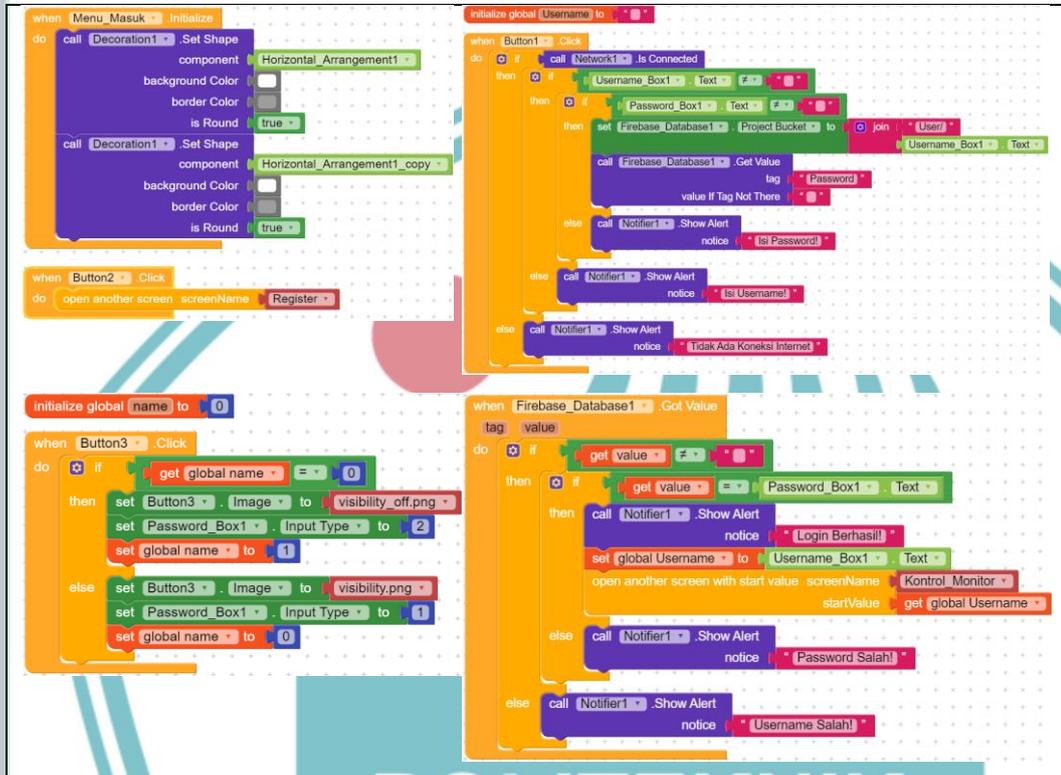


## Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Akun apabila pengguna belum memiliki akun, pengguna harus mendaftarkan akunnya terlebih dahulu agar bisa menggunakan aplikasi “Perangkap Satwa”. Pada Gambar 3.18 menunjukkan *sketch* untuk tampilan Masuk digunakan program blok seperti pada Gambar 3.18.



Gambar 3. 18 *Sketch* Masuk Akun

Gambar blok pemrograman di atas menunjukkan aplikasi yang dirancang untuk sistem *login* sederhana dengan koneksi ke *Firestore Database*. Ketika tombol *login* (*Button1*) diklik, aplikasi pertama-tama mengecek koneksi internet. Jika terhubung, aplikasi akan mengambil *input username* dan *password* dari pengguna, kemudian mengatur *Project Bucket* di *Firestore* sesuai dengan *username* yang dimasukkan. Selanjutnya, aplikasi mengambil nilai *password* dari *Firestore* untuk *username* tersebut dan membandingkannya dengan *input password* yang diberikan pengguna. Jika *password* cocok, aplikasi menampilkan pesan "*Login Berhasil*", menyimpan *username* ke dalam *variabel global*, dan membuka layar baru untuk *monitoring*. Jika tidak cocok, aplikasi akan menampilkan notifikasi kesalahan sesuai dengan kondisi yang terjadi, seperti "*Password Salah*" atau "*Username Salah*".



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , pennisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Selain itu, pada saat layar `Menu\_Masuk` diinisialisasi, aplikasi mengatur tampilan dua komponen visual agar memiliki warna latar belakang yang berbeda dengan sudut membulat, untuk mempercantik antarmuka pengguna. Terdapat juga fitur untuk beralih antara menampilkan dan menyembunyikan *password* saat tombol (`Button3`) diklik. Jika *password* ditampilkan, ikon tombol akan berubah dan sebaliknya. Aplikasi ini juga menyediakan opsi untuk membuka layar pendaftaran baru (`Register`) saat tombol pendaftaran (`Button2`) diklik.

Sebelum masuk ke dalam aplikasi, pengguna harus membuat akun terlebih dahulu dengan mendaftarkannya pada *screen* Membuat Akun. Gambar 3.19 menunjukkan tampilan Membuat Akun pada aplikasi.

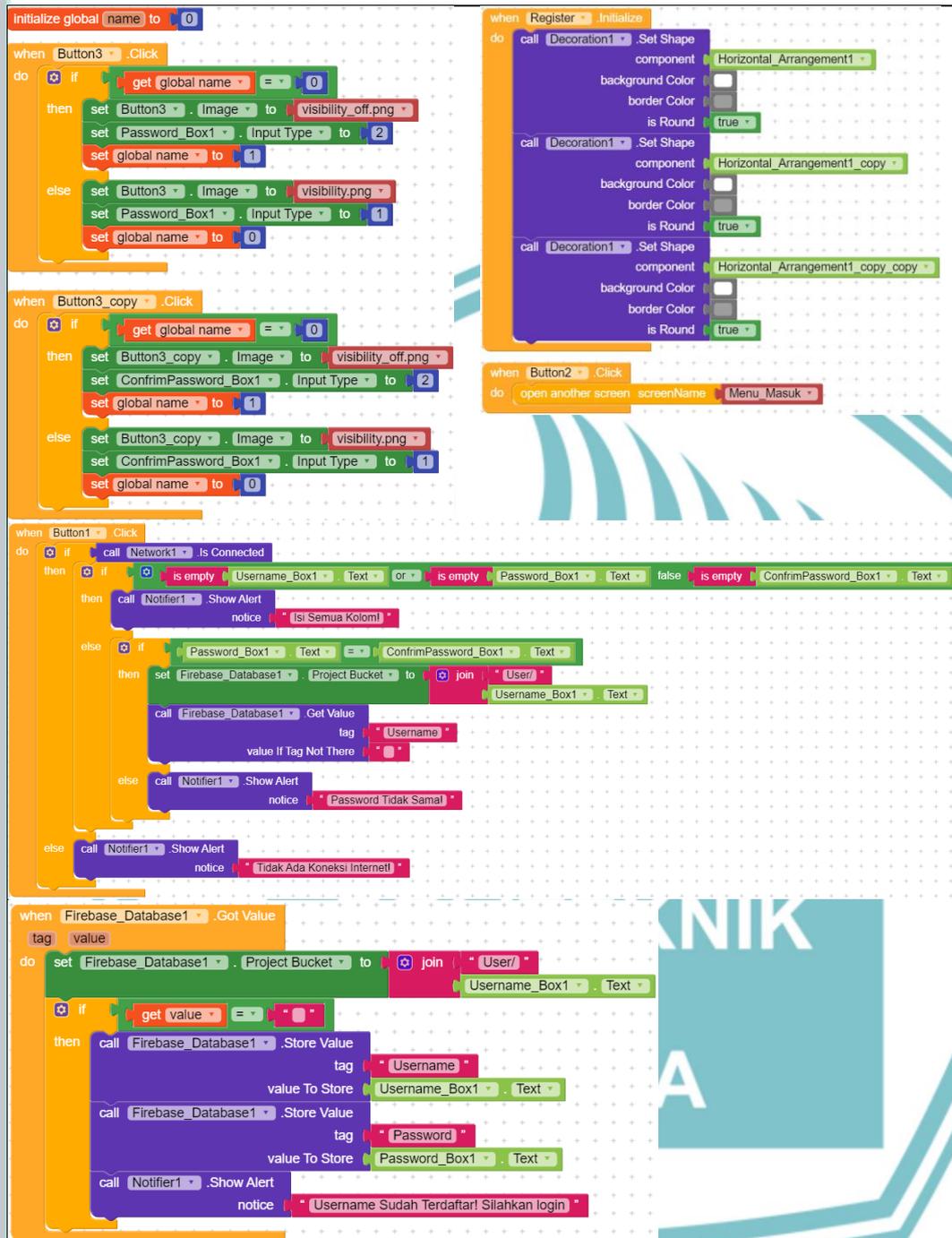
Gambar 3. 19 Tampilan Daftar Akun

Berdasarkan Gambar 3.19 menunjukkan tampilan Membuat Akun pada aplikasi. Terdapat *pallette text box* untuk membuat *username* dan *password*, kemudian tombol “Buat Akun” difungsikan untuk mendaftarkan akun yang dibuat. Pada Gambar 3.20 menunjukkan *sketch* untuk tampilan daftar akun, program blok dapat dilihat seperti pada Gambar 3.20.



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 3. 20 Sketch Daftar Akun

Gambar blok kode di atas menunjukkan logika untuk sebuah aplikasi registrasi pengguna yang dirancang menggunakan Kodular. Saat tombol registrasi (*Button1*) diklik, aplikasi terlebih dahulu memeriksa koneksi internet. Jika terhubung, aplikasi akan memeriksa apakah kolom *Username\_Box1*, *Password\_Box1*, dan *ConfirmPassword\_Box1* sudah terisi. Jika salah satu kolom kosong, aplikasi menampilkan pesan peringatan. Jika semua kolom terisi,



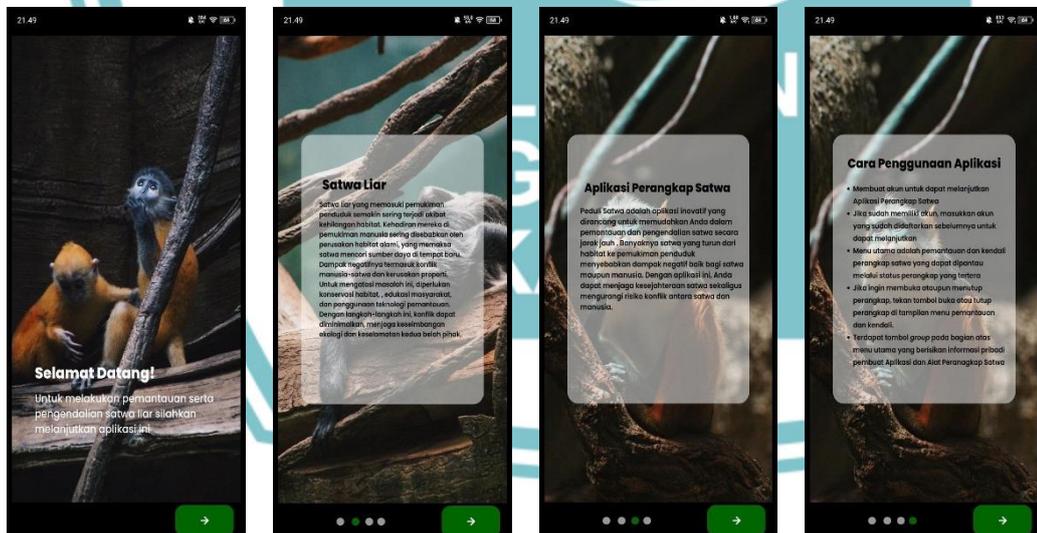
## © Hak Cipta milik Politeknik Negeri Jakarta

aplikasi akan memeriksa kesamaan antara `Password\_Box1` dan `ConfirmPassword\_Box1`. Jika kedua *input password* cocok, aplikasi mengambil nilai dari *Firestore database* untuk memeriksa apakah *username* sudah terdaftar. Jika *username* belum terdaftar, aplikasi menyimpan *username* dan *password* baru tersebut ke dalam *database*, dan menampilkan pesan bahwa registrasi berhasil. Jika *username* sudah ada, pesan peringatan akan ditampilkan.

Selain itu, terdapat dua tombol (`Button3` dan `Button3\_copy`) yang masing-masing digunakan untuk menampilkan dan menyembunyikan *password*. Saat tombol ini diklik, *input tipe password* akan berganti antara mode terlihat dan tersembunyi, dengan ikon tombol juga berubah sesuai statusnya. Terdapat juga blok kode untuk mengatur tampilan visual komponen ketika layar `Register` diinisialisasi, dengan menambahkan sudut yang membulat pada komponen tertentu untuk mempercantik antarmuka. Tombol `Button2` digunakan untuk berpindah ke layar lain yang bernama "Menu\_Masuk" setelah proses registrasi selesai.

### 3. Membuat tampilan tentang aplikasi

Tampilan 'Tentang Aplikasi' berisi pemberitahuan mengenai aplikasi yang ingin digunakan. Gambar 3.21 menunjukkan tampilan 'Tentang Aplikasi' pada aplikasi.



(a)

(b)

(c)

(d)

Gambar 3. 21 (a) Tampilan Selamat Datang (b) Pemberitahuan Tentang Satwa Liar (c) Pemberitahuan Mengenai Aplikasi Peduli Satwa (d) Cara Penggunaan Aplikasi

### Hak Cipta :

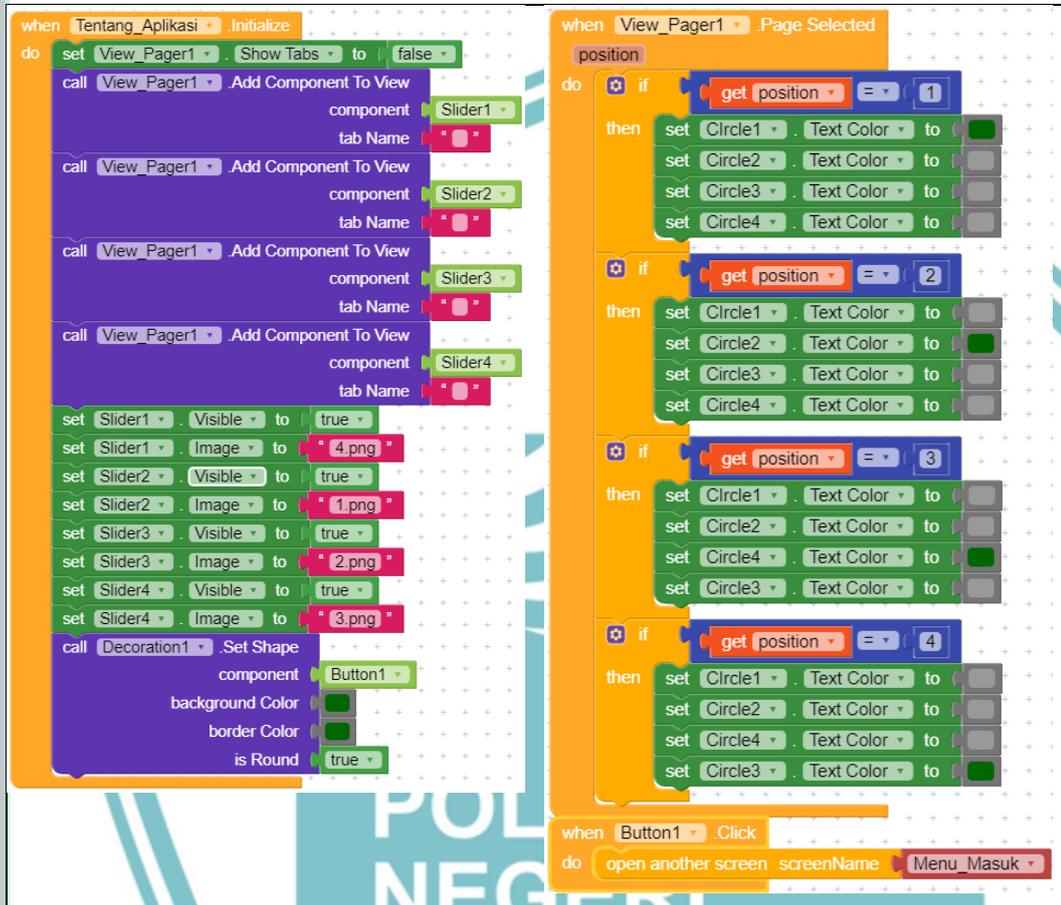
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Tampilan Tentang Aplikasi akan menampilkan tombol “Mulai” untuk memasuki halaman utama yaitu *screen* “Pemantauan dan Kendali Satwa”. Gambar 3.22 merupakan *sketch* untuk tampilan Tentang Aplikasi, dapat dilihat program *block* pada gambar.



Gambar 3. 22 *Sketch* Tentang Aplikasi

Gambar 3.22 merupakan *sketch* tentang aplikasi, dan berfungsi untuk mengatur tampilan dan interaksi di layar "Tentang Aplikasi". Pada saat layar ini diinisialisasi, kode ini pertama-tama menyembunyikan tab di dalam komponen `View\_Pager1`, lalu menambahkan tiga komponen `Slider` (*Slider1*, *Slider2*, *Slider3* dan *Slider4*) ke dalam `View\_Pager1` sebagai tab dengan nama "0". Gambar yang ditampilkan pada setiap *slider* diatur dengan file gambar tertentu, dan ketiga *slider* tersebut diatur agar terlihat di layar. Selain itu, kode ini juga mengubah tampilan `Button1`, termasuk warna latar belakang, warna garis batas, dan membuatnya berbentuk bulat.

Kemudian, saat pengguna menggeser halaman di `View\_Pager1`, kode ini akan mengecek posisi halaman yang sedang dipilih dan mengubah warna teks pada tiga



## © Hak Cipta milik Politeknik Negeri Jakarta

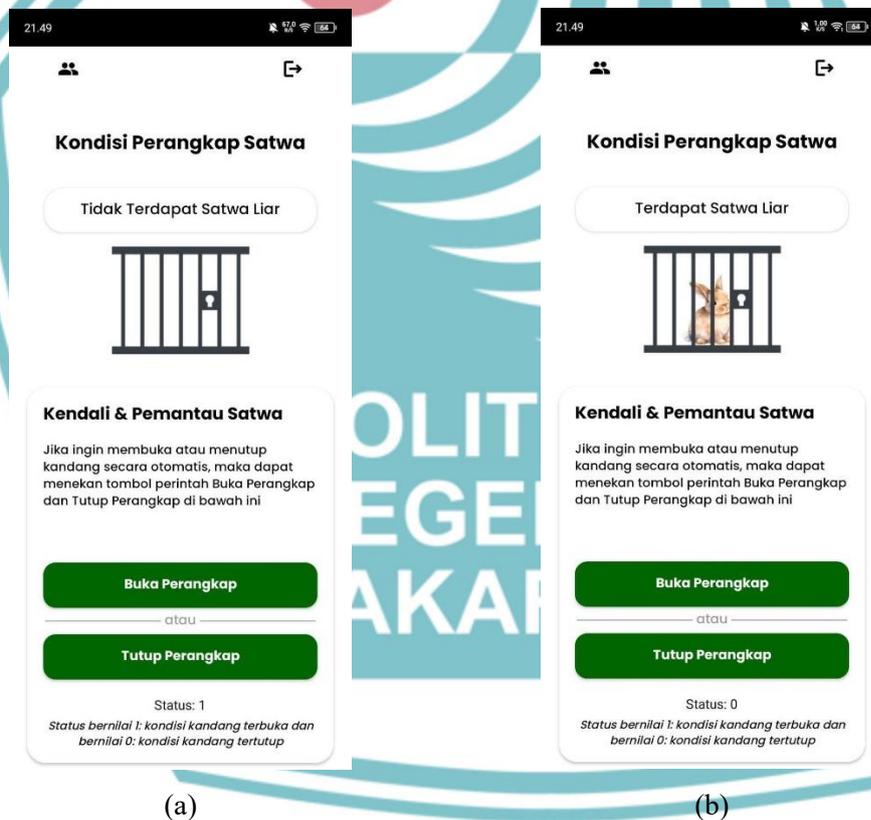
### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

komponen lingkaran (`Circle1`, `Circle2`, `Circle3`, dan `Circle4`) sesuai dengan halaman yang dipilih, sehingga memberikan indikasi visual kepada pengguna tentang halaman mana yang sedang ditampilkan. Terakhir, kode ini juga mengatur fungsi `Button1`, di mana saat tombol tersebut ditekan, aplikasi akan membuka layar baru yang bernama `Menu\_Masuk`.

#### 4. Membuat tampilan kendali dan pemantauan

Tampilan pemantauan satwa pada aplikasi ini menampilkan status yang dikirimkan dari Firebase secara *real-time*. Selain itu, pada tampilan aplikasi ini juga memungkinkan pengguna untuk mengendalikan perangkat satwa secara jarak jauh. Gambar 3.23 menunjukkan antarmuka Pemantauan dan Kendali pada aplikasi tersebut.



Gambar 3. 23 (a) Kondisi Perangkap Terbuka (b) Kondisi Perangkap Tertutup

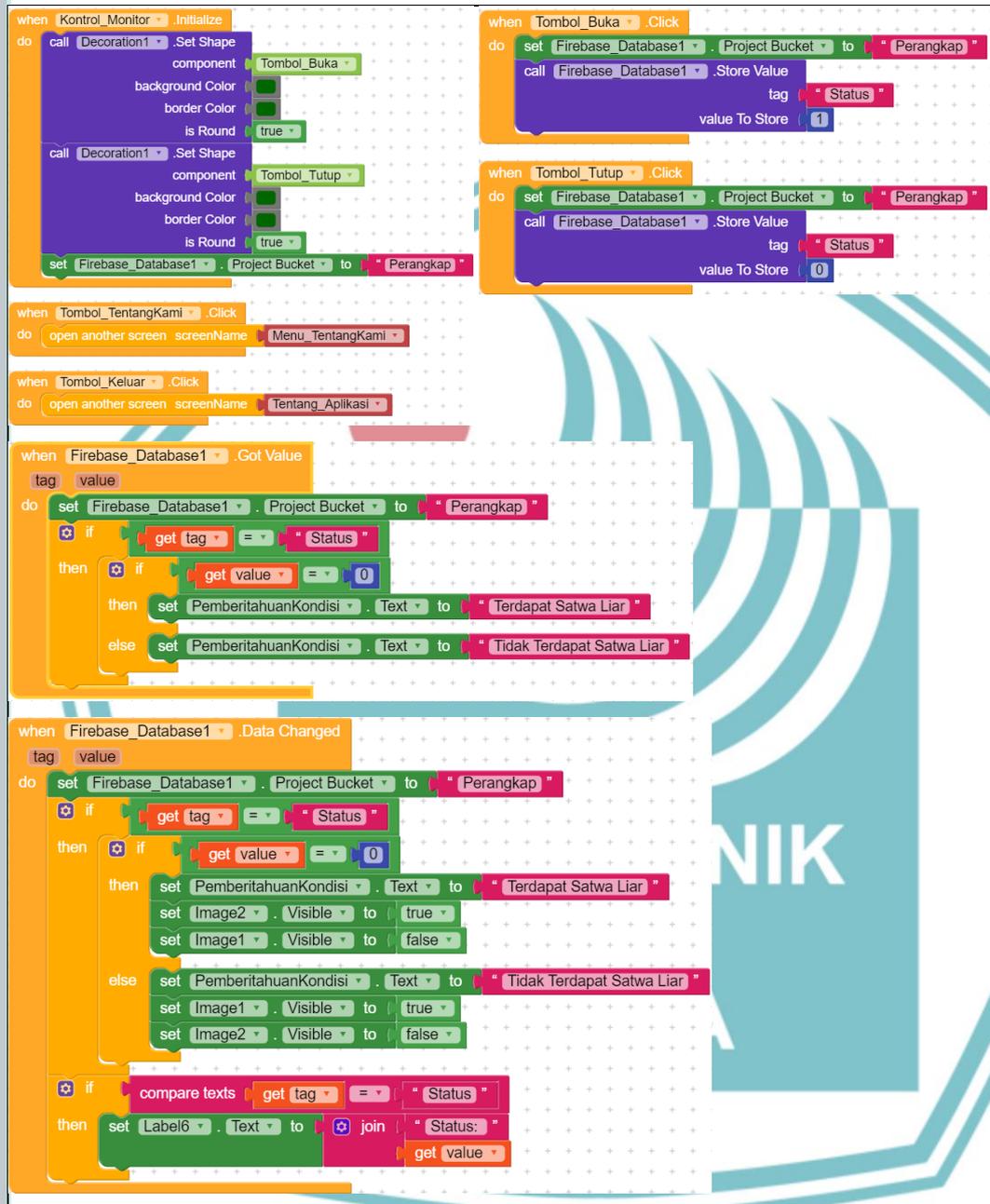
Tampilan pemantauan dan kendali akan menampilkan hasil dari sensor ultrasonik dan akan mengirimkan perintah ke alat untuk menggerakkan servo. Pada pemantauan dapat dilihat kondisi terkini kandang apakah terdapat satwa yang terperangkap dan tampilan menu pemantauan akan menampilkan keberadaan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

satwa. Gambar 3.24 merupakan *sketch* untuk tampilan Kendali dan Pemantauan dapat dilihat program *block* pada gambar.



Gambar 3. 24 Sketch Kendali dan Pemantauan

Gambar 3.24 menggambarkan logika untuk kendali dan pemantauan status perangkap satwa liar menggunakan Firebase *database*. Saat layar `Kontrol\_Monitor` diinisialisasi, aplikasi mengatur tampilan tombol dengan sudut membulat dan menghubungkan *bucket* Firebase ke proyek bernama "Perangkap." Terdapat dua tombol utama, `Tombol\_Buka` dan `Tombol\_Tutup`, yang masing-masing digunakan untuk mengirimkan nilai `1` dan `0` ke Firebase dengan tag



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

`Status`. Nilai `1` menandakan bahwa perangkat dibuka, sementara nilai `0` menandakan perangkat ditutup.

Ketika aplikasi menerima perubahan data dari Firebase, ia memeriksa tag dan nilai yang diterima. Jika tag `Status` memiliki nilai `0`, maka teks pada `PemberitahuanKondisi` akan menampilkan pesan "Tidak Terdapat Satwa Liar" dan menyembunyikan gambar `Image2`, sambil menampilkan `Image1`. Sebaliknya, jika nilainya `1`, pesan "Terdapat Satwa Liar" akan muncul, dan gambar `Image2` akan terlihat sementara `Image1` disembunyikan. Terdapat juga blok kode untuk mengupdate label `Label6` dengan teks status yang diterima dari Firebase, memastikan informasi status ditampilkan dengan benar kepada pengguna.

Selain itu, aplikasi menyediakan dua tombol navigasi tambahan, `Tombol\_TentangKami` dan `Tombol\_Keluar`, yang digunakan untuk berpindah ke layar "Menu\_TentangKami" dan "Tentang\_Aplikasi". Secara keseluruhan, blok kode membentuk sistem pemantauan yang responsif terhadap perubahan kondisi perangkat satwa liar dan memberikan umpan balik secara *real-time* kepada pengguna melalui Firebase.

### 3.2.5. Penyambungan Alat ke Database Firebase

Bagian pengirim pada alat akan mengirimkan data ke bagian penerima melalui sistem transmisi LoRa. Setelah data diterima, bagian penerima kemudian akan mengirimkannya ke *database* melalui jaringan internet. Koneksi ini sangat penting karena memungkinkan alat dan aplikasi dapat berkomunikasi satu sama lain dengan lancar. Pada pengirim dan penerima penggunaan pin untuk transmisi lora tersebut dapat dilihat pada Tabel 3.2

Tabel 3.2 Penggunaan pin Transmisi Lora pada pengirim dan penerima

Lora RFM95	Pin Komponen
NSS	5
RST	27
DIO0	26
SCK	18
MISO	19
MOSI	23



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Bedasarkan tabel diatas, menunjukkan pin Lora RFM95 yang digunakan dalam sistem transmisi. NSS (5) berfungsi sebagai seleksi slave, mengaktifkan komunikasi SPI. Pin RST (27) digunakan untuk mereset modul, dan DIO0 (26) berfungsi sebagai pin interrupt yang mengindikasikan status data. SCK (18) adalah pin clock SPI, sementara MISO (19) dan MOSI (23) mengatur jalur data masuk dan keluar dalam komunikasi.

Setelah memastikan koneksi hardware sudah benar, langkah selanjutnya adalah menghubungkan penerima ke Firebase. Hal ini dilakukan dengan memprogram esp32 pada penima agar dapat mengirim data dari modul LoRa ke Firebase melalui internet, menggunakan API Firebase.

1. Menyambungkan ESP32 penerima ke Firebase

Sebelum ESP32 terhubung ke Firebase, ESP32 perlu disambungkan ke internet terlebih dahulu. Hal ini dapat dilakukan dengan memasukan program seperti pada Gambar 3.25.

```
// Insert your network credentials
#define WIFI_SSID "samsunga32"
#define WIFI_PASSWORD "qweqwe00"
```

Gambar 3.25 Menyambungkan ESP32 dengan Internet

Setelah ESP32 terhubung pada internet, selanjutnya dengan menghubungkan ke Firebase untuk memulai interaksi dengan *real-time database* seperti pada Gambar 3.26.

```
// Insert Firebase project API Key
#define API_KEY "AIzaSyA28iLg3KMKjGS-bSZXBNTyjH5LESnrR8"

// Insert RTDB URL
#define DATABASE_URL "https://peduli-satwa-new-7068d-default-rtdb.firebaseio.com/"
```

Gambar 3.26 Menyambungkan ESP32 dengan Firebase

Pada Gambar di atas API\_KEY” yang diberikan oleh Firebase berfungsi untuk otentikasi dan memberikan akses kepada ESP32 untuk berinteraksi dengan layanan Firebase. Selanjutnya, “DATABASE\_URL” berfungsi untuk mengarahkan Lokasi basis data *real-time* Firebase yang akan diakses oleh perangkat ESP32. Firebase *database* digunakan untuk mengelola data yang dikirimkan atau diterima dari Firebase. FirebaseAuth menangani proses otentikasi dengan Firebase, memastikan hanya pengguna yang terverifikasi yang dapat mengakses layanan.

Sementara itu, Firebase Config menyimpan semua konfigurasi yang diperlukan untuk menghubungkan ESP32 ke Firebase, termasuk API key dan URL *database*.



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB IV PEMBAHASAN

Setelah melakukan perancangan dan realisasi sistem pada android, proses selanjutnya adalah melakukan pengujian alat. Proses ini merupakan tahapan akhir yang dilakukan dalam pembuatan tugas akhir ini. Pengujian ini dilakukan untuk mengetahui alat dapat berfungsi dengan baik atau tidak. Pengujian dilaksanakan berdasarkan lokasi dan waktu sebagai berikut:

Lokasi	: Beji Timur, Depok
Waktu	: Agustus 2024
Pelaksanaan	: Rizka Dwi Sakha
Pembimbing	: Rifqi Fuadi Hasani, S.T., M.T.

Berikut ini merupakan pengujian yang dilakukan untuk tahap terakhir pembuatan tugas akhir:

1. Pengujian aplikasi Android
2. Pengujian *Quality of Service* (QoS)
3. Pengujian aplikasi dengan *speedtest*

### 4.1 Pengujian Aplikasi Android

Pengujian ini dilakukan untuk membuktikan bahwa aplikasi android dapat terhubung dengan Firebase dan mikrokontroler, yaitu ESP32. Data-data dikirimkan dari ESP32 melalui LoRa RFM95W. Hasil pembacaan pada aplikasi android akan menampilkan status kondisi dari keberadaan satwa yang telah dideteksi oleh sensor ultrasonik sehingga aplikasi android dapat memantau satwa serta dapat mengendalikan perangkat secara jarak jauh.

#### 4.1.1. Deskripsi Pengujian Aplikasi Android

Pengujian dilakukan dengan memasukkan hewan ke dalam perangkat agar sensor ultrasonik dapat mendeteksi keberadaan hewan tersebut dan servo akan secara otomatis menutup perangkat, jika perangkat ingin dibuka secara otomatis dalam jarak jauh aplikasi Android Perangkat Satwa dapat menjalankan perintah dan hasil pemantauan akan ditampilkan pada aplikasi Android. Perangkat yang digunakan dalam pengujian aplikasi Android ini meliputi beberapa komponen utama, yaitu *Smartphone*, aplikasi Android ‘Perangkat Satwa’, dan Firebase *Real-time Database*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

#### 4.1.2. Prosedur Pengujian

Proses pengujian aplikasi Android terdiri dari beberapa tahapan yang sistematis. Terdapat beberapa tahapan yang dilakukan dalam pengujian aplikasi Android.

##### 4.1.2.1. Prosedur Pengujian *Set Up* Alat

Prosedur pengujian *set up* alat adalah langkah-langkah yang dilakukan untuk memastikan bahwa semua komponen alat yang dirancang bekerja dengan benar dan sesuai dengan spesifikasi yang telah ditentukan.

1. Menghubungkan perangkat Android dengan jaringan internet untuk memastikan akses data, seperti pada Gambar 4.1.



Gambar 4. 1 Menghubungkan Android dengan Jaringan Internet

2. Mengunduh aplikasi “Perangkap Satwa” dari platform Kodular dengan cara melakukan *scan* menggunakan *Smartphone*, seperti pada Gambar 4.2



Gambar 4. 2 *Scan Barcode* Menggunakan *Smartphone*

3. Setelah mengunduh aplikasi “Perangkap Satwa”, pengguna dapat membuka aplikasi dan mendaftarkan akun terlebih dahulu jika belum memiliki akun.

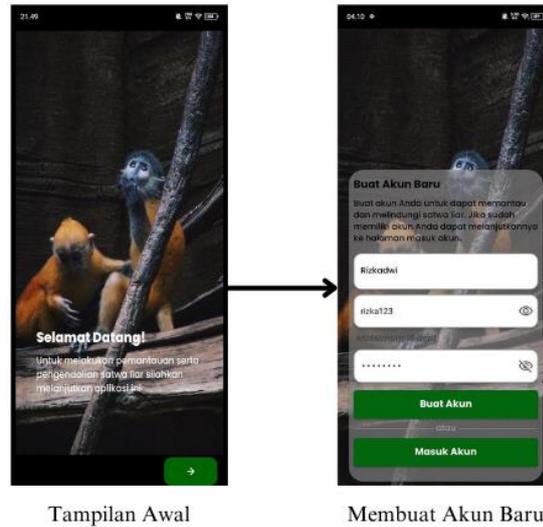


## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

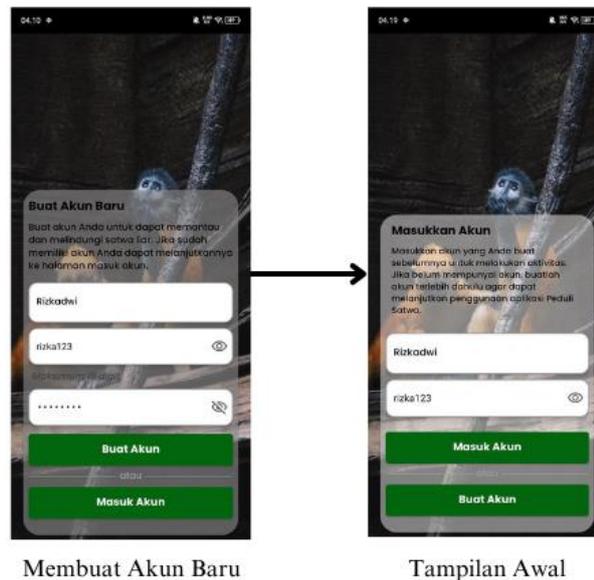
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Dengan cara membuat *username* dengan nama yang unik dan membuat *password*. Gambar 4.3 merupakan ilustrasi membuat akun baru.



Gambar 4. 3 Tahapan Membuat Akun Baru

4. Pengguna yang sudah memiliki akun, dapat langsung masuk ke dalam akun pengguna yang telah terdaftar dengan cara menekan tombol ‘Masuk Akun’, kemudian memasukkan *username* dan *password* yang telah dibuat sebelumnya.



Gambar 4. 4 Tahapan Masuk Akun

5. Pengguna mengakses aplikasi untuk melihat hasil pemantauan mengenai kondisi lingkungan dari keberadaan satwa secara jarak jauh.

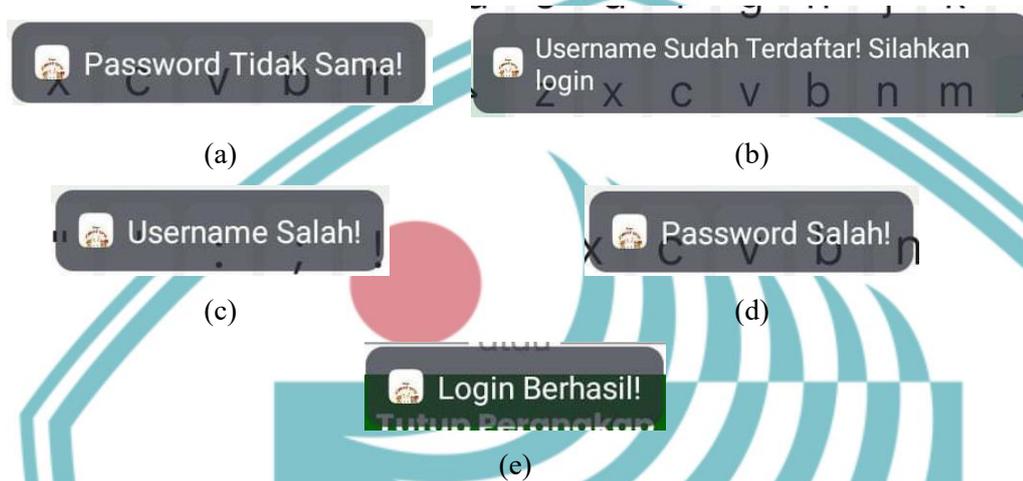


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

#### 4.1.2.2. Hasil Pengujian Pembuatan dan Masuk Akun

Pelindung satwa atau pengurus satwa dan lingkungan yang ingin melakukan pemantauan dan pengendalian perangkat harus memiliki akun. Tersedia menu untuk mendaftarkan akun untuk membuat akun dan masuk akun agar dapat melanjutkan aplikasi.



Gambar 4. 5 Pengujian Pembuatan Akun (a) *Password* Tidak Sama (b) *Username* Sudah Terdaftar! Silahkan *Login* (c) *Username* Salah (d) *Password* Salah (e) *Login* Berhasil

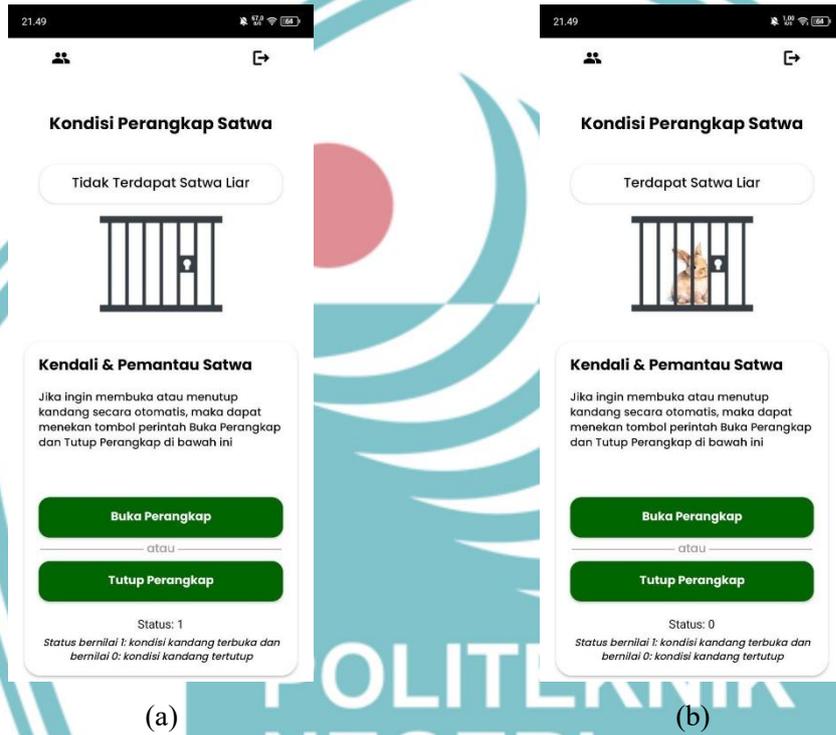
Pada Gambar 4.5 dimana menampilkan kumpulan notifikasi yang berfungsi sebagai umpan balik pada proses validasi pengguna dalam sebuah aplikasi, terutama terkait dengan prosedur login dan pendaftaran. Setiap notifikasi ini muncul sebagai hasil dari tindakan spesifik yang dilakukan oleh pengguna. Misalnya, notifikasi "*Password* Tidak Sama!" akan muncul ketika pengguna memasukkan dua password yang berbeda saat proses pendaftaran atau perubahan password, menunjukkan bahwa kedua *input* tersebut tidak cocok. Notifikasi "*Password* Salah!" dan "*Username* Salah!" ditampilkan ketika pengguna mencoba untuk login dengan informasi yang tidak sesuai dengan data yang tersimpan dalam sistem aplikasi. Sebaliknya, notifikasi "*Login* Berhasil!" mengindikasikan bahwa pengguna telah berhasil masuk ke dalam aplikasi setelah memasukkan informasi yang benar. Selain itu, notifikasi "*Username* Sudah Terdaftar! Silahkan *login*" muncul ketika pengguna mencoba mendaftarkan dengan *username* yang sudah ada dalam *database*, sehingga sistem meminta pengguna untuk *login* dari pada membuat akun baru. Notifikasi-notifikasi ini berperan penting dalam memastikan



pengguna memahami status dan hasil dari tindakan mereka selama proses autentikasi, dengan memberikan petunjuk yang jelas dan informatif.

4.1.2.3. Hasil Pengujian Pemantauan Satwa

Terdapat kondisi menampilkan status keberadaan satwa, yaitu saat terdapat satwa yang terperangkap di dalam perangkap dan saat tidak ada satwa yang terperangkap. Hasil pengujian ditampilkan pada Gambar 4.6.



Gambar 4. 6 Tampilan Pemantauan Aplikasi (a) Kondisi Tidak Terdapat Satwa (b) Kondisi Terdapat Satwa

Pada Gambar 4.7 menunjukkan hasil pemantauan satwa liar yang berasal dari Firebase *real-time*.



Gambar 4. 7 Gambar *Real-time Database* Pemantauan (a) Status “0” (b) Status “1”

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



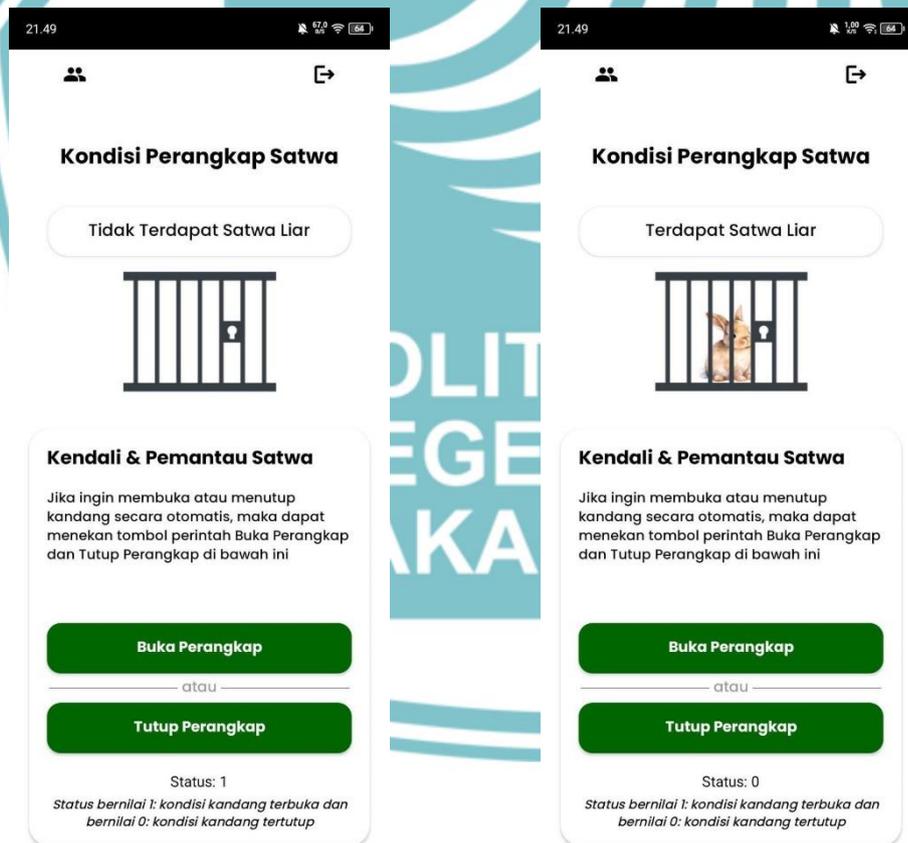
**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Gambar 4.7 menunjukkan kondisi dari hasil pemantauan satwa yang dideteksi oleh sensor ultrasonik. Setiap perubahan jika terdapat objek yang terdeteksi data akan masuk ke Firebase *real-time database*. Hasil pemantauan menunjukkan bahwa data gambar yang diterima oleh aplikasi Perangkat Satwa sesuai dengan data pada Firebase *real-time database*.

#### 4.1.2.4. Hasil Pengujian Kendali Perangkat

Untuk kendali jarak jauh perangkat dengan membuka ataupun menutup perangkat secara otomatis, yaitu jika saat perangkat ingin dibuka, maka tombol ‘Buka Perangkat’ digunakan begitupun jika saat perangkat ingin ditutup, maka tombol ‘Tutup Perangkat’ dapat digunakan. Pada Gambar 4.8 menunjukkan hasil kendali satwa yang dapat dikendalikan menggunakan tombol perintah buka ataupun tutup



(a)

(b)

Gambar 4. 8 Kendali Perangkat Satwa (a) Kondisi Perangkat dibuka (b) Kondisi Perangkat ditutup

Pada Gambar 4.9 menunjukkan hasil kendali perangkat satwa liar yang ada pada Firebase *real-time database*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4. 9 Kendali Perangkap Satwa (a) Status Perangkap “0” (b) Status Perangkap (1)

#### 4.1.3. Analisa Data

Dari hasil pengujian pada aplikasi Android, dapat dianalisis bahwa proses yang telah dilakukan adalah dengan mengunduh aplikasi 'Perangkap Satwa' pada smartphone. Aplikasi ini menampilkan kondisi keberadaan satwa yang terdeteksi oleh sensor ultrasonik yang terhubung dengan ESP32. Alat tersebut tersambung ke Firebase *database*. Pengguna juga dapat mengirimkan perintah untuk membuka dan menutup perangkap, di mana perintah tersebut akan diubah di Firebase dan dapat mengendalikan alat. Servo kemudian akan bergerak untuk membuka dan menutup perangkap berdasarkan perintah tersebut. Data yang diterima dan dikirim dari Firebase ke aplikasi diperoleh dari proses pengiriman dan penerimaan data sensor ultrasonik yang dikirimkan menggunakan ESP32 dan LoRa.

#### 4.2 Pengujian *Quality of Service*

Pengujian ini bertujuan untuk mengetahui kecepatan jaringan internet agar dapat memastikan konektivitas yang baik antara mikrokontroler dan aplikasi Android. Mikrokontroler tersebut bertugas mengirimkan data berupa status ke *database*, sedangkan aplikasi Android mengirimkan perintah ke *database*, yang kemudian dapat digunakan untuk mengendalikan servo pada perangkat. Parameter yang diukur dalam pengujian ini meliputi *throughput*, *packet loss*, dan *delay*.

##### 4.2.1. Deskripsi Pengujian

Pengujian ini menganalisis kinerja jaringan internet operator Three dengan mengukur *Quality of Service* (QoS) menggunakan aplikasi Wireshark pada laptop yang terhubung ke *hotspot* seluler. Wireshark digunakan untuk memantau dan menganalisis data lalu lintas jaringan, seperti *throughput*, *packet loss*, *latency*, dan *jitter*. Pengukuran dilakukan pada tiga waktu berbeda yaitu pagi, siang, dan malam



**Hak Cipta milik Politeknik Negeri Jakarta**

**Hak Cipta :**

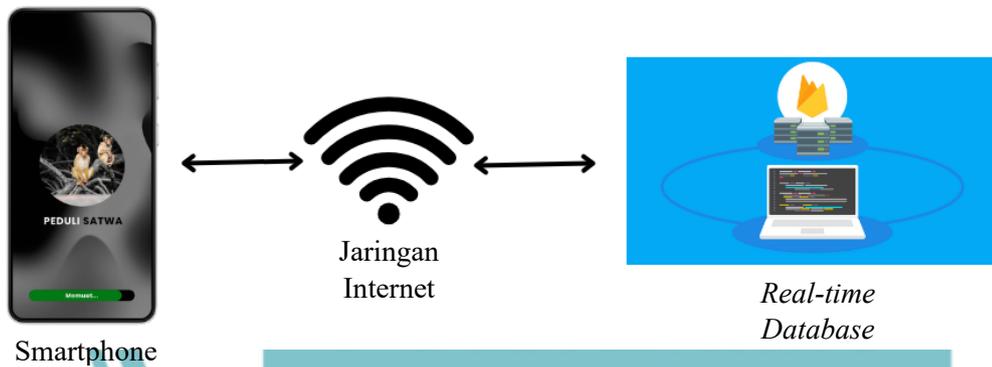
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

hari untuk mengetahui perbedaan performa jaringan berdasarkan waktu dan juga mengukur selama tiga hari untuk mengetahui perbandingannya. Hasil analisis ini membantu mengevaluasi kualitas layanan internet operator Three dan mengidentifikasi potensi penurunan kinerja pada waktu-waktu tertentu. Terdapat beberapa perangkat yang digunakan dalam melakukan pengambilan data *Quality of Service*, yaitu laptop, aplikasi Wireshark, dan *real-time database* Firebase.

**4.2.2. Prosedur Pengujian**

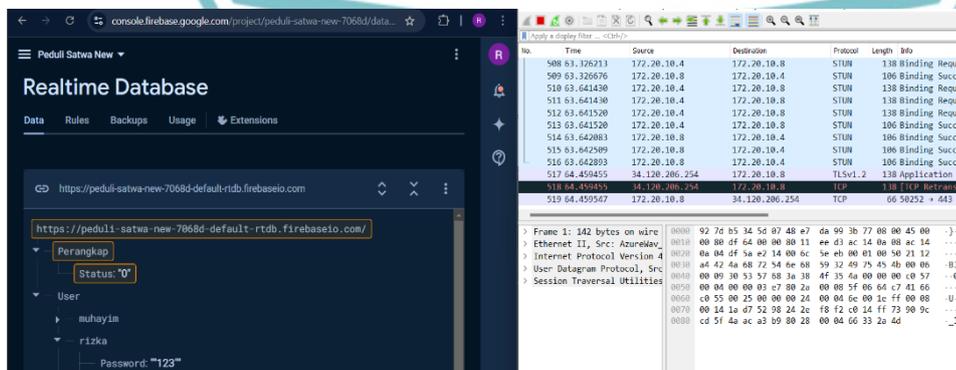
Terdapat beberapa tahap untuk melakukan pengambilan data *Quality of Service* pada aplikasi Wireshark:

1. Menyiapkan *smartphone* yang sudah terhubung ke jaringan internet untuk membuka aplikasi Peduli Satwa yang terhubung dengan *real-time database* Firebase.



Gambar 4. 10 *Set Up* Pengujian Aplikasi Android

2. Membuka *real-time database* pada Firebase yang sudah terdapat data status dan juga kendali perangkat satwa.
3. Menyiapkan laptop yang sudah terhubung ke jaringan internet untuk membuka *website* Firebase dan juga aplikasi Wireshark seperti pada Gambar 4.11.



Gambar 4. 11 *Website* Firebase dan Aplikasi Wireshark



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4. Membuka *real-time database* di Firebase untuk melihat perubahan status perangkat satwa secara *real-time*.
5. Melakukan pemantauan dan kendali secara *real-time* pada aplikasi Peduli Satwa.
6. Melakukan sinkronasi data perangkat satwa pada *real-time database* dan aplikasi Android.
7. Lakukan pengukuran selama 2 menit. Setelah itu, untuk melihat hasil pengukuran dapat dilihat pada *Statistic > Capture File Properties* seperti pada Gambar 4.12.



Gambar 4. 12 *Statistic > Capture File Properties*

8. Menampilkan status perangkat satwa dan kendali perangkat, dimana jika perangkat terbuka status bernilai 0 yang menandakan tidak terdapat satwa di dalam perangkat dan kondisi perangkat dalam keadaan terbuka dan jika perangkat tertutup status bernilai 1 yang berarti terdapat satwa yang terperangkap dan juga perangkat dalam kondisi tertutup.
9. *Statistics > Capture file properties*  
Parameter hasil pengukuran dapat dilihat dalam statistic pada Gambar 4.13.

Measurement	Captured	Displayed	Marked
Packets	1608	1608 (100.0%)	—
Time span, s	96.074	96.074	—
Average pps	16.7	16.7	—
Average packet size, B	256	256	—
Bytes	411835	411835 (100.0%)	0
Average bytes/s	4286	4286	—
Average bits/s	34 k	34 k	—

Gambar 4. 13 *Capture File Properties*

#### 4.2.3. Hasil Pengujian *Quality of Service*

Data hasil pengujian diperoleh dari proses pengambilan data yang dilakukan pada tiga waktu berbeda, yaitu pagi, siang, dan malam hari. Pengambilan data pada berbagai waktu ini bertujuan untuk mengetahui perbedaan kecepatan jaringan pada setiap periode waktu tersebut. Untuk memastikan bahwa data yang dihasilkan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

berasal dari Firebase *database* dan sesuai dengan yang diharapkan, dilakukan proses penyaringan data dengan menggunakan protokol HTTP dan TCP. Penyaringan ini berguna untuk memisahkan dan mengidentifikasi lalu lintas data yang relevan, sehingga hanya data yang valid dan berasal dari komunikasi dengan Firebase *database* yang akan dianalisis lebih lanjut. Hal ini penting untuk menjaga akurasi dan keandalan hasil pengukuran, serta untuk memastikan bahwa analisis kualitas jaringan benar-benar mencerminkan performa jaringan pada berbagai waktu yang diujikan. Berikut hasil pengujian kualitas jaringan pada waktu yang berbeda.

#### 4.2.3.1. Pengukuran *Quality of Service* Hari Pertama

##### 1. Pengukuran *Quality of Service* pagi hari pada Pukul 06.04 WIB

Pengukuran jaringan internet dilakukan dengan melibatkan Firebase *database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari Firebase, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.14 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
32931	50.360847	172.20.10.8	34.104.35.123	HTTP	448	HEAD /edgedl/diffgen-puff
32933	50.410451	34.104.35.123	172.20.10.8	HTTP	604	HTTP/1.1 200 OK
32935	50.483874	172.20.10.8	34.104.35.123	HTTP	520	GET /edgedl/diffgen-puff
32938	50.532630	34.104.35.123	172.20.10.8	HTTP	371	HTTP/1.1 206 Partial Con
33007	54.684637	172.20.10.8	34.104.35.123	HTTP	523	GET /edgedl/diffgen-puff:

Gambar 4. 14 HTTP Wireshark Pengukuran Jaringan Pagi Hari Pertama

Gambar 4.14 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `172.20.10.8` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan Firebase *database*. Permintaan `GET` ini digunakan untuk mengambil data dari Firebase, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.15 hasil pengujian di pagi hari.


**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

No.	Time	Source	Destination	Protocol	Length	Info
1589	0.247955	172.20.10.8	142.251.12.102	TCP	54	52076 → 443 [ACK] Seq=1 Ack=1
1601	0.249032	172.20.10.8	142.251.12.102	TCP	1454	52076 → 443 [ACK] Seq=1 Ack=1
1602	0.249032	172.20.10.8	142.251.12.102	TLSv1.3	443	[Client Hello]
2100	0.324517	74.125.68.156	172.20.10.8	TCP	66	443 → 52077 [SYN, ACK] Seq=0
2102	0.324615	172.20.10.8	74.125.68.156	TCP	54	52077 → 443 [ACK] Seq=1 Ack=1
2107	0.325603	172.20.10.8	74.125.68.156	TCP	1454	52077 → 443 [ACK] Seq=1 Ack=1

Gambar 4. 15 TCP Wireshark Pengukuran Jaringan Pagi Hari Pertama

Gambar 4.15 menunjukkan perangkat dengan IP `172.20.10.8` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `142.251.12.102`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar 4.16 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Measurement	Captured	Displayed
Packets	36637	886 (2.4%)
Time span, s	210.840	210.683
Average pps	173.8	4.2
Average packet size, B	1061	423
Bytes	38866449	374358 (1.0%)
Average bytes/s	184 k	1776
Average bits/s	1474 k	14 k

Gambar 4. 16 Capture File Properties Pengukuran Jaringan Pagi Hari Pertama

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.1 yang merupakan parameter dari hasil pengukuran.

Tabel 4.1 Parameter Hasil Pengukuran Pagi Hari Pertama

Measurement	Captured	Displayed
<i>Packets</i>	36637	36637 (100.0%)
<i>Time span</i>	210.840	210.840
<i>Bytes</i>	38866449	38866449 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{a. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{38866449 \text{ Bytes}}{210.840 \text{ s}}
 \end{aligned}$$

$$= 184.241 \text{ KB/s} \times 8 = 1.474 \text{ Kbps}$$

$$b. \text{ Packet Loss} = \frac{\text{Paket dikirim} - \text{Paket diterima}}{\text{Paket diterima}} \times 100\%$$

$$= \frac{36637 - 36637}{36637} \times 100\% = 0\%$$

$$c. \text{ Delay} = \frac{\text{Waktu pengiriman data}}{\text{Paket diterima}}$$

$$= \frac{210.840}{36637} = 5.75 \text{ ms}$$

## 2. Pengukuran *Quality of Service* siang hari Pukul 15.50

Pengukuran jaringan internet dilakukan dengan melibatkan Firebase *database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari Firebase, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.17 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
3426	16.414137	192.168.54.85	23.43.248.27	HTTP	165	GET /connecttest.txt HTTP/1.1
3428	16.444551	23.43.248.27	192.168.54.85	HTTP	241	HTTP/1.1 200 OK (text/plain)
3854	51.962604	192.168.54.85	23.43.248.26	HTTP	165	GET /connecttest.txt HTTP/1.1
3858	51.996917	23.43.248.26	192.168.54.85	HTTP	241	HTTP/1.1 200 OK (text/plain)

Gambar 4. 17 HTTP Wireshark Pengukuran Jaringan Siang Hari Pertama

Gambar 4.17 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `192.168.54.85` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan Firebase *database*. Permintaan `GET` ini digunakan untuk mengambil data dari Firebase, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.18 hasil pengujian di pagi hari.

No.	Time	Source	Destination	Protocol	Length	Info
26	1.407862	192.168.54.85	172.67.9.68	TCP	54	61151 → 443 [ACK] Seq=1 Ack=1
27	1.409480	192.168.54.85	172.67.9.68	TLSv1.3	583	Client Hello
33	1.445105	172.67.9.68	192.168.54.85	TCP	54	443 → 61151 [ACK] Seq=1 Ack=53
34	1.445105	172.67.9.68	192.168.54.85	TLSv1.3	273	Server Hello, Change Cipher Sp

Gambar 4. 18 TCP Wireshark Pengukuran Jaringan Siang Hari Pertama



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Gambar 4.18 menunjukkan perangkat dengan IP `192.168.54.85` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `172.67.9.68`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar 4.19 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Statistics	Captured	Displayed	Marked
Measurement	4722	1343 (28.4%)	—
Packets	151.651	150.288	—
Time span, s	31.1	8.9	—
Average pps	683	309	—
Average packet size, B	3224924	415376 (12.9%)	0
Bytes	21 k	2763	—
Average bytes/s	170 k	22 k	—
Average bits/s			

Gambar 4. 19 *Capture File Properties* Pengukuran Jaringan Siang Hari Pertama

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.2 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 2 Parameter Hasil Pengukuran Siang Hari Pertama

Measurement	Captured	Displayed
<i>Packets</i>	4722	4722 (100.0%)
<i>Time span</i>	151.651	151.651
<i>Bytes</i>	3224924	3224924 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{a. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{3224924 \text{ Bytes}}{151.651 \text{ s}} \\
 &= 21.265 \text{ KB/s} \times 8 = 170 \text{ Kbps}
 \end{aligned}$$

$$\begin{aligned}
 \text{b. } \textit{Packet Loss} &= \frac{\textit{Paket dikirim} - \textit{Paket diterima}}{\textit{Paket diterima}} \times 100\% \\
 &= \frac{4722 - 4722}{4722} \times 100\% = 0\%
 \end{aligned}$$

$$\begin{aligned}
 \text{c. } \textit{Delay} &= \frac{\textit{Waktu pengiriman data}}{\textit{Paket diterima}} \\
 &= \frac{151.651}{4722} = 32.11 \text{ ms}
 \end{aligned}$$



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 3. Pengukuran *Quality of Service* malam hari pukul 21.35

Pengukuran jaringan internet dilakukan dengan melibatkan Firebase *database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari Firebase, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.20 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
1692	25.442227	172.20.10.2	114.10.52.216	HTTP	208	GET /connecttest.txt HTTP/1.1
1696	25.754707	114.10.52.216	172.20.10.2	HTTP	241	HTTP/1.1 200 OK (text/plain)
2437	59.801863	172.20.10.2	114.10.52.179	HTTP	208	GET /connecttest.txt HTTP/1.1
2442	59.853451	114.10.52.179	172.20.10.2	HTTP	241	HTTP/1.1 200 OK (text/plain)

Gambar 4. 20 HTTP Wireshark Pengukuran Jaringan Malam Hari Pertama

Gambar 4.20 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `172.20.10.2` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan Firebase *database*. Permintaan `GET` ini digunakan untuk mengambil data dari Firebase, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.21 hasil pengujian di malam hari.

No.	Time	Source	Destination	Protocol	Length	Info
62	2.371132	104.22.0.235	172.20.10.2	TCP	66	443 → 51797 [SYN, ACK]
66	2.371300	172.20.10.2	104.22.0.235	TCP	54	51797 → 443 [ACK] Seq=...
95	2.373341	172.20.10.2	104.22.0.235	TLSv1.3	583	Client Hello
107	2.460595	172.20.10.2	172.253.118.95	TCP	66	51797 → 443 [SYN] Seq=...

Gambar 4. 21 TCP Wireshark Pengukuran Jaringan Malam Hari Pertama

Gambar 4.21 menunjukkan perangkat dengan IP `172.20.10.2` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `104.22.0.235`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

performa jaringan. Pada Gambar 4.22 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Interface	Dropped packets	Capture filter	Link type
Wi-Fi	0 (0.0%)	none	Ethernet
<b>Statistics</b>			
<b>Measurement</b>	<b>Captured</b>	<b>Displayed</b>	
Packets	3481	3481 (100.0%)	
Time span, s	240.622	240.622	
Average pps	14.5	14.5	
Average packet size, B	435	435	
Bytes	1512605	1512605 (100.0%)	
Average bytes/s	6286	6286	
Average bits/s	50 k	50 k	

Gambar 4. 22 *Capture File Properties* Pengukuran Jaringan Malam Hari Pertama

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.3 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 3 Parameter Hasil Pengukuran Malam Hari Pertama

<i>Measurement</i>	<i>Captured</i>	<i>Displayed</i>
<i>Packets</i>	3481	3481 (100.0%)
<i>Time span</i>	240.622	240.622
<i>Bytes</i>	1512605	1512605 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{a. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{1512605 \text{ Bytes}}{240.622 \text{ s}} \\
 &= 6.286 \text{ KB/s} \times 8 = 50 \text{ Kbps} \\
 \text{b. } \textit{Packet Loss} &= \frac{\textit{Paket dikirim} - \textit{Paket diterima}}{\textit{Paket diterima}} \times 100\% \\
 &= \frac{3481 - 3481}{3481} \times 100\% = 0\% \\
 \text{c. } \textit{Delay} &= \frac{\textit{Waktu pengiriman data}}{\textit{Paket diterima}} \\
 &= \frac{240.622}{3481} = 69.12 \text{ ms}
 \end{aligned}$$

#### 4.2.3.2. Pengukuran *Quality of Service* Hari Kedua

1. Pengukuran *Quality of Service* Pagi Hari Pukul 05.04 WIB

Pengukuran jaringan internet dilakukan dengan melibatkan Firebase *database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari Firebase, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.23



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
418	12.324237	172.20.10.8	74.125.130.94	HTTP	254	GET /r/r1.cr1 HTTP/1.1
420	12.373185	74.125.130.94	172.20.10.8	HTTP	277	HTTP/1.1 304 Not Modified
428	12.467772	172.20.10.8	23.202.109.56	HTTP	281	GET / HTTP/1.1
431	12.516512	23.202.109.56	172.20.10.8	HTTP	317	HTTP/1.1 304 Not Modified

Gambar 4. 23 HTTP Wireshark Pengukuran Jaringan Pagi Hari kedua

Gambar 4.23 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `172.20.10.8` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan Firebase *Database*. Permintaan `GET` ini digunakan untuk mengambil data dari Firebase, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.24 hasil pengujian di pagi hari.

No.	Time	Source	Destination	Protocol	Length	Info
4	0.282967	172.20.10.8	172.67.9.68	TCP	66	51531 → 443 [SYN] Seq=0 Win=64240
5	0.639335	172.67.9.68	172.20.10.8	TCP	66	443 → 51531 [SYN, ACK] Seq=0 Ack=
6	0.639434	172.20.10.8	172.67.9.68	TCP	54	51531 → 443 [ACK] Seq=1 Ack=1 Win
7	0.640531	172.20.10.8	172.67.9.68	TLSv1.3	583	Client Hello

Gambar 4. 24 TCP Wireshark Pengukuran Jaringan Pagi Hari kedua

menunjukkan perangkat dengan IP `172.20.10.8` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `172.67.9.68`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.



Measurement	Captured	Displayed
Packets	3785	3785 (100.0%)
Time span, s	56.058	56.058
Average pps	67.5	67.5
Average packet size, B	374	374
Bytes	1413805	1413805 (100.0%)
Average bytes/s	25 k	25 k
Average bits/s	201 k	201 k

Gambar 4. 25 *Capture File Properties* Pengukuran Jaringan Pagi Hari kedua

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.4 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 4 Parameter Hasil Pengukuran Pagi Hari kedua

Measurement	Captured	Displayed
<i>Packets</i>	3785	3785 (100.0%)
<i>Time span</i>	56.058	56.058
<i>Bytes</i>	1413805	1413805 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{b. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{1413805 \text{ Bytes}}{56.058 \text{ s}} \\
 &= 25.22 \text{ KB/s} \times 8 = 201.76 \text{ Kbps} \\
 \text{c. } \textit{Packet Loss} &= \frac{\text{Paket dikirim} - \text{Paket diterima}}{\text{Paket diterima}} \times 100\% \\
 &= \frac{3785 - 3785}{3785} \times 100\% = 0\% \\
 \text{d. } \textit{Delay} &= \frac{\text{Waktu pengiriman data}}{\text{Paket diterima}} \\
 &= \frac{56.058}{3785} = 14.81 \text{ ms}
 \end{aligned}$$

## 2. Pengukuran *Quality of Service* Siang Hari Pukul 14.34 WIB

Pengukuran jaringan internet dilakukan dengan melibatkan *Firestore database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari *Firestore*, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.26 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

No.	Time	Source	Destination	Protocol	Length	Info
3101	12.010890	192.168.233.84	34.104.35.123	HTTP	448	HEAD /edged1/diffgen-puffin/kiabhabjdbkjd
3107	12.045559	34.104.35.123	192.168.233.84	HTTP	603	HTTP/1.1 200 OK
3108	12.080150	192.168.233.84	34.104.35.123	HTTP	520	GET /edged1/diffgen-puffin/kiabhabjdbkjdpi
3123	12.132471	34.104.35.123	192.168.233.84	HTTP	370	HTTP/1.1 206 Partial Content
3222	13.964642	192.168.233.84	152.195.38.76	HTTP	260	GET /pca3.cr1 HTTP/1.1

Gambar 4. 26 HTTP Wireshark Pengukuran Jaringan Siang Hari kedua

Gambar 4.26 menunjukkan adanya permintaan 'GET' dari perangkat dengan IP `192.168.233.84` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan Firebase *database*. Permintaan 'GET' ini digunakan untuk mengambil data dari Firebase, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.27 hasil pengujian di pagi hari.

No.	Time	Source	Destination	Protocol	Length	Info
6	0.595490	192.168.233.84	104.22.1.235	TCP	66	61261 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK
9	0.815247	104.22.1.235	192.168.233.84	TCP	66	443 → 61261 [SYN, ACK] Seq=0 Ack=1 Win=22400 Len=0 MSS=1400
10	0.815367	192.168.233.84	104.22.1.235	TCP	54	61261 → 443 [ACK] Seq=1 Ack=1 Win=65792 Len=0
11	0.816859	192.168.233.84	104.22.1.235	TLSv1.3	583	Client Hello
13	1.272982	104.22.1.235	192.168.233.84	TCP	54	443 → 61261 [ACK] Seq=1 Ack=530 Win=22912 Len=0
14	1.272982	104.22.1.235	192.168.233.84	TLSv1.3	273	Server Hello, Change Cipher Spec, Application Data

Gambar 4. 27 TCP Wireshark Pengukuran Jaringan Siang Hari kedua

Gambar 4.27 menunjukkan perangkat dengan IP `192.168.233.84` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `104.22.1.235`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase *database* melalui aplikasi Android. Proses 'Client Hello' menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar 4.28 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Statistics			
Measurement	Captured	Displayed	Marked
Packets	4722	4722 (100.0%)	—
Time span, s	151.651	151.651	—
Average pps	31.1	31.1	—
Average packet size, B	683	683	—
Bytes	3224924	3224924 (100.0%)	0
Average bytes/s	21 k	21 k	—
Average bits/s	170 k	170 k	—

Gambar 4. 28 Capture File Properties Pengukuran Jaringan Siang Hari kedua



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.5 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 5 Parameter Hasil Pengukuran Siang Hari kedua

Measurement	Captured	Displayed
<i>Packets</i>	4722	4722 (100.0%)
<i>Time span</i>	151.651	151.651
<i>Bytes</i>	3224924	3224924 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{b. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{3224924 \text{ Bytes}}{151.651 \text{ s}} \\
 &= 21.265 \text{ KB/s} \times 8 = 170 \text{ Kbps} \\
 \text{c. } \textit{Packet Loss} &= \frac{\textit{Paket dikirim} - \textit{Paket diterima}}{\textit{Paket diterima}} \times 100\% \\
 &= \frac{4722 - 4722}{4722} \times 100\% = 0\% \\
 \text{d. } \textit{Delay} &= \frac{\textit{Waktu pengiriman data}}{\textit{Paket diterima}} \\
 &= \frac{151.651}{4722} = 32.11 \text{ ms}
 \end{aligned}$$

3. Pengukuran *Quality of Service* malam hari pukul 21.06

Pengukuran jaringan internet dilakukan dengan melibatkan *Firestore database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari *Firestore*, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.29 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
92625	45.139282	172.20.10.8	34.104.35.123	HTTP	448	HEAD /edgedl/diffgen-puffin/hfnkpinl
92636	45.510683	34.104.35.123	172.20.10.8	HTTP	644	HTTP/1.1 200 OK
92648	45.545237	172.20.10.8	34.104.35.123	HTTP	520	GET /edgedl/diffgen-puffin/hfnkpinl
92665	45.763039	34.104.35.123	172.20.10.8	HTTP	411	HTTP/1.1 206 Partial Content
92723	49.755232	172.20.10.8	34.104.35.123	HTTP	523	GET /edgedl/diffgen-puffin/hfnkpinl

Gambar 4. 29 HTTP Wireshark Pengukuran Jaringan Malam Hari kedua

Gambar 4.29 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `172.20.10.8` ke server eksternal melalui protokol HTTP, sebagai bagian dari



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

pengujian kualitas jaringan dalam konteks komunikasi dengan Firebase *database*. Permintaan `GET` ini digunakan untuk mengambil data dari Firebase, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.30 hasil pengujian di malam hari.

No.	Time	Source	Destination	Protocol	Length	Info
7911	2.491475	172.20.10.8	104.22.1.235	TCP	54	62690 → 443 [ACK] Seq=1 Ack
7922	2.492898	172.20.10.8	104.22.1.235	TLSv1.3	583	Client Hello
8371	2.572045	172.20.10.8	142.251.221.116	TCP	66	62691 → 443 [SYN] Seq=0 Win
8669	2.639532	104.22.1.235	172.20.10.8	TCP	54	443 → 62690 [ACK] Seq=1 Ack

Gambar 4. 30 TCP Wireshark Pengukuran Jaringan Malam Hari kedua

Gambar 4.30 menunjukkan perangkat dengan IP `172.20.10.8` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `104.22.1.235`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar 4.31 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Measurement	Captured	Displayed
Packets	93153	1055 (1.1%)
Time span, s	63.817	61.545
Average pps	1459.7	17.1
Average packet size, B	1137	463
Bytes	105943162	488982 (0.5%)
Average bytes/s	1660 k	7945
Average bits/s	13 M	63 k

Gambar 4. 31 *Capture File Properties* Pengukuran Jaringan Malam Hari kedua

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.6 yang merupakan parameter dari hasil pengukuran.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Tabel 4. 6 Parameter Hasil Pengukuran Malam Hari kedua

<i>Measurement</i>	<i>Captured</i>	<i>Displayed</i>
<i>Packets</i>	93153	93153 (100.0%)
<i>Time span</i>	63.817	63.817
<i>Bytes</i>	105943165	1512605 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{b. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{105943165 \text{ Bytes}}{63.817 \text{ s}} \\
 &= 1.660 \text{ KB/s} \times 8 = 13 \text{ Kbps} \\
 \text{c. } \textit{Packet Loss} &= \frac{\text{Paket dikirim} - \text{Paket diterima}}{\text{Paket diterima}} \times 100\% \\
 &= \frac{93153 - 93153}{93153} \times 100\% = 0\% \\
 \text{d. } \textit{Delay} &= \frac{\text{Waktu pengiriman data}}{\text{Paket diterima}} \\
 &= \frac{63.817}{93153} = 0.685 \text{ ms}
 \end{aligned}$$

#### 4.2.3.3. Pengukuran *Quality of Service* Hari ketiga

##### 1. Pengukuran *Quality of Service* Pagi Hari Pukul 5.10 WIB

Pengukuran jaringan internet dilakukan dengan melibatkan *Firestore database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari *Firestore*, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.32 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
42580	163.579916	172.20.10.8	114.10.52.179	HTTP	178	GET /ncsi.txt HTTP/1.1
42583	163.655472	114.10.52.179	172.20.10.8	HTTP	233	HTTP/1.1 200 OK (text/plain)

Gambar 4. 32 HTTP Wireshark Pengukuran Jaringan Pagi Hari ketiga

Gambar 4.32 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `172.20.10.8` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan *Firestore Database*. Permintaan `GET` ini digunakan untuk mengambil data dari *Firestore*, yang



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan Firebase, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.33 hasil pengujian di pagi hari.

No.	Time	Source	Destination	Protocol	Length	Info
23	1.211821	172.67.9.68	172.20.10.8	TCP	66	443 → 52556 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=
24	1.211853	172.20.10.8	172.67.9.68	TCP	54	52556 → 443 [ACK] Seq=1 Ack=1 Win=65535 Len=0
25	1.213799	172.20.10.8	172.67.9.68	TLSv1.3	583	Client Hello
26	1.262780	172.67.9.68	172.20.10.8	TCP	54	443 → 52556 [ACK] Seq=1 Ack=530 Win=73728 Len=0

Gambar 4. 33 TCP Wireshark Pengukuran Jaringan Pagi Hari ketiga

menunjukkan perangkat dengan IP `172.20.10.8` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `172.67.9.68`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Measurement	Captured	Displayed
Packets	42661	332 (0.8%)
Time span, s	173.955	173.656
Average pps	245.2	1.9
Average packet size, B	1107	294
Bytes	47225674	97744 (0.2%)
Average bytes/s	271 k	562
Average bits/s	2171 k	4502

Gambar 4. 34 Capture File Properties Pengukuran Jaringan Pagi Hari ketiga

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.7 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 7 Parameter Hasil Pengukuran Pagi Hari ketiga

Measurement	Captured	Displayed
<i>Packets</i>	42611	42611 (100.0%)
<i>Time span</i>	173.955	173.955
<i>Bytes</i>	47225674	47225674 (100.0%)



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{c. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{47225674 \text{ Bytes}}{173.955 \text{ s}} \\
 &= 25.22 \text{ KB/s} \times 8 = 201.76 \text{ Kbps}
 \end{aligned}$$

$$\begin{aligned}
 \text{d. } \textit{Packet Loss} &= \frac{\textit{Paket dikirim} - \textit{Paket diterima}}{\textit{Paket diterima}} \times 100\% \\
 &= \frac{42611 - 42611}{42611} \times 100\% = 0\%
 \end{aligned}$$

$$\begin{aligned}
 \text{e. } \textit{Delay} &= \frac{\textit{Waktu pengiriman data}}{\textit{Paket diterima}} \\
 &= \frac{173.955}{42611} = 4.08 \text{ ms}
 \end{aligned}$$

2. Pengukuran *Quality of Service* Siang Hari Pukul 14.15 WIB

Pengukuran jaringan internet dilakukan dengan melibatkan *Firestore database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari *Firestore*, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.35 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
46410	80.480690	172.20.10.8	114.10.52.179	HTTP	165	GET /connecttest.txt HTTP/1.1
46431	80.530960	114.10.52.179	172.20.10.8	HTTP	241	HTTP/1.1 200 OK (text/plain)

Gambar 4. 35 HTTP Wireshark Pengukuran Jaringan Siang Hari

Gambar 4.35 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `114.10.52.8` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan *Firestore database*. Permintaan `GET` ini digunakan untuk mengambil data dari *Firestore*, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan *Firestore*, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.36 hasil pengujian di pagi hari.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

No.	Time	Source	Destination	Protocol	Length	Info
13	0.996831	104.22.1.235	172.20.10.8	TCP	66	443 → 54095 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0
14	0.996940	172.20.10.8	104.22.1.235	TCP	54	54095 → 443 [ACK] Seq=1 Ack=1 Win=65792 Len=0
15	0.998739	172.20.10.8	104.22.1.235	TLSv1.3	583	Client Hello

Gambar 4. 36 TCP Wireshark Pengukuran Jaringan Siang Hari ketiga

Gambar 4.36 menunjukkan perangkat dengan IP `172.20.10.8` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `104.22.1.235`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase *database* melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar 4.37 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Measurement	Captured	Displayed
Packets	46724	326 (0.7%)
Time span, s	87.688	86.362
Average pps	532.8	3.8
Average packet size, B	1033	183
Bytes	48279990	59690 (0.1%)
Average bytes/s	550 k	691
Average bits/s	4404 k	5529

Gambar 4. 37 *Capture File Properties* Pengukuran Jaringan Siang Hari

Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.8 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 8 Parameter Hasil Pengukuran Siang Hari ketiga

Measurement	Captured	Displayed
<i>Packets</i>	46724	46724 (100.0%)
<i>Time span</i>	87.688	87.688
<i>Bytes</i>	48279990	48279990 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{c. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{48279990 \text{ Bytes}}{87.688 \text{ s}} \\
 &= 550.588 \text{ KB/s} \times 8 = 4.40 \text{ Kbps}
 \end{aligned}$$



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

$$\begin{aligned}
 \text{d. } \textit{Packet Loss} &= \frac{\text{Paket dikirim} - \text{Paket diterima}}{\text{Paket diterima}} \times 100\% \\
 &= \frac{46724 - 46724}{46724} \times 100\% = 0\% \\
 \text{e. } \textit{Delay} &= \frac{\text{Waktu pengiriman data}}{\text{Paket diterima}} \\
 &= \frac{87.688}{46724} = 1.876 \text{ ms}
 \end{aligned}$$

3. Pengukuran *Quality of Service* Malam Hari Pukul 21.35 WIB

Pengukuran jaringan internet dilakukan dengan melibatkan *Firestore database* yang terhubung ke aplikasi Android. Untuk memastikan data berasal dari *Firestore*, dilakukan penyaringan menggunakan protokol HTTP. Gambar 4.38 menunjukkan hasil penyaringan ini, yang bertujuan untuk memverifikasi dan memastikan hanya data yang relevan dan valid digunakan dalam analisis, sehingga hasil pengukuran lebih akurat dan dapat diandalkan.

No.	Time	Source	Destination	Protocol	Length	Info
92	1.477077	172.20.10.8	114.10.52.216	HTTP	178	GET /ncsi.txt HTTP/1.1
100	1.639371	114.10.52.216	172.20.10.8	HTTP	233	HTTP/1.1 200 OK (text/plain)
45859	22.111428	172.20.10.8	114.10.52.216	HTTP	165	GET /connecttest.txt HTTP/1.1
45864	22.266285	114.10.52.216	172.20.10.8	HTTP	241	HTTP/1.1 200 OK (text/plain)

Gambar 4. 38 Gambar 4. 39 HTTP Wireshark Pengukuran Jaringan Malam Hari ketiga

Gambar 4.38 menunjukkan adanya permintaan `GET` dari perangkat dengan IP `172.20.10.8` ke server eksternal melalui protokol HTTP, sebagai bagian dari pengujian kualitas jaringan dalam konteks komunikasi dengan *Firestore database*. Permintaan `GET` ini digunakan untuk mengambil data dari *Firestore*, yang dipantau menggunakan Wireshark untuk memastikan validitas data. Penyaringan ini penting untuk memastikan bahwa data yang dianalisis benar-benar terkait dengan *Firestore*, sehingga hasil pengukuran jaringan akurat. Selain itu, protokol TCP juga dianalisis menggunakan penyaringan data di Wireshark, seperti yang ditunjukkan pada Gambar 4.40 hasil pengujian di malam hari.

No.	Time	Source	Destination	Protocol	Length	Info
26	0.706871	104.22.0.235	172.20.10.8	TCP	66	443 → 50389 [SYN, ACK] Seq=
27	0.706991	172.20.10.8	104.22.0.235	TCP	54	50389 → 443 [ACK] Seq=1 Ack
28	0.708397	172.20.10.8	104.22.0.235	TLSv1.3	583	Client Hello
32	0.812781	104.22.0.235	172.20.10.8	TCP	54	443 → 50389 [ACK] Seq=1 Ack
33	0.812781	104.22.0.235	172.20.10.8	TLSv1.3	273	Server Hello, Change Cipher

Gambar 4. 40 TCP Wireshark Pengukuran Jaringan Malam Hari



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Gambar 4.40 menunjukkan perangkat dengan IP `172.20.10.8` memulai koneksi aman menggunakan protokol TLS 1.3 dengan server ber-IP `104.22.1.235`. Ini adalah bagian dari proses pengujian di mana perangkat terhubung dengan Firebase database melalui aplikasi Android. Proses `Client Hello` menandakan awal koneksi yang terenkripsi untuk memastikan keamanan data. Penggunaan Wireshark membantu memverifikasi bahwa perangkat berhasil membangun koneksi aman dengan Firebase, memastikan integritas dan keamanan data selama pengujian performa jaringan. Pada Gambar 3.41 merupakan hasil dari pengukuran jaringan yang ditampilkan pada *capture file properties* pada Wireshark.

Measurement	Captured	Displayed
Packets	47174	758 (1.6%)
Time span, s	78.236	77.268
Average pps	603.0	9.8
Average packet size, B	1096	481
Bytes	51699302	364374 (0.7%)
Average bytes/s	660 k	4715
Average bits/s	5286 k	37 k

Gambar 4. 41 *Capture File Properties* Pengukuran Jaringan Malam Hari ketiga  
 Dari hasil *capture* yang telah ditunjukkan didapatkan hasil yang ditunjukkan pada Tabel 4.9 yang merupakan parameter dari hasil pengukuran.

Tabel 4. 9 Parameter Hasil Pengukuran Malam Hari ketiga

<i>Measurement</i>	<i>Captured</i>	<i>Displayed</i>
<i>Packets</i>	47174	47174 (100.0%)
<i>Time span</i>	78.236	78.236
<i>Bytes</i>	51699302	51699302 (100.0%)

Parameter QoS yang ingin diketahui adalah *Throughput*, *Packet Loss*, dan *Delay*. Dihitung dengan persamaan 2.1, 2.2, dan 2.3.

$$\begin{aligned}
 \text{c. } \textit{Throughput} &= \frac{\textit{Bytes}}{\textit{Time Span (s)}} \\
 &= \frac{51699302 \text{ Bytes}}{78.236 \text{ s}} \\
 &= 660.812 \text{ KB/s} \times 8 = 5.28 \text{ Kbps}
 \end{aligned}$$

$$\begin{aligned}
 \text{d. } \textit{Packet Loss} &= \frac{\textit{Paket dikirim} - \textit{Paket diterima}}{\textit{Paket diterima}} \times 100\% \\
 &= \frac{47174 - 47174}{47174} \times 100\% = 0\%
 \end{aligned}$$

$$\begin{aligned}
 \text{e. } \textit{Delay} &= \frac{\textit{Waktu pengiriman data}}{\textit{Paket diterima}} \\
 &= \frac{78.236}{47174} = 1.658 \text{ ms}
 \end{aligned}$$



## © Hak Cipta milik Politeknik Negeri Jakarta

### 4.2.4. Analisa Data

Pengukuran *Quality of Service* (QoS) dilakukan selama tiga hari pada waktu yang berbeda pagi, siang, dan malam untuk mengevaluasi kinerja jaringan yang terhubung dengan Firebase database melalui aplikasi Android. Setiap sesi pengukuran melibatkan analisis throughput, packet loss, dan delay dengan menggunakan Wireshark untuk memastikan data yang diambil berasal dari Firebase. Pada hari pertama, pengukuran dilakukan pada pukul 09.34 WIB, 15.50 WIB, dan 21.06 WIB. Di pagi hari, *throughput* mencapai 1.474 Kbps dengan *delay* 5.75 ms dan *packet loss* 0%, menunjukkan jaringan yang sangat baik. Siang harinya, *throughput* turun menjadi 170 Kbps dengan *delay* 32.11 ms, yang masih cukup baik meski terjadi peningkatan delay. Pada malam hari, *throughput* menurun signifikan menjadi 50 Kbps dengan *delay* 69.12 ms, mengindikasikan jaringan yang lebih lambat.

Pada hari kedua, pengukuran dilakukan pada pukul 05.04 WIB, 14.34 WIB, dan 21.06 WIB. Hasil pengukuran pagi hari menunjukkan *throughput* 201.76 Kbps dengan *delay* 14.81 ms dan *packet loss* 0%, yang menandakan jaringan cukup stabil di pagi hari. Siang hari, *throughput* sedikit menurun menjadi 170 Kbps dengan *delay* 32.11 ms, serupa dengan hari pertama. Pada malam hari, *throughput* kembali menurun ke 50 Kbps dengan *delay* meningkat menjadi 69.12 ms, konsisten dengan pengukuran sebelumnya yang menunjukkan penurunan performa jaringan di malam hari.

Pada hari ketiga, dari hasil pengukuran *Quality of Service* (QoS) yang dilakukan pada pagi, siang, dan malam hari, dapat disimpulkan bahwa jaringan memiliki performa yang berbeda-beda tergantung pada waktu pengujian. Pada pagi hari, *throughput* jaringan tercatat sebesar 201.76 Kbps, jauh lebih tinggi dibandingkan *throughput* pada siang dan malam hari yang masing-masing hanya sebesar 4.40 Kbps dan 5.28 Kbps. Hal ini menunjukkan bahwa pada pagi hari, jaringan lebih bebas dari kemacetan, kemungkinan karena jumlah pengguna yang lebih sedikit, sehingga kecepatan transfer data dari perangkat ke Firebase database lebih tinggi. Meskipun begitu, *delay* pada pagi hari mencapai 4.08 ms, lebih tinggi dibandingkan siang dan malam hari yang hanya 1.876 ms dan 1.658 ms, yang menunjukkan adanya sedikit peningkatan waktu pengiriman data di pagi hari. Di

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

sisi lain, tidak ada packet loss yang terdeteksi pada ketiga waktu pengukuran, menunjukkan bahwa jaringan sangat stabil dan andal, dengan semua data berhasil dikirim dan diterima tanpa ada yang hilang. Secara keseluruhan, jaringan yang diuji menunjukkan performa yang sangat baik untuk komunikasi antara perangkat dan Firebase database, dengan variasi performa yang bisa dipengaruhi oleh beban jaringan pada waktu yang berbeda.

Secara keseluruhan, pengukuran QoS ini menunjukkan bahwa jaringan memiliki performa terbaik di pagi hari dengan *throughput* yang lebih tinggi dan *delay* yang lebih rendah, sedangkan performa menurun secara signifikan pada malam hari dengan *throughput* yang lebih rendah dan *delay* yang lebih tinggi. Hasil ini penting untuk dipertimbangkan dalam aplikasi yang membutuhkan konektivitas *real-time*, terutama pada malam hari saat performa jaringan cenderung menurun.

### 4.3 Pengujian Internet Menggunakan Speedtest

Pengujian ini dilakukan untuk mengetahui kecepatan jaringan internet agar saling terhubung sehingga mikrokontroller dapat menerima data sensor ultrasonik dan mengirimkan data sehingga dapat menggerakkan servo.

#### 4.3.1 Deskripsi Pengujian

Pengujian internet menggunakan Speedtest adalah metode untuk mengukur performa koneksi internet. Speedtest mengevaluasi tiga parameter utama: ping, kecepatan unduh (*download speed*), dan kecepatan unggah (*upload speed*). Ping mengukur waktu respons koneksi, di mana nilai yang lebih rendah menunjukkan koneksi yang lebih cepat dalam merespons permintaan. Kecepatan unduh mengukur seberapa cepat data dapat diterima dari internet ke perangkat pengguna, yang penting untuk kegiatan seperti *streaming* video dan mengunduh file. Sementara itu, kecepatan unggah menunjukkan seberapa cepat data dapat dikirim dari perangkat ke internet, penting untuk kegiatan seperti mengirim atau menerima data. Untuk melakukan pengujian, pengguna hanya perlu mengakses aplikasi atau situs web Speedtest dan memulai tes, setelah itu hasil akan segera ditampilkan, memberikan gambaran yang jelas tentang kualitas dan kecepatan koneksi internet yang digunakan. Pengujian ini dilakukan dengan membandingkan kualitas internet menggunakan seluler dengan provider Three. Pada pengujian ini menggunakan



**Hak Cipta :**

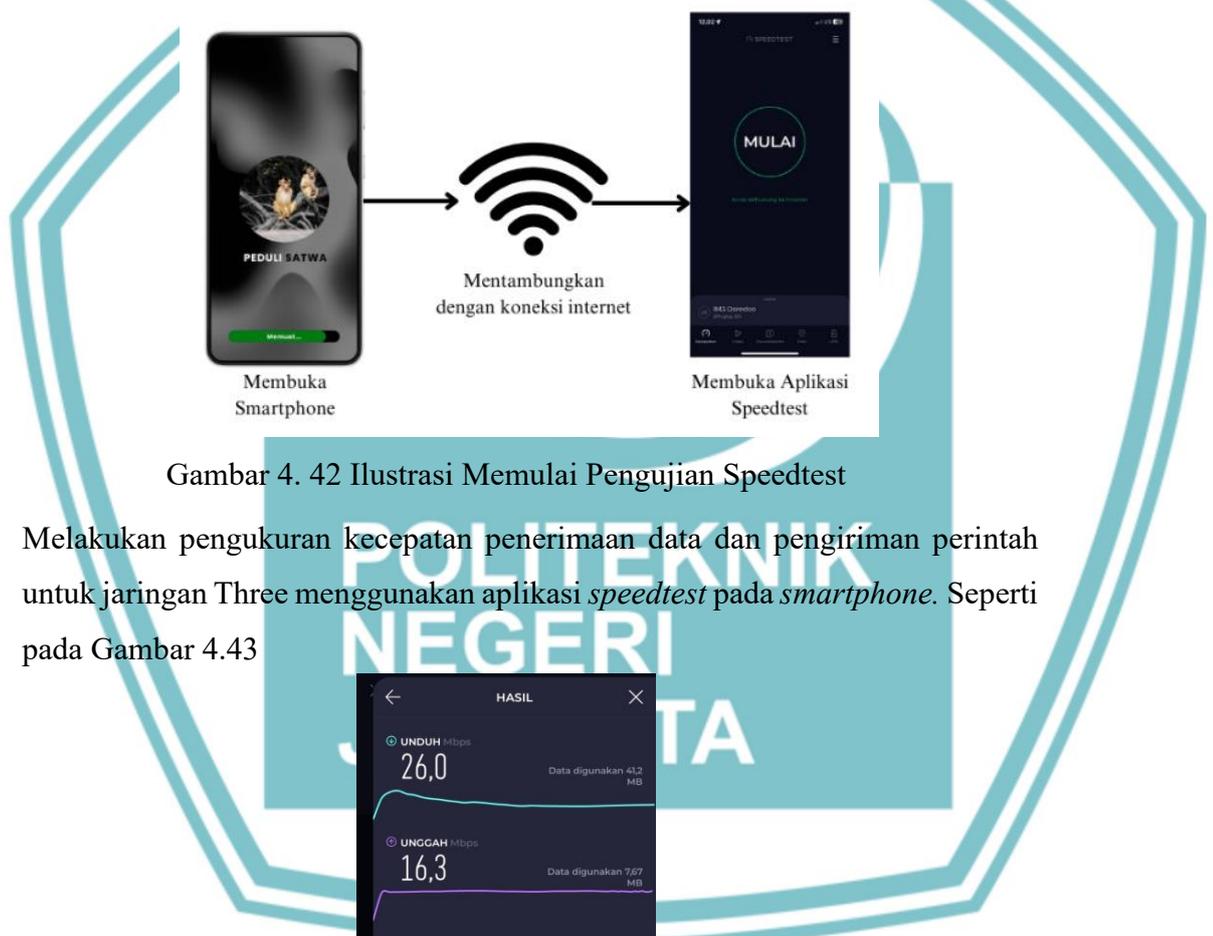
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

*smartphone* yang terhubung dengan jaringan seluler Three. Alat yang digunakan antara lain, yaitu *smartphone* dan Speedtest

#### 4.3.3 Prosedur Pengujian

Terdapat beberapa tahapan untuk melakukan pengujian internet dengan menggunakan Speedtest:

1. Menyambungkan *smartphone* dengan internet dan membuka aplikasi *speedtest*.  
Ilustrasi penyambungan *smartphone* ke internet dan membuka aplikasi Speedtest yang sudah terinstal pada *smartphone*.



Gambar 4. 42 Ilustrasi Memulai Pengujian Speedtest

2. Melakukan pengukuran kecepatan penerimaan data dan pengiriman perintah untuk jaringan Three menggunakan aplikasi *speedtest* pada *smartphone*. Seperti pada Gambar 4.43



Gambar 4. 43 Hasil Pengujian Speedtest



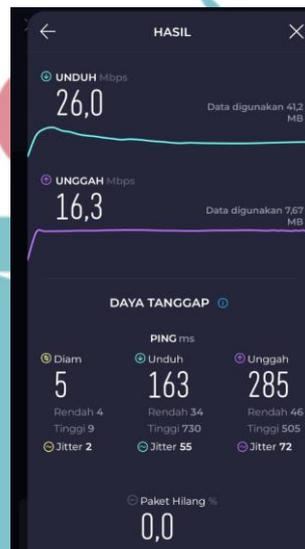
**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3. Mencatat hasil pengujian kecepatan penerimaan data status dan pengiriman perintah berupa data. Hasil tersebut akan menentukan kualitas jaringan yang sedang digunakan pada saat itu dan juga kualitas pengiriman serta penerimaan data dari ataupun ke Firebase *database*.

#### 4.3.4 Data Hasil Pengujian

Hasil pengujian yang telah dilakukan untuk mendapatkan hasil *upload* dan *download* menggunakan aplikasi Speedtest by Ookla dan menggunakan provider Three. Hasil pengujian Speedtest yang dilakukan dapat dilihat pada Gambar 4.44.



Gambar 4. 44 Hasil Pengukuran Speedtest

Pada pengujian ini meliputi kecepatan pengambilan data dan pengiriman perintah dari operator Three. Hasil yang didapatkan pada saat pengujian Speedtest dengan menggunakan provider Three dengan kecepatan penerimaan data berupa status sebesar 26 Mbps dan kecepatan pengiriman perintah sebesar 16.3 Mbps.

#### 4.3.5 Analisa Data

Pengujian kecepatan internet menggunakan aplikasi Speedtest dilakukan untuk mengukur kecepatan internet Three agar penguji dapat mengetahui dan membandingkan kecepatan internet dalam menerima dan mengirim data kepada Firebase. Kecepatan internet menggunakan jaringan internet Three mendapatkan hasil penerimaan data status sebesar 26 Mbps dan kecepatan pengiriman perintah sebesar 16.3 Mbps. Secara keseluruhan, nilai kecepatan pada provider Three sangat baik untuk menerima dan mengirim data dari mikrokontroler kepada Firebase



## © Hak Cipta milik Politeknik Negeri Jakarta

karena provider tersebut mendapatkan jaringan yang bagus. Maka dari itu pada pengujian Tugas Akhir ini menggunakan provider Three sebagai koneksi internet.

### 4.4 Analisa Sistem

Pengujian yang dilakukan mendapatkan beberapa Kesimpulan, aplikasi Android ‘Perangkap Satwa’ dapat terintegrasi dengan sistem *hardware* melalui hubungan Firebase *database*. Aplikasi dapat menampilkan data jumlah pengunjung, kondisi karya, dan Lokasi karya dengan baik. Selain itu, dapat memantau dan mengendalikan perangkat satwa di aplikasi hingga pengiriman data perintah ke Firebase *database*.

*Quality of Service* pada jaringan yang digunakan memiliki hasil yang baik terutama pada hari pertama. Pengiriman data dari Firebase ke sistem atau sebaliknya, berjalan dengan baik dan tidak ada paket yang hilang. Pada pengujian performasi jaringan, dapat disimpulkan bahwa di daerah ini provider Three memiliki kualitas sinyal cukup bagus. Nilai RSRP berada dalam kategori sangat baik hingga baik.

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

POLITEKNIK  
NEGERI  
JAKARTA

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB V PENUTUP

### 5.1 Kesimpulan

Berdasarkan hasil pengujian dari aplikasi pameran karya seni, dapat disimpulkan bahwa:

1. Prototipe perangkat satwa liar berbasis Android dirancang untuk memudahkan pengawasan dan kendali jarak jauh secara *real-time*. Terhubung dengan Firebase dan menggunakan sensor ultrasonik serta sistem transmisi ESP32 dan LoRa, aplikasi Android memungkinkan pengguna mengelola perangkat secara efisien, menawarkan solusi efektif untuk menjaga keamanan pemukiman dari satwa liar.
2. Aplikasi ini dirancang untuk memantau serta mengendalikan perangkat satwa. Ketika sensor ultrasonik mendeteksi keberadaan satwa, menu pemantauan dalam aplikasi akan menampilkan kondisi perangkat. Pengendalian dilakukan dengan membuka dan menutup perangkat melalui tombol perintah yang tersedia.
3. Aplikasi Android "Perangkat Satwa" menggunakan Firebase untuk memantau satwa dan mengendalikan perangkat secara jarak jauh melalui ESP32 dan LoRa, memungkinkan operasi perangkat secara *real-time*.
4. Pengukuran *Quality of Service* (QoS) selama tiga hari menunjukkan performa jaringan paling optimal terjadi di pagi hari, terutama pada hari pertama dengan *throughput* 1.474 Kbps, *delay* 5.75 ms, dan *packet loss* 0%. Siang hari menunjukkan penurunan *throughput* 170 Kbps dengan *delay* 32.11 ms, dan pada malam hari *throughput* menurun menjadi 50 Kbps, *delay* menjadi 69.12 ms. Provider Three sebagai koneksi internet mendapatkan hasil kecepatan penerimaan data 26 Mbps dan pengiriman perintah 16.3 Mbps, provider sangat baik untuk komunikasi yang andal antara mikrokontroler dan Firebase.

### 5.2 Saran

Dari tugas akhir “Rancang Bangun Prototype Sistem Perangkat Satwa Liar di Pemukiman Penduduk Berbasis Android” ini, penulis menyarankan untuk adanya pengembangan dalam pembuatan aplikasi, melengkapi sistem autentikasi untuk membuat akun agar terjamin keamanan data atau akun setiap pelindung satwa.



## DAFTAR PUSTAKA

- Adzan Abdul Zabbar, F. N. (2015). Keamanan HTTP dan HTTPS Berbasis Web Menggunakan Sistem Operasi Kali Linux. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 69-74.
- Anisya Sonita, R. F. (2018). Aplikasi E-Order Menggunakan Firebase dan Algoritme Knuth Morris Pratt Berbasis Android. *Jurnal Pseudocode*, 38-45.
- Erwin, S. M. (2023). *Pengantar & Penerapan Internet Of Things: Konsep Dasar & Penerapan IoT di Berbagai Sektor*. Jambi: PT. Sonpedia Publishing Indonesia.
- Lutvi Aditya Rahmadan, S. Z. (2023). Perancangan Media Pembelajaran Kewirausahaan Berbasis Android Menggunakan Kodular pada Kelas XII MAN 1 Agam. *JATI(Jurnal Mahasiswa Teknik Informatika)*, 1-7.
- Muyasir, R. M. (2022). Perancangan Aplikasi Media Pembelajaran Dasar Desain Grafis Berbasis Android Menggunakan Web Kodular. *JINTECH: Journal of Information Technology*, 22-28.
- Ratrie Lara Ditha, S. T. (2023). Rancang Bangun Aplikasi Layanan Pengaduan pada Dinas Pendidikan Kabupaten OKU Berbasis Android Menggunakan Android Studio. *JIK, VOL. 14, No.2*, 25-35.
- Rohaya, S. (2008). Internet: pengertian, sejarah, fasilitas dan koneksinya. *Jurnal/Fihris/Fihris Vol. III No. 1,2, 2*.
- Romasella Tri Novita, R. T. (2021). Analisis Keamanan Wifi Menggunakan Wireshark . *JES (Jurnal Elektro Smart) Vol. 1, No. 1*, 7-9.
- Selay, A. A. (2022). Internet Of Things. *arimah Tauhid, 1*, 860-868.
- Triawan, M. A. (2023). Sistem Pemantauan Lingkungan Menggunakan Sensor BME280 Berbasis Internet of Things. *Generic*, 37-41.
- Wulandari, R. (2016). Analisis QoS (Quality of Service) pada Jaringan Internet (Studi Kasus : Upt Loka Uji Teknik Penambangan Jampang Kulon-Lipi). *Jurnal Teknik Informatika dan Sistem Informasi, vol. 2, no. 2*, 1-11.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritis atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## DAFTAR RIWAYAT HIDUP



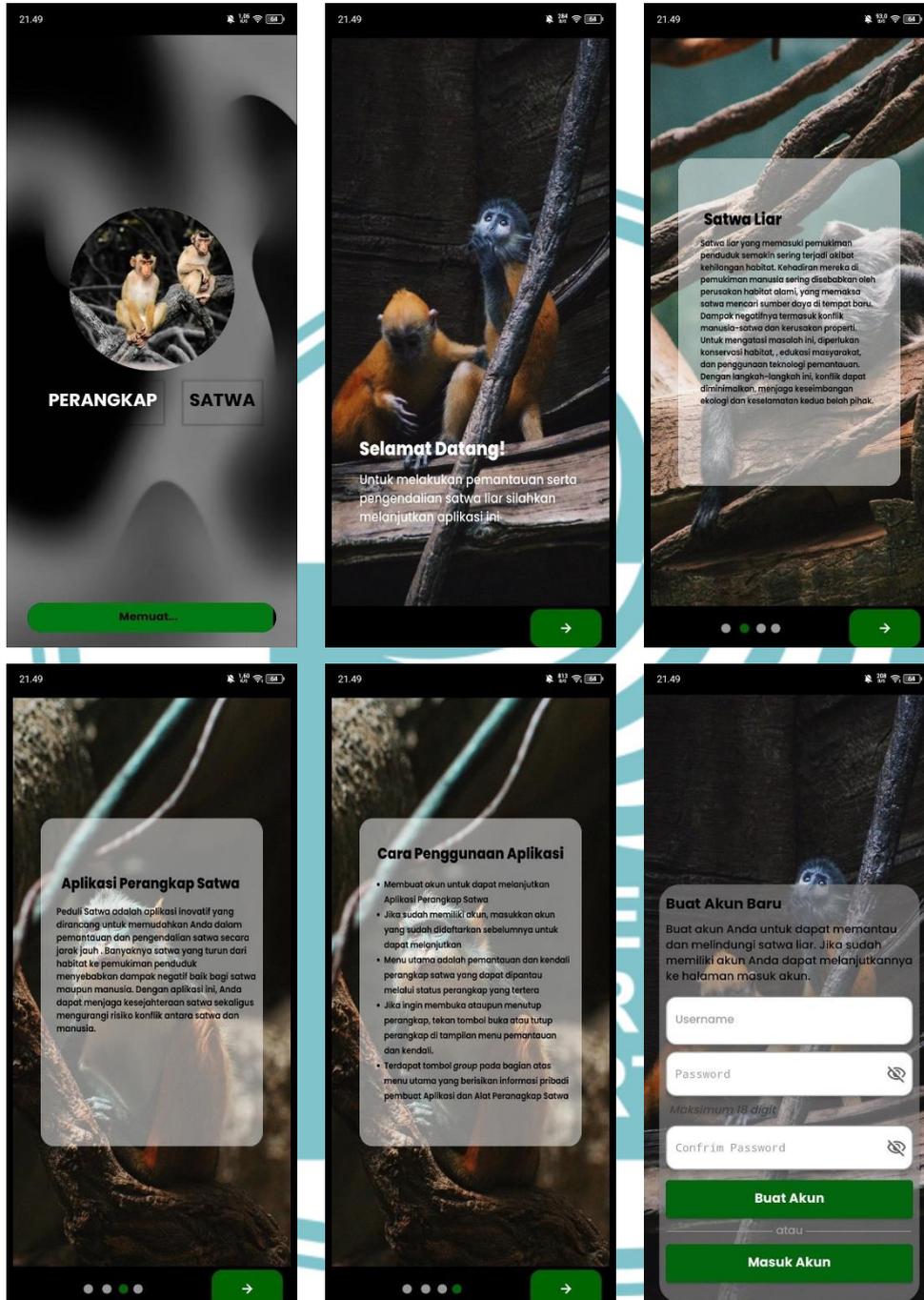
Rizka Dwi Sakha, lahir di Jakarta pada tanggal 20 Agustus 2003. Memulai pendidikan formal di SDIT Daarul Fataa Kabupaten Bogor pada Tahun 2009 hingga lulus pada Tahun 2015. Setelah itu melanjutkan Pendidikan ke SMPN 1 Bojonggede dan lulus pada Tahun 2018. Penulis melanjutkan Pendidikan ke SMAN 1 Tajurhalang dan lulus pada Tahun 2021. Setelah lulus dari Sekolah Menengah Atas, penulis melanjutkan pendidikan ke jenjang Diploma III di Jurusan Teknik Elektro Program Studi Telekomunikasi Politeknik Negeri Jakarta.

### Hak Cipta milik Politeknik Negeri Jakarta

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



```

when Screen1.Initialize
do
  call Animation_Utilities1.Overshoot Vertical
  component Vertical_Arrangement1
  start Position 1000
  end Position 0
  duration 1400
  tension 0

  call Animation_Utilities1.Overshoot Vertical
  component Label_PEDULI
  start Position 200
  end Position 0
  duration 2500
  tension 0

  call Animation_Utilities1.Overshoot Vertical
  component Card_PEDULI
  start Position -500
  end Position 0
  duration 2000
  tension 0

  call Animation_Utilities1.Overshoot Vertical
  component Card_SATWA
  start Position -500
  end Position 0
  duration 2500
  tension 0

  call Animation_Utilities1.Overshoot Vertical
  component Card_BackgroundMemuat
  start Position 0
  end Position 500
  duration 50
  tension 0

  set Clock1.Timer Enabled to true

when Clock1.Timer
do
  set Clock1.Timer Enabled to false
  call Animation_Utilities1.Overshoot Vertical
  component Card_BackgroundMemuat
  start Position 500
  end Position 0
  duration 1400
  tension 0

  call Animation_Utilities1.Overshoot Horizontal
  component Card_HijauMemuat
  start Position -1000
  end Position 0
  duration 4000
  tension 0

  call Animation_Utilities1.Overshoot Horizontal
  component Label_Memuat
  start Position 1000
  end Position 0
  duration 3000
  tension 0

  set Clock2.Timer Enabled to true

when Clock2.Timer
do
  set Clock2.Timer Enabled to false
  call Animation_Utilities1.Overshoot Vertical
  component Vertical_Arrangement1
  start Position 0
  end Position call Animation_Utilities1.Get Bottom Position
  component Vertical_Arrangement1
  duration 3000
  tension 0

  open another screen screenName Tentang_Aplikasi
  
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

when Register.Initialize
do
  call Decoration1.Set Shape
  component Horizontal_Arrangement1
  background Color
  border Color
  is Round true
  call Decoration1.Set Shape
  component Horizontal_Arrangement1_copy
  background Color
  border Color
  is Round true
  call Decoration1.Set Shape
  component Horizontal_Arrangement1_copy_copy
  background Color
  border Color
  is Round true

when Button2.Click
do
  open another screen screenName Menu_Masuk

initialize global name to 0

when Button3.Click
do
  if get global name = 0
  then
    set Button3.Image to visibility_off.png
    set Password_Box1.Input Type to 2
    set global name to 1
  else
    set Button3.Image to visibility.png
    set Password_Box1.Input Type to 1
    set global name to 0

when Button3_copy.Click
do
  if get global name = 0
  then
    set Button3_copy.Image to visibility_off.png
    set ConfrimPassword_Box1.Input Type to 2
    set global name to 1
  else
    set Button3_copy.Image to visibility.png
    set ConfrimPassword_Box1.Input Type to 1
    set global name to 0

when Button1.Click
do
  if call Network1.Is Connected
  then
    if is empty Username_Box1.Text or is empty Password_Box1.Text or false is empty ConfrimPassword_Box1.Text
    then
      call Notifier1.Show Alert
      notice Isi Semua Kolom!
    else
      if Password_Box1.Text = ConfrimPassword_Box1.Text
      then
        set Firebase_Database1.Project Bucket to join User/ Username_Box1.Text
        call Firebase_Database1.Get Value
        tag Username
        value If Tag Not There
      else
        call Notifier1.Show Alert
        notice Password Tidak Samal
    else
      call Notifier1.Show Alert
      notice Tidak Ada Koneksi internet!
  
```

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

when Firebase_Database1 Got Value
tag value
do set Firebase_Database1 . Project Bucket to join *User*
   Username_Box1 . Text
if get value == *
then call Firebase_Database1 . Store Value
   tag Username
   value To Store Username_Box1 . Text
call Firebase_Database1 . Store Value
   tag Password
   value To Store Password_Box1 . Text
call Notifier1 . Show Alert
   notice Username Sudah Terdaftar! Silahkan login

when Menu_Masuk .Initialize
do call Decoration1 .Set Shape
   component Horizontal_Arrangement1
   background Color
   border Color
   is Round true
call Decoration1 .Set Shape
   component Horizontal_Arrangement1_copy
   background Color
   border Color
   is Round true

when Button2 .Click
do open another screen screenName Register

initialize global name to 0

when Button3 .Click
do if get global name == 0
then set Button3 . Image to visibility_off.png
   set Password_Box1 . Input Type to 2
   set global name to 1
else set Button3 . Image to visibility.png
   set Password_Box1 . Input Type to 1
   set global name to 0

initialize global Username to * *

when Button1 .Click
do if call Network1 .Is Connected
then if Username_Box1 . Text != *
then if Password_Box1 . Text != *
then set Firebase_Database1 . Project Bucket to join *User*
   Username_Box1 . Text
call Firebase_Database1 .Get Value
   tag Password
   value If Tag Not There
else call Notifier1 .Show Alert
   notice !Is! Password!
else call Notifier1 .Show Alert
   notice !Is! Username!
else call Notifier1 .Show Alert
   notice !Tidak Ada Koneksi Internet!

```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

when Firebase_Database1 .Got Value
tag value
do
  if get value != ""
  then
    if get value == Password_Box1 . Text
    then
      call Notifier1 .Show Alert
      notice "Login Berhasil!"
      set global Username to Username_Box1 . Text
      open another screen with start value screenName Kontrol_Monitor
      startValue get global Username
    else
      call Notifier1 .Show Alert
      notice "Password Salah!"
    else
      call Notifier1 .Show Alert
      notice "Username Salah!"

when Kontrol_Monitor .Initialize
do
  call Decoration1 .Set Shape
  component Tombol_Buka
  background Color
  border Color
  is Round true
  call Decoration1 .Set Shape
  component Tombol_Tutup
  background Color
  border Color
  is Round true
  set Firebase_Database1 . Project Bucket to "Perangkap"

when Firebase_Database1 .Got Value
tag value
do
  set Firebase_Database1 . Project Bucket to "Perangkap"
  if get tag == "Status"
  then
    if get value == 0
    then
      set PemberitahuanKondisi .Text to "Terdapat Satwa Liar"
    else
      set PemberitahuanKondisi .Text to "Tidak Terdapat Satwa Liar"

when Tombol_TentangKami .Click
do
  open another screen screenName Menu_TentangKami

when Tombol_Keluar .Click
do
  open another screen screenName Tentang_Aplikasi

when Tombol_Buka .Click
do
  set Firebase_Database1 . Project Bucket to "Perangkap"
  call Firebase_Database1 .Store Value
  tag "Status"
  value To Store 1

when Tombol_Tutup .Click
do
  set Firebase_Database1 . Project Bucket to "Perangkap"
  call Firebase_Database1 .Store Value
  tag "Status"
  value To Store 0
    
```



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
when Firebase_Database1 .Data Changed
tag value
do set Firebase_Database1 . Project Bucket to Perangkap
if get tag == Status
then if get value == 0
then set PemberitahuanKondisi . Text to Terdapat Satwa Liar
set Image2 . Visible to true
set Image1 . Visible to false
else set PemberitahuanKondisi . Text to Tidak Terdapat Satwa Liar
set Image1 . Visible to true
set Image2 . Visible to false
if compare texts get tag == Status
then set Label6 . Text to join Status:
get value

when Button1 .Click
do open another screen screenName Kontrol_Monitor

when Button2 .Click
do open another screen screenName Tentang_Aplikasi
```