

BAB I

PENDAHULUAN

A. Latar Belakang

Dalam era globalisasi yang pesat ini, perkembangan teknologi telah membawa banyak kemajuan dalam berbagai aspek kehidupan. Penggunaan teknologi dalam menyelesaikan pekerjaan menjadi keharusan untuk meningkatkan efisiensi dan keamanan. Sistem pengamanan menjadi salah satu aspek yang penting dalam kehidupan manusia. Dalam banyak kasus, penjagaan oleh manusia secara langsung memiliki kelemahan karena dapat mengalami kelelahan dan kejenuhan, yang berdampak pada menurunnya keamanan itu sendiri.

Salah satu sektor yang dapat memanfaatkan teknologi pengamanan adalah peternakan, terutama peternakan ayam. Ayam adalah jenis hewan ternak yang memiliki nilai ekonomi tinggi karena dapat dimanfaatkan daging dan telurnya. Memelihara ayam bukan hanya sebagai hobi, tetapi juga sebagai bisnis yang menguntungkan. Namun, masalah keamanan di peternakan sering menjadi perhatian, mengingat potensi gangguan dari pencuri.

Mengacu pada penelitian sebelumnya tentang "Alat Pengaman Kandang Berbasis Mikrokontroler Arduino Uno" oleh Wire Bagye dan Tsurayya Azizah(2018), aplikasi pengaman kandang berbasis Android merupakan solusi yang inovatif dan efisien. Dalam aplikasi ini, digunakan alat Raspberry Pi dengan webcam dan sensor PIR sebagai sistem pengamanan yang terintegrasi. Sistem

pengamanan ini dapat digunakan untuk pemantauan pemilik kandang melalui aplikasi Android, sehingga memastikan keamanan kandang tetap terjaga.

Aplikasi pengaman kandang berbasis Android dengan mengacu pada penggunaan alat berbasis Orange Pi, webcam, dan sensor PIR serta penelitian sebelumnya tentang alat pengaman berbasis Arduino Uno menjadi sebuah solusi modern dan efektif dalam meningkatkan keamanan di sektor peternakan. Berdasarkan uraian latar belakang tersebut, maka penulis melakukan penelitian yang berjudul “Aplikasi Pengaman Kandang Berbasis Android”.

B. Rumusan Masalah

Berdasarkan dari uraian latar belakang diatas, maka rumusan masalah yang akan dibahas adalah bagaimana Merancang sebuah Aplikasi yang dapat memantau Kandang melalui smartphone Android?

C. Batasan Masalah

Penelitian ini memiliki beberapa batasan masalah dengan harapan penelitian terfokus dengan batasan-batasan yang dibuat. Adapun batasan masalah tersebut adalah sebagai berikut:

1. Alat yang dibangun ini bersifat prototype.
2. Prototype ini menggunakan sensor *Passive Infrared Sensor* (PIR) yang hanya dapat menerima pancaran cahaya inframerah.
3. Prototype tidak melakukan kontak langsung terhadap objek penelitian karena alat ini hanya menerima respon inframerah dan melakukan tahapan prosedur alat.

D. Tujuan Penelitian

Sesuai dengan permasalahan dan batasan masalah yang telah diuraikan diatas, maka tujuan penelitian ini adalah untuk membuat alat pengaman kandang menggunakan orange pi dan webcam serta sensor Passive Infrared Reciever (PIR) serta membangun sebuah aplikasi Android yang saling terintegrasi.

E. Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk dapat mengetahui pemanfaatan Smartphone Android dalam memantau kandang peternak.

F. Sistematika Penulisan

Agar pembahasan lebih sistematis, maka tulisan ini dibuat dalam lima bab, yaitu:

BAB I PENDAHULUAN

Berisi tentang uraian latar belakang pemilihan judul perancangan alat pengaman kandang menggunakan sensor gerak pir berbasis android, rumusan masalah, batasan masalah, tujuan penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Dalam bab ini berisikan tentang teori-teori yang berkaitan dengan mikrokontroller, sensor Passive Infrared Sensor (PIR), Bahasa pemrograman Kotlin, Python, library, dan Orange Pi.

BAB III METODE PENELITIAN

Dalam bab ini menjelaskan mengenai tahapan-tahapan yang dilalui dalam penyelesaian penelitian ini, yaitu tempat penelitian, waktu penelitian, metode pengumpulan data, alat, dan bahan penelitian, tahapan penelitian, metode pengujian serta gambaran desain sistem yang akan dirancang dan dibuat.

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

Dalam bab ini menjelaskan mengenai rancangan sistem yang telah dibuat, metode dan teknik pengujian.

BAB V KESIMPULAN DAN SARAN

Dalam bab ini dipaparkan mengenai kesimpulan dan saran dari hasil.

BAB II

TINJAUAN PUSTAKA

A. Tinjauan Penelitian Terdahulu

Penelitian ini dilaksanakan dengan memperhatikan hasil-hasil penelitian sebelumnya sebagai perbandingan dan telaahan. Pada penelitian Wire Bagye dan Tsurayya Azizah, (2018). dengan judul Alat Pengaman Kandang Berbasis Mikrokontroler Arduino Uno, Dalam penelitian ini, para peneliti menggunakan Arduino Uno untuk membuat sistem pengaman kandang yang efektif dan efisien. Sistem ini dirancang untuk dapat mendeteksi pergerakan binatang di dalam kandang dan memberikan peringatan kepada peternak jika ada binatang yang keluar dari kandang. Penelitian ini memiliki kelebihan pada penggunaan sensor pir dalam mendeteksi jika ada seseorang yang masuk kedalam kandang ternak, adapun kelemahan sistem ini terletak pada jenis *prototype* yang baru menggunakan satu sensor.

Dalam penelitian mengenai Pembuatan Prototipe Sistem Keamanan Laboratorium Berbasis Arduino Mega yang dilakukan oleh Gagat Mughni Pradiptal, Nida Nabilah, dkk. Sistem keamanan laboratorium yang dikembangkan ini terdiri dari berbagai sensor dan perangkat elektronik yang terhubung dengan mikrokontroler Arduino Mega. Penelitian ini memiliki kelebihan pada Arduino mega yang memiliki memori penyimpanan dan memori flash yang lebih besar dibandingkan Arduino uno. Penelitian ini menciptakan sistem keamanan tetapi

untuk pengamanan suatu ruangan. Adapun kelemahan penelitian ini sensor PIR yang digunakan hanya untuk menerima signal Inframerah yang kemudian sistem ini belum menggunakan sistem feedback jika sensor PIR menerima signal orang yang berada dalam jangkauan sensornya.

Penelitian yang berjudul "Penggunaan Sensor Penerima *Infrared Pasif* (PIR) untuk Mendeteksi Pergerakan Berbasis Gerbang Layanan Pesan Singkat (*Short Message Service*)" oleh Rozali Toyib, Iwan Bustami, dkk. PIR digunakan untuk mendeteksi gerakan pada area yang diinginkan, kemudian mengirimkan informasi melalui *Short Message Service Gateway* (SMSC) kepada penerima yang ditentukan. Tujuan dari penelitian ini adalah membuat sebuah sistem yang efisien dalam mendeteksi pergerakan dan menyampaikan informasi dengan cepat dan akurat kepada penerima. Penelitian ini memiliki kelebihan yaitu sistem yang dibangun menggunakan sismin Atmega dimana *developer* dapat melakukan custom sistem sendiri dan sekaligus memiliki kekurangan dalam tingkat kesulitan pembangunan sistem yang tidak seperti arduino uno yang siap pakai.

B. Android

Android merupakan sistem operasi untuk perangkat mobile yang dikembangkan oleh Google. Sistem operasi ini dirancang khusus untuk digunakan pada perangkat mobile seperti *smartphone*, tablet, dan *smartwatch*. Dalam pengembangannya, Android menggunakan bahasa pemrograman Java dan berbasis pada sistem operasi Linux. Keuntungan utama dari Android adalah kemampuannya untuk mendukung berbagai aplikasi dan ketersediaan beragam aplikasi melalui *Google Play Store*.

Android juga dikenal dengan antarmuka pengguna yang ramah pengguna dan mudah digunakan. Antarmuka layar sentuh yang intuitif memudahkan pengguna untuk beralih antara berbagai fungsi dan aplikasi. Android juga memiliki fitur-fitur canggih seperti pengenalan wajah, sensor sidik jari, dan asisten virtual seperti Google Assistant yang semakin mempermudah pengguna dalam mengakses informasi dan mengendalikan perangkat.

Selain itu, Android juga memiliki fleksibilitas yang tinggi dalam mengkustomisasi tampilan dan pengaturan perangkat. Pengguna dapat dengan mudah mengubah tema, ikon, dan gaya tampilan perangkat sesuai dengan preferensi masing-masing. Keunggulan ini menjadikan Android sebagai pilihan sistem operasi yang populer di kalangan pengguna global.

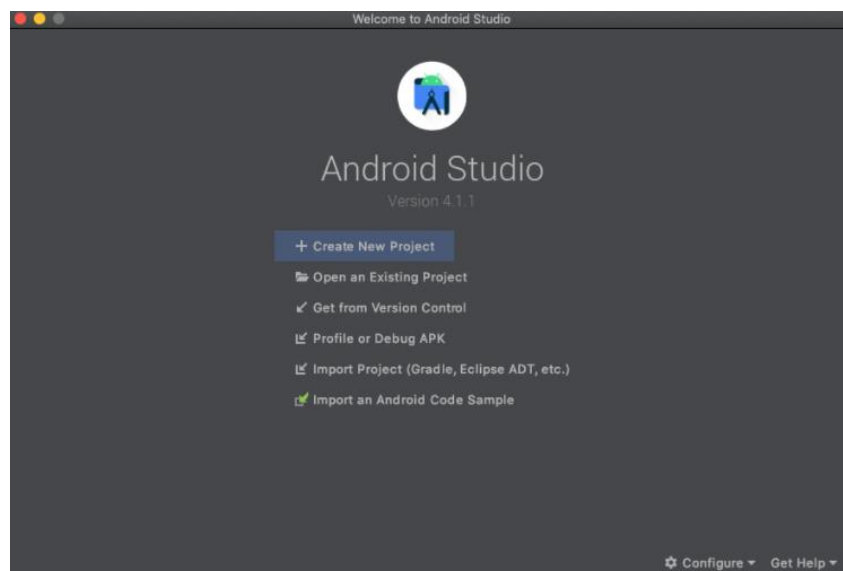
Android telah menjadi sistem operasi mobile yang mendominasi pasar smartphone dan perangkat mobile lainnya, berkat fitur-fitur melimpah, antarmuka pengguna yang intuitif, dan fleksibilitas tinggi. Android tidak hanya memengaruhi kemajuan teknologi, tetapi juga mengubah cara orang berinteraksi dengan perangkat mobile mereka dan mendapatkan informasi.

C. Android Studio IDE

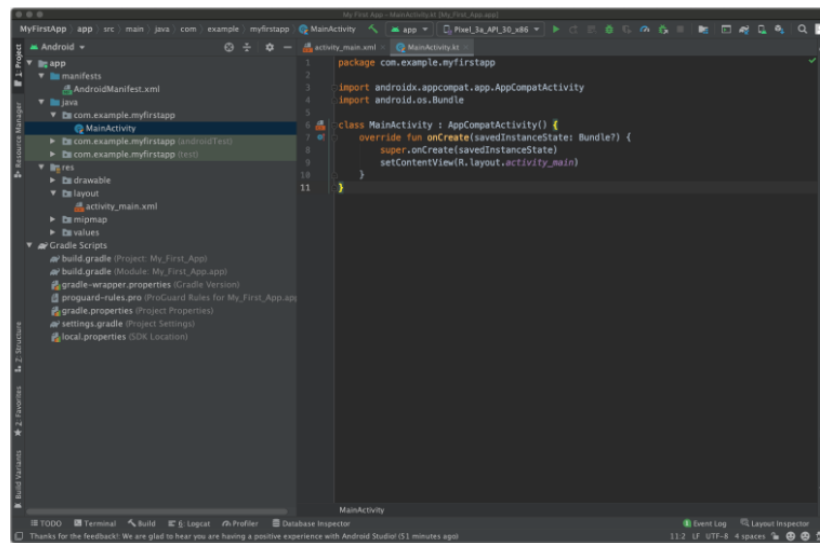
Android *Studio* merupakan sebuah *Integrated Development Environment* (IDE) resmi yang digunakan untuk mengembangkan aplikasi Android. IDE ini tersedia secara gratis atau sebagai open source. Pada konferensi Google I/O tahun 2013 pada tanggal 16 Mei 2013, Google mengumumkan perilisan Android Studio.

Sejak saat itu, Android Studio menjadi IDE resmi untuk pengembangan aplikasi Android menggantikan Eclipse. Android Studio dibangun berdasarkan IntelliJ IDEA, mirip dengan Eclipse, dan dilengkapi dengan plugin ADT. IDE ini memiliki berbagai fitur, antara lain:

1. Sebuah proyek build Gradle.
2. Pembaruan dan perbaikan kesalahan kode (*refactoring*).
3. Alat bernama "*Lint*" yang diaktifkan dapat memantau dengan cepat kecepatan, manfaat, dan daya saing sebuah aplikasi.
4. Untuk aspek keamanan, Android Studio mendukung tanda tangan *Proguard* dan aplikasi.
5. Penyederhanaan antarmuka pengguna (GUI) dari aplikasi Android. Setiap aplikasi yang dikembangkan didukung oleh *Google Cloud Platform*. Menurut Juansyah (2015).



Gambar 2.1 Tampilan awal android *studio*
Sumber: *Android Developer* (<https://developer.android.com>)



Gambar 2.2 Halaman kerja *android studio*

Sumber: *Android Developer* (<https://developer.android.com>)

D. Kotlin

Kotlin merupakan bahasa yang berjalan pada *Java Virtual Machine* (JVM). Bahasa pemrograman ini dikembangkan oleh perusahaan JetBrains dan sejak pertama kali diperkenalkan pada tahun 2011, Kotlin telah menjadi salah satu bahasa paling populer di kalangan pengembang perangkat lunak. Keunggulan interoperabilitas Kotlin dengan Java menjadi nilai tambah yang signifikan. Pengembang dapat dengan mudah mengintegrasikan kode Kotlin ke dalam proyek Java yang sudah ada dan sebaliknya. Hal ini memungkinkan migrasi proyek secara bertahap tanpa kehilangan fungsionalitas yang ada. Kotlin menawarkan sintaksis yang bersih dan ekspresif, membuatnya lebih mudah dipahami dan ditulis oleh para pengembang. Dengan fitur-fitur seperti *smart casts*, *null safety*, dan *extension functions*, Kotlin membantu mengurangi kesalahan pemrograman dan meningkatkan kualitas kode.



Gambar 2.3 Logo kotlin

Sumber: Mailslurp (www.mailslurp.com)

Bahasa ini dirancang untuk mendukung pengembangan aplikasi Android. Sejak Google secara resmi mendukung Kotlin sebagai bahasa pemrograman resmi untuk pengembangan Android pada tahun 2017, penggunaan Kotlin di dalam ekosistem Android semakin meningkat.

(Edeh Israel Chidera. 2023). Salah satu perbedaan paling signifikan antara Kotlin dan Java adalah sintaksisnya. Kotlin memiliki sintaksis yang lebih ringkas dibandingkan Java, yang berarti memerlukan lebih sedikit kode untuk melakukan operasi yang sama. Konsep seperti data yang tidak dapat diubah, lambda, dan fungsi tingkat tinggi didukung oleh Kotlin, memberikan kemampuan kepada para pengembang untuk menulis kode yang lebih deklaratif dan mudah diuji..

Tabel 2.1 Perbedaan penulisan bahasa java dan kotlin

Java	kotlin
<pre> public class MyClass { private int myField; public MyClass(int myField) { this.myField = myField; } public int getMyField() { return myField; } public void setMyField(int myField) { this.myField = myField; } } </pre>	<pre> class MyClass(private var myField: Int) { fun getMyField() = myField fun setMyField(value: Int) { myField = value } } </pre>

Dengan adanya Kotlin/*Native*, pengembang dapat menggunakan Kotlin untuk mengembangkan aplikasi selain platform JVM, seperti iOS dan bahkan aplikasi desktop. Hal ini memberikan opsi platform yang lebih fleksibel untuk pengembangan. Kotlin menyediakan alat pengembangan yang kuat, seperti IntelliJ IDEA, yang dikembangkan oleh JetBrains. IntelliJ IDEA memberikan dukungan penuh untuk Kotlin, termasuk *refactorings*, *debugging*, dan fitur-fitur pengembangan lainnya.

Kotlin memiliki komunitas yang aktif dan berkembang pesat. Terdapat berbagai sumber daya, tutorial, dan forum daring yang membantu pengembang dalam memahami dan menguasai Kotlin. Ini menciptakan lingkungan yang mendukung pertumbuhan dan pertukaran pengetahuan. Beberapa perusahaan terkemuka, seperti Netflix, Uber, dan Pinterest, telah mengadopsi Kotlin dalam pengembangan produk mereka. Hal ini menunjukkan bahwa Kotlin bukan hanya populer di kalangan komunitas pengembang, tetapi juga diakui oleh industri sebagai bahasa yang dapat diandalkan untuk proyek-proyek besar dan kompleks.

E. Orange Pi

Orange Pi adalah serangkaian komputer papan tunggal (SBC) yang diproduksi oleh Shenzhen Xunlong Software Co., Ltd. Seri *Orange Pi* dirancang untuk memberikan solusi komputasi terjangkau dan serbaguna dengan berbagai fitur yang dapat disesuaikan. Papan tersebut memiliki berbagai varian, termasuk model-model yang cocok untuk keperluan pengembangan, *media center*, atau proyek *Internet of Things* (IoT).

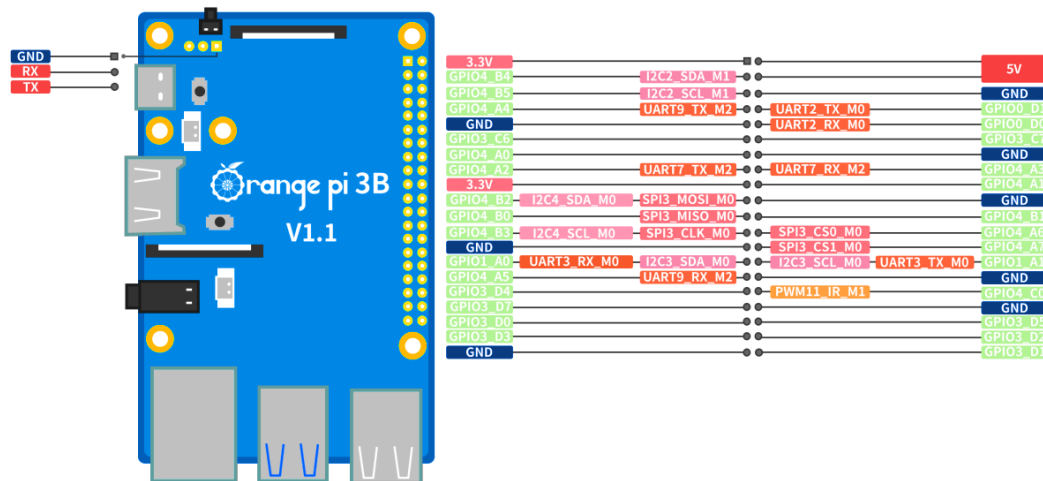
Salah satu varian *Orange Pi* yang populer adalah *Orange Pi 3 B*. Model ini memiliki spesifikasi yang cukup tangguh untuk kebutuhan komputasi sehari-hari. *Orange Pi 3 B* ditenagai oleh prosesor *Quad-core Cortex-A53 Allwinner H6*, dilengkapi dengan GPU *Mali-T720 MP2*. Papan ini mendukung koneksi *Wi-Fi* dan *Bluetooth*, memungkinkan integrasi yang mudah dengan jaringan dan perangkat lainnya. Selain itu, *Orange Pi 3 B* dilengkapi dengan port *USB*, slot kartu *microSD*, dan soket *HDMI* untuk kenyamanan pengguna dalam mengakses dan menyambungkan perangkat tambahan.



Gambar 2.4 Orange Pi 3 B
Sumber : Orange Pi Org (www.orangepi.org)

Dengan dukungan komunitas yang aktif dan harga yang terjangkau, *Orange Pi* dan *Orange Pi 3 B* menjadi pilihan menarik bagi para pengembang dan pembuat proyek DIY yang ingin mendapatkan keuntungan dari keberagaman fitur dan fleksibilitas yang ditawarkan oleh papan SBC ini. Meskipun bukan pesaing langsung dari *Raspberry Pi*, *Orange Pi* memiliki tempatnya sendiri dalam

ekosistem papan tunggal dengan menyediakan solusi yang bersaing dalam hal performa dan harga.



Gambar 2.5 Orange Pi 3 B *pinout*
Sumber: Orange Pi Org (www.orangepi.org)

Orange Pi 3 B memiliki keunggulan dalam hal konektivitas dengan dukungan untuk gigabit *Ethernet*, memungkinkan transfer data yang lebih cepat melalui kabel. Selain itu, papan ini dilengkapi dengan soket USB 3.0. memungkinkan pengguna untuk menghubungkan perangkat penyimpanan eksternal dengan kecepatan transfer yang lebih tinggi. Fitur ini membuatnya cocok untuk aplikasi yang membutuhkan akses cepat terhadap data atau proyek multimedia yang menuntut performa tinggi.

Dalam hal dukungan sistem operasi, *Orange Pi* dapat menjalankan berbagai distribusi Linux, seperti Ubuntu, Debian, dan lainnya. Selain itu, komunitas *Orange Pi* yang aktif juga memberikan dukungan untuk sistem operasi kustom dan proyek-proyek pengembangan yang beragam. Pilihan ini memberikan fleksibilitas kepada

pengguna untuk menyesuaikan pengalaman komputasi mereka sesuai dengan kebutuhan spesifik.

F. Python

(Sayuti Rahman, dkk. 2023). Python adalah bahasa pemrograman yang menggunakan interpreter untuk menjalankan kode programnya. Interpreter tersebut dapat menerjemahkan kode secara langsung, dan Python bisa berjalan pada berbagai platform seperti Windows, Linux, dan sebagainya. Python merupakan bahasa pemrograman tingkat tinggi yang mendapatkan popularitas pesat dalam dunia pengembangan perangkat lunak, dirancang dengan sintaksis yang mudah dipahami dan bersifat dinamis. Ini menjadikannya pilihan utama baik untuk pengembang pemula maupun yang berpengalaman, dengan mendukung paradigma pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional, Python memberikan tingkat fleksibilitas yang tinggi dalam menangani berbagai jenis proyek pengembangan perangkat lunak.

Tabel 2.2 Contoh penulisan bahasa python

```
@app.route('/start_camera', methods=['POST'])
def start_camera():
    global cap
    if not cap.isOpened():
        cap = cv2.VideoCapture(0)
        cap.set(4, 240)
    return redirect(url_for('index'))

@app.route('/stop_camera', methods=['POST'])
def stop_camera():
    global cap
    if cap.isOpened():
        cap.release()
    return redirect(url_for('index'))
```

Keunggulan Python tidak hanya terletak pada kejelasan sintaksisnya, tetapi juga pada ekosistemnya yang kaya dengan berbagai pustaka dan framework. Python digunakan secara luas dalam berbagai bidang seperti pengembangan web, pengolahan data, kecerdasan buatan, dan pengembangan perangkat lunak umum. Dengan berbagai modul seperti NumPy, Pandas, dan Django, Python menjadi pilihan kuat untuk berbagai proyek.

Selain itu, Python memiliki komunitas pengguna yang besar dan aktif, memberikan pengembang akses mudah ke dukungan dan sumber daya untuk mengatasi tantangan pengembangan yang dihadapi. Dengan filosofi "baterai termasuk," Python mendorong pengembang untuk fokus pada pemecahan masalah daripada mengelola detail teknis. Kombinasi kesederhanaan sintaksis, kekuatan fungsional, dan dukungan komunitas menjadikan Python bahasa yang sangat efisien dan cepat untuk prototip, pengujian, dan pengembangan perangkat lunak pada skala besar.

G. Sensor Pir Hc-Sr501

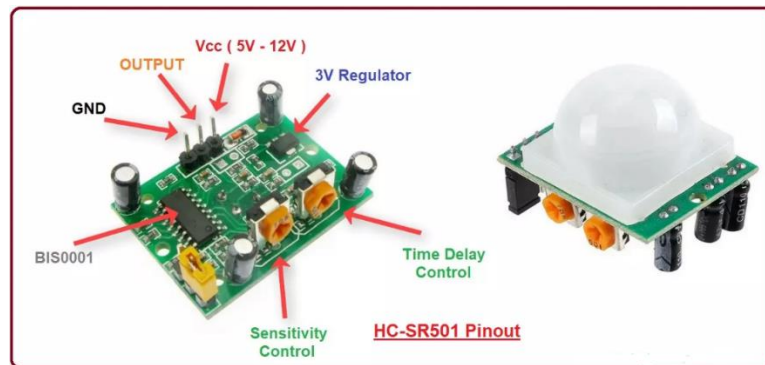
Sensor PIR (*Passive Infrared Sensor*) adalah alat sensor yang berfungsi untuk mendeteksi keberadaan pergerakan objek berdasarkan perubahan suhu yang dihasilkan oleh objek tersebut. Sensor ini menggunakan gelombang inframerah yang tak terlihat oleh mata manusia untuk deteksi pergerakan dan memicu tindakan tertentu, seperti menyalakan lampu atau mengaktifkan alarm. Sensor PIR umumnya digunakan dalam sistem keamanan, pengendalian lampu otomatis, dan berbagai aplikasi yang membutuhkan deteksi gerakan. Keunggulan sensor PIR adalah

sensitivitasnya terhadap pergerakan manusia dan hewan, serta penggunaan energi yang efisien.

Sensor PIR HC-SR501 merupakan salah satu jenis sensor PIR yang sering digunakan dalam proyek-proyek elektronika, sensor ini memiliki sensitivitas yang baik dan cukup akurat dalam mendeteksi gerakan manusia dan hewan. Selain itu, HC-SR501 dilengkapi dengan potensiometer yang memungkinkan pengguna untuk mengatur jarak deteksi dan sensitivitas sensor. Sensor ini juga sangat mudah digunakan karena cukup dihubungkan dengan board Arduino atau mikrokontroler lainnya melalui pin input/output digital. Sensor PIR HC-SR501 umumnya digunakan dalam berbagai aplikasi seperti pengendalian lampu otomatis, sistem keamanan, dan pengukuran suhu.

Sensor PIR HC-SR501 mampu mendeteksi gerakan manusia dan hewan dalam jarak tertentu hingga 7 meter dan sudut deteksi hingga 120 derajat. Sensor ini juga dilengkapi dengan delay time adjustment dan sensitivity adjustment yang dapat disesuaikan dengan kebutuhan pengguna. Sensor HC-SR501 dapat dioperasikan pada tegangan 5VDC dan memiliki konsumsi daya yang rendah.

Keunggulan sensor ini adalah kemampuannya dalam mendeteksi gerakan manusia dan hewan, serta mudah digunakan dan dikombinasikan dengan board mikrokontroler seperti Arduino.



Gambar 2.6 Sensor PIR HC-SR501

Sumber: The Eengineering Projects (www.theengineeringprojects.com)

H. Webcam

Webcam (kamera web) merupakan kamera yang menampilkan gambar secara real-time dan dapat diakses melalui *World Wide Web*, aplikasi pesan instan, atau aplikasi panggilan video. Sejauh ini, aplikasi webcam hanya digunakan untuk merekam dan menampilkan objek. (Julfikar Ali Andre. 2016).

Kita mengenal fungsi dari *webcam* yang memungkinkan kita mengirim pesan cepat, seperti videochat atau konferensi video pribadi. Webcam juga dapat digunakan untuk mentransfer media secara langsung, tetapi sebagian besar pengguna hanya menggunakannya untuk *videochat*.



Gambar 2.7 Webcam

Sumber: Logitech (www.logitech.com)

Gambar yang diperoleh dapat segera diolah secara digital dan dibagikan melalui internet. Namun, berbeda dengan kamera digital, webcam tidak memiliki kartu memori atau *flash memory card*. Desain webcam tidak memerlukan penyimpanan gambar karena fungsinya utamanya adalah melakukan perekaman dan pengiriman langsung gambar yang diperoleh.

I. *Flowchart*

Flowchart adalah salah satu diagram yang digunakan untuk menggambarkan suatu proses atau alur kerja dari suatu sistem atau program. Dalam bidang informatika, *flowchart* digunakan untuk merepresentasikan alur kerja dari suatu program atau sistem komputer. *Flowchart* dapat mempermudah pengembangan program karena dapat membantu pengembang dalam memahami alur kerja dari program yang dibangun.

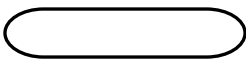
Menurut buku "*Information Systems Development: Reflections, Challenges and New Directions*" karya Robert D. Galliers dan Wendy L. Currie, *flowchart* merupakan alat bantu memiliki signifikansi penting dalam pengembangan sistem informasi. *Flowchart* dapat digunakan untuk merepresentasikan proses bisnis atau

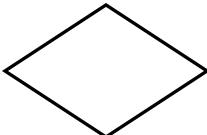
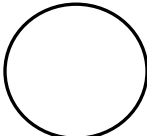
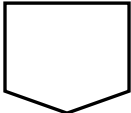
alur kerja dari suatu sistem informasi. Selain itu, *flowchart* juga dapat digunakan sebagai alat untuk mengevaluasi dan mengoptimalkan proses bisnis atau sistem informasi yang sudah ada.

Flowchart juga bisa dimanfaatkan untuk mempermudah komunikasi antara pengembang program dengan pihak-pihak terkait atau pengguna akhir. Dalam buku "*System Analysis and Design: An Object-Oriented Approach with UML*" karya Alan Dennis, Barbara Haley Wixom, dan David Tegarden, disebutkan bahwa *flowchart* dapat digunakan untuk memudahkan komunikasi antara pengembang program dengan pengguna akhir. *Flowchart* dapat membantu pengguna akhir dalam memahami alur kerja dari program yang dibangun sehingga dapat memberikan masukan yang lebih baik dalam proses pengembangan program.

Berikut merupakan beberapa simbol yang dipakai dalam membuat sebuah *flowchart* :

Tabel 2.3 Simbol *flowchart*

Simbol	Keterangan
	Simbol <i>Terminator</i> digunakan untuk menunjukkan awal atau akhir program.
	Garis alir (<i>flow line</i>) mengindikasikan arah aliran program.
	<i>Preparation</i> atau persiapan adalah proses awal untuk inialisasi

Simbol	Keterangan
	Proses proses kalkulasi atau pengolahan data
	<i>Input / output</i> atau data masukan/keluaran yaitu proses data masukan/keluaran, parameter, informasi
	Proses yang telah ditentukan sebelumnya (<i>sub program</i>) Inisiasi sub program / memulai eksekusi sub program
	<i>Decision</i> atau keputusan adalah tahap dimana pernyataan dibandingkan, data diseleksi, dan memberikan opsi untuk langkah selanjutnya.
	<i>On page connector</i> atau penghubung di dalam halaman merupakan elemen yang menghubungkan bagian-bagian dari flowchart yang terletak pada satu halaman.
	<i>Off page connector</i> atau penyambung antar halaman Elemen penghubung antara bagian-bagian flowchart yang terletak pada lembaran yang berbeda.

J. Unified Modeling Language (UML)

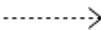
Dharwiyanti, S. dan Wahono, R.S. pada tahun 2003 menyatakan bahwa *Unified Modelling Language* (UML) merupakan standar industri dalam hal visualisasi, perancangan, dan dokumentasi sistem perangkat lunak. UML menyediakan kerangka kerja untuk desain model sistem. Beberapa aspek berbeda dari sistem perangkat lunak, seperti struktur kelas, interaksi antar objek, dan alur kontrol, dapat diwakili dalam UML.

Penggunaan UML memungkinkan para pengembang perangkat lunak untuk lebih mudah memahami dan membuat sistem yang lebih kompleks. UML memungkinkan pengembang untuk merepresentasikan sistem dalam bentuk visual yang mudah dipahami, sehingga dapat membantu dalam mengkomunikasikan desain sistem dengan anggota tim pengembang dan stakeholder. Selain itu, UML juga memungkinkan pengembang untuk mengidentifikasi kesalahan desain pada tahap awal pengembangan sehingga dapat diminimalkan risiko kesalahan pada tahap implementasi.

Diagram *Use Case* menggambarkan interaksi antara *Use Case* dan aktor. Aktor dapat berupa individu, perangkat, atau sistem lain yang berinteraksi dengan sistem. Fungsionalitas sistem atau kebutuhan yang harus dipenuhi oleh pengguna dijelaskan dalam suatu *Use Case* (Nurokhim dan Ratnasari Nur Rohmah. 2002).

Adapun simbol-simbol *Use Case* Diagram antara lain :

Tabel 2.4 Simbol *use case diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menentukan peran penting yang dimainkan oleh pengguna saat berinteraksi dengan kasus penggunaan..
2		<i>Dependency</i>	Kaitan di mana perubahan yang terjadi pada suatu elemen independen akan berdampak pada elemen yang bergantung padanya. elemen yang tidak

NO	GAMBAR	NAMA	KETERANGAN
			mandiri (<i>independent</i>).
3		<i>Generalization</i>	Ikatan di mana objek turunan (descendant) mewarisi perilaku dan struktur data dari objek yang berada di atasnya, yaitu objek induk (ancestor).
4		<i>Include</i>	Menyatakan secara eksplisit use case sumber.
5		<i>Extend</i>	Menentukan bahwa use case target memperluas perilaku dari use case sumber pada titik tertentu.
6		<i>Association</i>	Apa yang menghubungkan satu objek dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem dengan batasan tertentu.
8		<i>Use Case</i>	Penjelasan dari rangkaian tindakan yang dilakukan oleh sistem untuk mencapai hasil yang

NO	GAMBAR	NAMA	KETERANGAN
			dapat diukur bagi seorang aktor.
9		<i>Collaboration</i>	Interaksi antara aturan-aturan dan elemen lainnya saling berkolaborasi untuk menghasilkan perilaku yang lebih kompleks daripada sekadar jumlah individu dan elemennya (sinergi).
10		<i>Note</i>	Faktor fisik yang hadir saat aplikasi berjalan dan mencerminkan suatu sumber daya komputasi.

K. *White box* dan *Black box*

(Setian, Roni. 2021). Pengujian kotak putih, juga dikenal sebagai "*White Box Testing*," merupakan jenis uji coba yang menguji perangkat lunak dengan menganalisis struktur internal dan kode-kodenya. Uji coba *White Box* berfokus pada data masukan dan keluaran perangkat lunak, sementara uji coba *Black Box* hanya melihat hasil akhir.

Latihan dalam kotak putih, yang juga dikenal sebagai latihan struktural atau latihan dalam kotak terbuka, adalah metode pengujian perangkat lunak di mana dilakukan analisis mendalam terhadap struktur di dalam suatu aplikasi. Dalam

whitebox testing, para pengujian memiliki akses penuh ke kode sumber perangkat lunak dan menggunakan pengetahuan tersebut untuk merancang dan melaksanakan skenario pengujian. Inti dari pengujian *whitebox* adalah memastikan bahwa setiap bagian dari kode beroperasi sesuai dengan spesifikasi dan memastikan bahwa semua jalur eksekusi telah diuji secara menyeluruh.

Jenis-jenis *whitebox* testing melibatkan berbagai tingkat kompleksitas dalam menganalisis kode sumber. Salah satu jenis yang umum adalah *Unit Testing*, di mana setiap unit atau modul kecil dari program diuji secara terpisah untuk memastikan fungsionalitasnya. *Integration Testing* melibatkan pengujian keseluruhan sistem setelah semua unit telah diintegrasikan. Sedangkan *System Testing* lebih berfokus pada pengujian keseluruhan aplikasi untuk memastikan bahwa seluruh sistem berfungsi dengan baik. Terakhir, ada *Acceptance Testing*, yang mencakup pengujian untuk memastikan bahwa sistem memenuhi persyaratan pengguna dan spesifikasi yang telah ditetapkan.

Whitebox testing memberikan keuntungan dalam mengidentifikasi *bug* dan kelemahan di tingkat kode, memastikan bahwa aplikasi beroperasi dengan baik, dan meningkatkan keandalan serta keamanan perangkat lunak. Namun, kekurangan utamanya adalah keterbatasan dalam pengujian keberfungsian keseluruhan sistem yang dapat diabaikan jika tidak diimbangi dengan jenis pengujian lainnya seperti *blackbox testing*.

(Setian, Roni. 2021). *Black box* testing atau yang juga dikenal sebagai *Behavioral Testing* adalah jenis pengujian yang dilakukan untuk mengamati hasil

input dan output dari perangkat lunak tanpa harus mengetahui struktur kode sumber perangkat lunak tersebut. Pengujian ini umumnya dilaksanakan pada tahap akhir pengembangan perangkat lunak untuk menentukan apakah perangkat lunak beroperasi dengan baik.

Tujuan utama dari *blackbox testing* adalah untuk mengevaluasi fungsi-fungsi eksternal suatu aplikasi dan memastikan bahwa sistem berperilaku sesuai dengan spesifikasi yang telah ditetapkan.

Contoh *blackbox testing* melibatkan pengujian pada berbagai tingkat, seperti *Unit Testing*, *Integration Testing*, *System Testing*, dan *Acceptance Testing*. Sebagai contoh, dalam *Unit Testing*, pengujian dapat dilakukan untuk memeriksa apakah fungsi-fungsi individu atau modul-modul kecil bekerja sesuai dengan harapan tanpa memperhatikan detail implementasi internalnya. Pada tingkat *Integration Testing*, fokusnya adalah pada interaksi antar unit atau modul, dan pengujian dilakukan untuk memastikan bahwa integrasi tersebut berjalan dengan baik. Dalam *System Testing*, seluruh aplikasi diuji secara keseluruhan untuk memastikan bahwa semua komponen bekerja bersama dengan baik. Sementara *Acceptance Testing* menilai apakah sistem telah memenuhi persyaratan pengguna dan spesifikasi yang telah ditetapkan.

Keuntungan dari *blackbox testing* melibatkan pemisahan antara pengembang dan tim pengujian, sehingga pengujian dapat dilakukan dengan sudut pandang yang independen. Selain itu, *blackbox testing* memungkinkan identifikasi *bug* atau kelemahan fungsionalitas aplikasi tanpa memerlukan pengetahuan

mendalam tentang implementasi internalnya. Namun, kekurangan utama adalah bahwa *blackbox testing* mungkin tidak efisien dalam menemukan bug atau masalah yang terkait dengan logika internal aplikasi. Oleh karena itu, kombinasi antara *blackbox* dan *whitebox* testing seringkali diterapkan untuk mencapai pengujian yang lebih komprehensif.

L. Kerangka Pikir

Keamanan kandang ternak merupakan hal yang penting dalam menjaga keberlangsungan peternakan. Namun, seringkali pengawasan secara langsung oleh manusia memiliki kelemahan karena keterbatasan waktu dan tenaga. Oleh karena itu, diperlukan sebuah solusi yang efisien dan otomatis dalam mengawasi kandang agar dapat mendeteksi adanya gangguan atau ancaman dari pencuri.



Untuk mengatasi masalah ini peneliti akan mengembangkan sebuah aplikasi pengaman kandang berbasis Android yang terintegrasi dengan *Orange Pi*, sensor PIR, dan webcam. Aplikasi ini akan berfungsi untuk mendeteksi gerakan di sekitar kandang melalui sensor PIR dan mengambil gambar menggunakan webcam.



Untuk membangun Aplikasi yang dapat berkomunikasi dengan *Orange Pi* melalui koneksi jaringan internet untuk mengontrol dan memantau sensor PIR dan *webcam*. Sensor PIR akan digunakan untuk mendeteksi pergerakan di sekitar kandang, dan webcam akan digunakan untuk mengambil gambar saat ada gerakan.



Menghasilkan sebuah prototype berbasis orange pi dan webcam yang dapat mendeteksi menggunakan sensor PIR dan sebuah aplikasi berbasis android.

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Dalam penelitian ini, metode yang akan diterapkan adalah metode eksperimen. Peneliti akan merancang sebuah aplikasi pengaman kandang yang berbasis Android. Setelah itu, akan dilakukan pengujian dan evaluasi terhadap performa aplikasi pengaman tersebut.

B. Lokasi dan Waktu Penelitian

1. Lokasi

Penelitian dilaksanakan di wilayah desa Pangaparan, kecamatan Lembang, Kabupaten Pinrang, dengan durasi pelaksanaan selama kurang lebih satu bulan.

2. Waktu

Penelitian ini dilaksanakan selama satu bulan.

C. Alat dan Bahan Penelitian

Dalam menjalankan proses penelitian untuk pembuatan aplikasi, perangkat keras dan perangkat lunak menjadi suatu kebutuhan esensial untuk mendukung kegiatan penelitian. Berikut merupakan perangkat keras dan perangkat lunak yang digunakan.

1. Perangkat keras

Perangkat keras yang dipakai untuk pembuatan aplikasi dan unggah kode ke mikrokontroler dapat ditemukan dalam tabel berikut :

Tabel 3.1 Spesifikasi perangkat keras

No	Spesifikasi	
1	Merk laptop	DELL Latitude E6440
2	Processor laptop	Intel(R) Core(TM) i5-4300M CPU @ 2.60 GHz (4CPUs), ~2.6GHz
3	RAM laptop	8 (8192) GB
4	Mikrokontroller	OrangePi 3B (4Gb RAM)
5	Komponen tambahan	Sensor PIR HC-SR 501, <i>USB Webcam</i> , kabel jumper, kabel listrik, kabel usb tipe c, <i>HDMI to VGA converter</i> .

2. Perangkat lunak

Perangkat lunak yang digunakan untuk membuat aplikasi *encode* dan *decode* gambar adalah sebagai berikut :

Tabel 3.2 Spesifikasi perangkat lunak

No	Spesifikasi	
1	Sistem operasi	Windows 10, <i>Thirdparty RaspOs</i>
2	Tool pemrograman	<i>Vs Code, Android Studio, Fritzing</i>
3	Bahasa pemrograman	<i>Python, Kotlin, html, css, javascript.</i>

D. Metode Pengumpulan Data

Penelitian ini akan difokuskan pada perancangan dan pembuatan prototipe alat pengaman kandang menggunakan sensor gerak PIR. Selain itu, pengujian dan evaluasi akan dilakukan di lingkungan kandang. Penelitian ini tidak akan membahas pengaruh alat pendeteksi terhadap lingkungan secara keseluruhan.

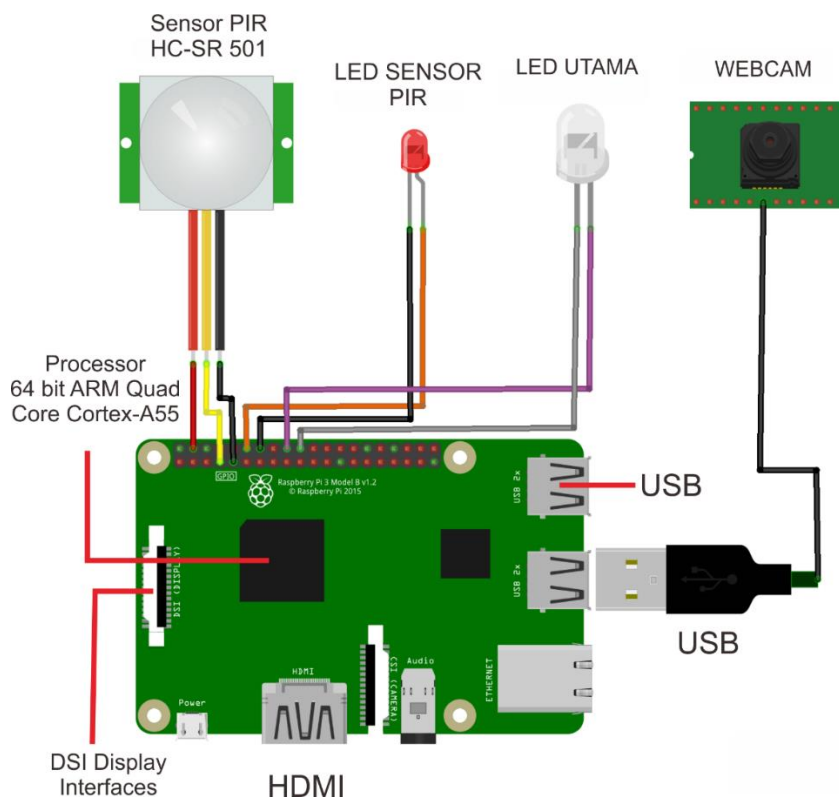
BAB IV

HASIL DAN PEMBAHASAN

A. Implementasi Sistem

Pada dasarnya hasil penelitian ini menghasilkan 2 unsur utama yaitu perangkat keras dan perangkat lunak.

1. Perangkat Keras (Hardware)



Gambar 4. 1. Diagram pengkabelan

Tabel 4.1 Pengoneksian pin *NodeMCU Esp8266*

No	OrangePi Pin	Keterangan
1	7	Dihubungkan Output pin PIR
2	GND	Dihubungkan ke GND pin PIR, Katoda main LED, katoda PIR LED
3	4	Dihubungkan ke VCC pin PIR
4	12	Dihubungkan ke anoda PIR LED
5	18	Dihubungkan ke anoda main LED

2. Perangkat Lunak (Software)

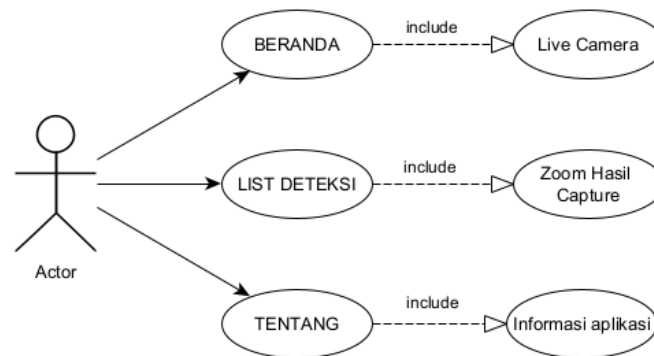
Interface yang dibuat adalah sebagai berikut



Gambar 4. 2. Tampilan aplikasi

B. Pengujian Sistem

1. Use Case Diagram



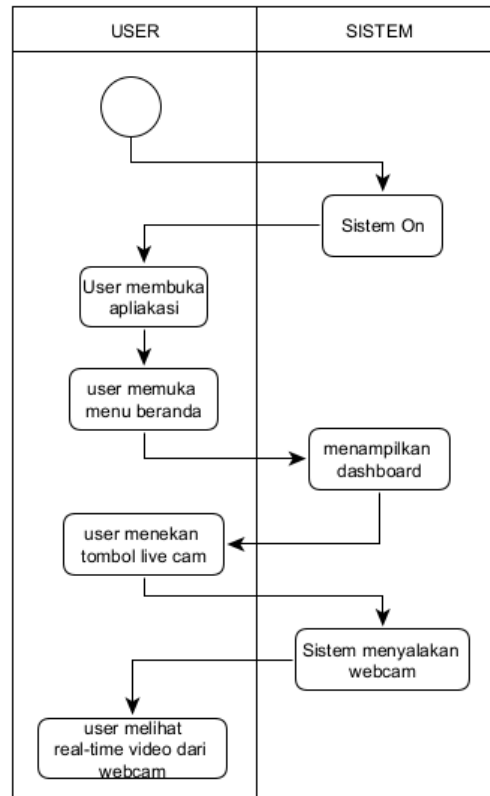
Gambar 4. 3. Use Case diagram

Tabel 4. 1. Deskripsi Use Case

Nama Use Case	Deskripsi Use Case
Menu Beranda	Merupakan halaman yang menampilkan beranda dan fungsi selanjutnya seperti live cam
Menu List Deteksi	Merupakan halaman yang menampilkan list hasil tangkapan kamera
Menu Tentang	Merupakan halaman yang menampilkan deskripsi informasi aplikasi
Fungsi Live Streaming	Merupakan fungsi untuk melakukan live streaming dari webcam
fungsi zoom	Merupakan fungsi untuk melakukan zoom pada hasil tangkapan kamera yang diklik
informasi aplikasi	Deskripsi dari aplikasi

2. Activity Diagram

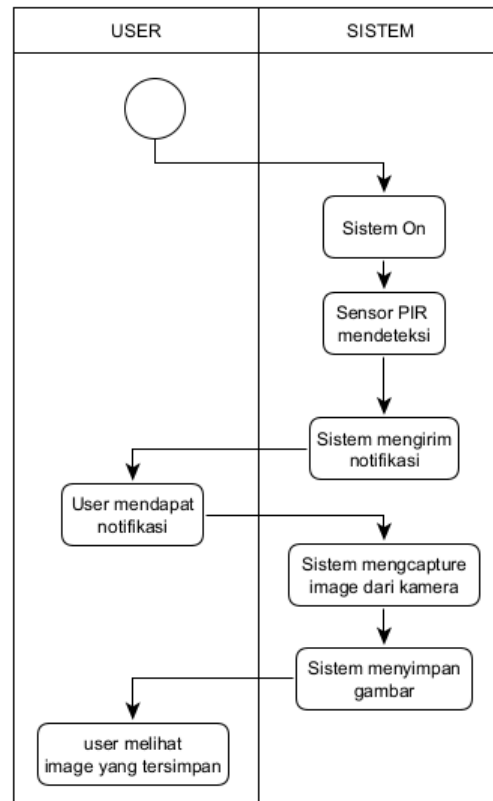
a. Live cam



Gambar 4. 4. Activity diagram *live cam*

User membuka aplikasi kemudian sistem berjalan, ketika user masuk ke menu beranda maka aplikasi menampilkan sebuah dashboard yang di dalam nya terdapat jumlah hasil tangkapan kamera dan tombol live cam. Ketika tombol live cam ditekan maka sistem akan menampilkan sebuah gambar yang diambil dari hasil tangkapan kamera seraca langsung (*live*), yang berguna untuk memonitoring keadaan secara *real-time*.

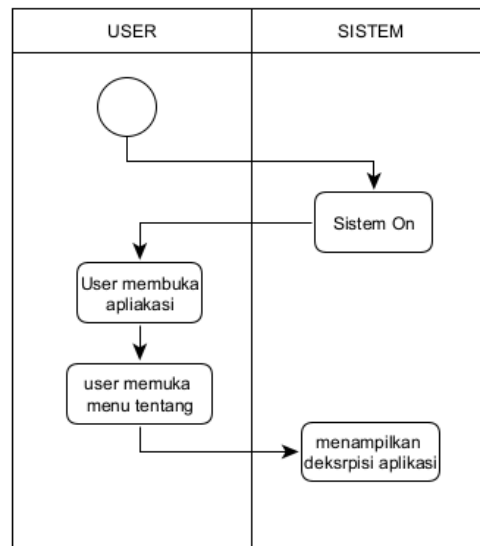
b. *Capture image*



Gambar 4. 5. *Activity diagram capture image*

Mulanya user menyalakan alat yang kemudian sistem akan menyalakan sensor PIR dan membuat sistem ini aktif untuk mendeteksi, ketika sensor menerima input (hasil deteksi) sistem akan mengirimkan sebuah notifikasi ke aplikasi yang dapat dilihat oleh *user*. Bersamaan dengan notifikasi, sistem juga *capture* dan menyimpan hasil jepretan kamerake dalam penyimpanan internal yang kemudian hasil *capture* ini dapat dilihat oleh user melalui aplikasi android.

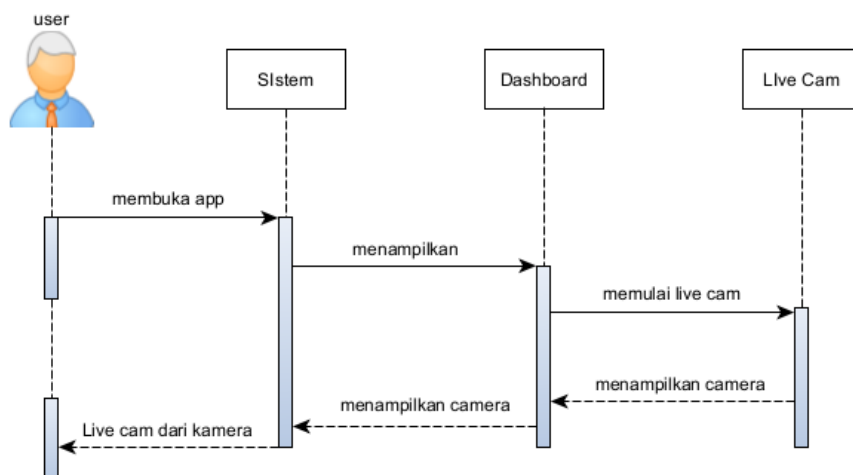
c. Menu Tentang



Gambar 4. 6. Activity diagram menu tenntang

3. Sequence Diagram

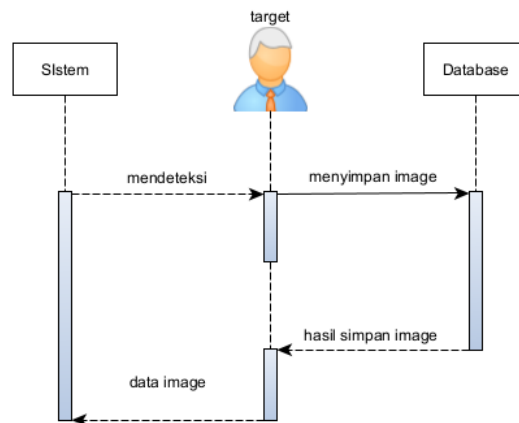
a. Sequence Live Cam



Gambar 4. 7. Sequence diagram live cam

Pada sequence live cam, user membuka aplikasi, sistem menampilkan dashboard dan di dalam dashboard terdapat tombol live cam, ketika tombol live cam ditekan maka sistem melakukan siaran langsung dari kamera

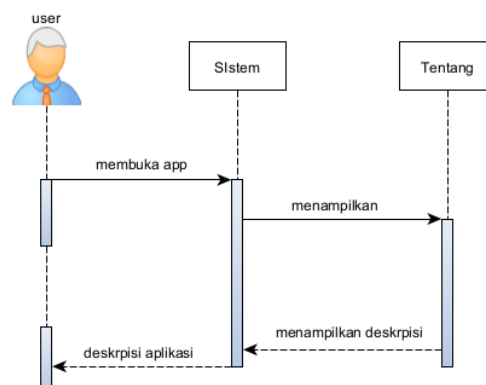
b. Sequence Capture Image



Gambar 4. 8. *Sequence diagram capture image*

Pada sequence diagram capture image, sistem menyala dan melakukan deteksi pada target ketika target mencapai jarak deteksi sensor, ketika target terdeteksi maka sistem melakukan capture image dan menyimpan ke dalam media penyimpanan internal alat. Data tersebut kemudian dapat ditampilkan melalui aplikasi android.

c. Sequence Menu Tentang



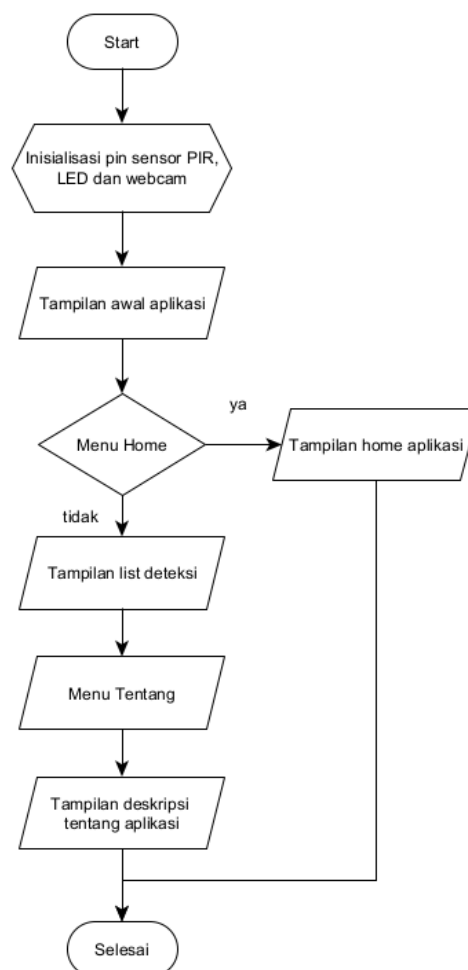
Gambar 4. 9. *Sequence diagram menu tentang*

4. Pengujian Whitebox

Pengujian *whitebox* dilakukan untuk melihat hasil kontrol yang dilakukan *NodeMCU Esp8266* berdasarkan perhitungan dan logika program untuk mengetahui efektifitas sistem yang telah dibangun.

a. Whitebox Aplikasi

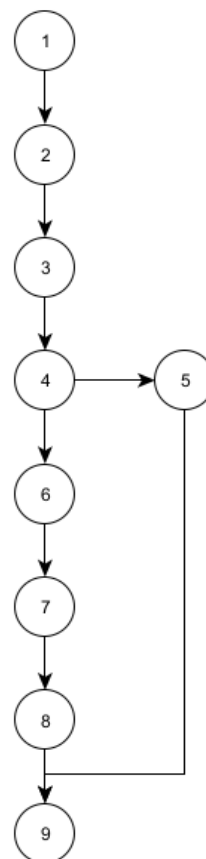
Pengujian *whitebox* pada aplikasi dilakukan untuk menguji sistem yang dibangun menggunakan metode alur seperti *activity diagram*, *flowchart* dan *flowgraph*.



Gambar 4. 10. Flowchart aplikasi

Pada flowchart diatas, ketika aplikasi dijalankan aplikasi menampilkan tampilan awal yang terdiri dari menu *home*, menu *list* deteksi dan menu tentang.

1) *Flowgraph* aplikasi



Gambar 4. 11. Flowgraph aplikasi

2) *Cyclomatic Complexity*

Menghitung *cyclomatic complexity* $V(G)$ dari *Edge* dan *Node*:

Menggunakan rumus $V(G) = E - N + 2$

E (*Edge*) = 9

N (*Node*) = 9

Penyelesaian: $V(G) = E - N + 2$
 $= 9 - 9 + 2$
 $= 2$

3) Berdasarkan perhitungan *cyclomatic complexity* dari *flowgraph* di atas mempunyai *region* = 2

4) *Independentpath* pada *flowgraph* di atas adalah:

Path 1 = 1 – 2 – 3 – 4 – 5 – 9

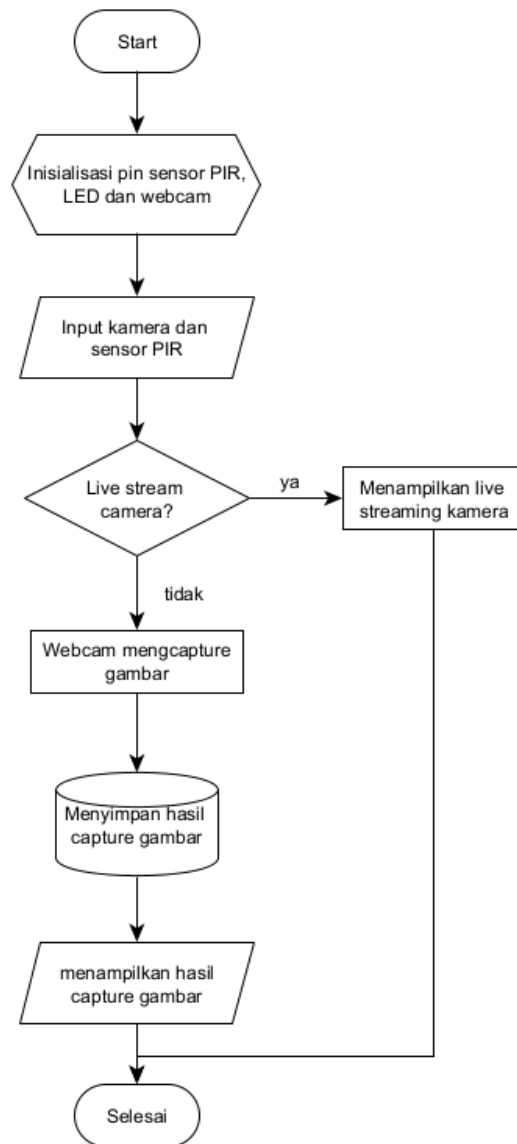
Path 2 = 1 – 2 – 3 – 4 – 6 – 7 – 8 – 9

5) Grafik matriks aktivasi *flowgraph*

Tabel 4.2 grafik matriks aktivasi *flowgraph* aplikasi

[illegible]

b. *Whitebox Alat*

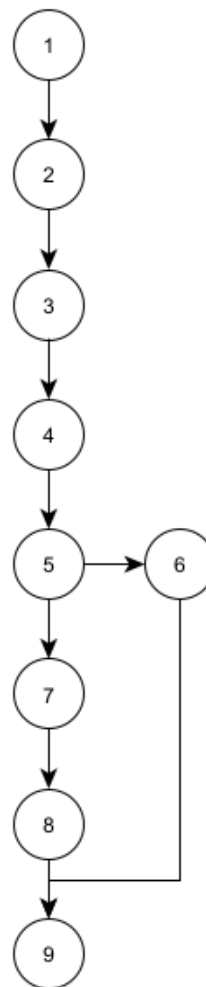


Gambar 4. 12. *Flowchart Alat*

Pada flowchart diatas, langkah pertama alat tersebut adalah OrangePi menginisialisasi pinout yang akan digunakan dalam mengatur sensor PIR, sebuah led dan sebuah webcam yang kemudian OrangePi menjalankan server localhost aplikasi.

Setelah inisialisasi selesai selanjutnya orangepi mengambil nilai dari sensor pir, apakah ada gerakan yang terdeteksi oleh sensor pir tersebut. Jika sensor mendeteksi gerakan maka webcam akan melakukan *auto capture* gambar dan menyimpannya pada media penyimpanan orangepi. Ketika user menggunakan aplikasi android dan menyalakan *live stream* pada aplikasi, maka *auto capture* gambar akan ditunda hingga pemakaian *resource webcam* pada *live stream* telah selesai.

1) *Flowgraph* alat



Gambar 4. 13. *Flowgraph* alat

2) *Cyclomatic Complexity*

Menghitung cyclomatic complexity $V(G)$ dari *Edge* dan *Node*:

Menggunakan rumus $V(G) = E - N + 2$

$$E (\text{Edge}) = 9$$

$$N (\text{Node}) = 9$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 9 - 9 + 2$$

$$= 2$$

3) Berdasarkan perhitungan *cyclomatic complexity* dari *flowgraph* di atas mempunyai *region* = 2

4) *Independent path* pada *flowgraph* di atas adalah:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 5 - 7 - 8 - 9$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 9$$

5) Grafik matriks aktivasi *flowgraph*

	1	2	3	4	5	6	7	8	9	E - 1
1		1								$1 - 1 = 0$
2			1							$1 - 1 = 0$
3				1						$1 - 1 = 0$
4					1					$1 - 1 = 0$
5						1	1			$2 - 1 = 1$
6									1	$1 - 1 = 0$
7								1		$1 - 1 = 0$
8									1	$1 - 1 = 0$

9										0
Sum (E) + 1										1 + 1 = 2

5. Pengujian *Blackbox*

a. *Blackbox* Aplikasi

Pengujian *blackbox* aplikasi bertujuan untuk mengetahui berfungsi atau tidaknya aplikasi yang telah dibuat. Pengujian dilakukan untuk melihat hasil deteksi yang dilakukan alat berdasarkan intruksi sensor yang digunakan pada sisi aplikasi.

1) Detail aplikasi

Halaman utama adalah tampilan aplikasi saat aplikasi dijalankan.

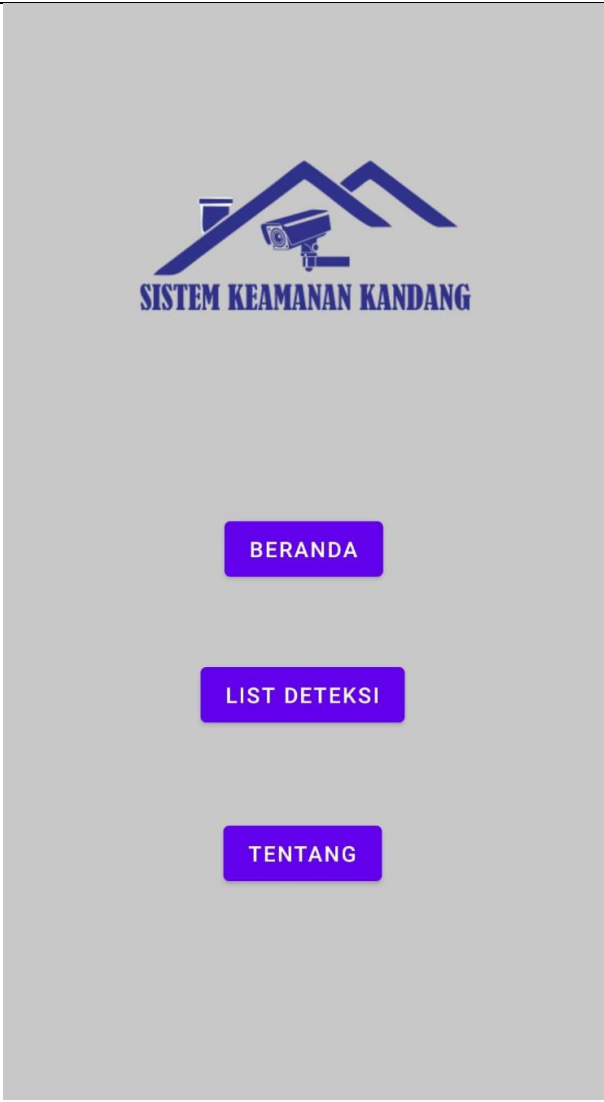


Gambar 4. 14. Halaman awal aplikasi

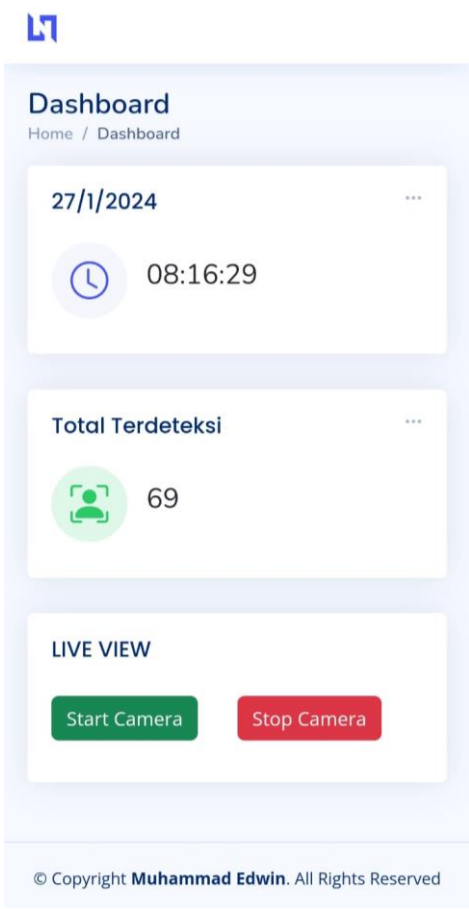
2) Pengujian aplikasi

Pengujian aplikasi dilakukan untuk mengetahui jalannya aplikasi,

Tabel 4.3 *Blackbox* tampilan awal

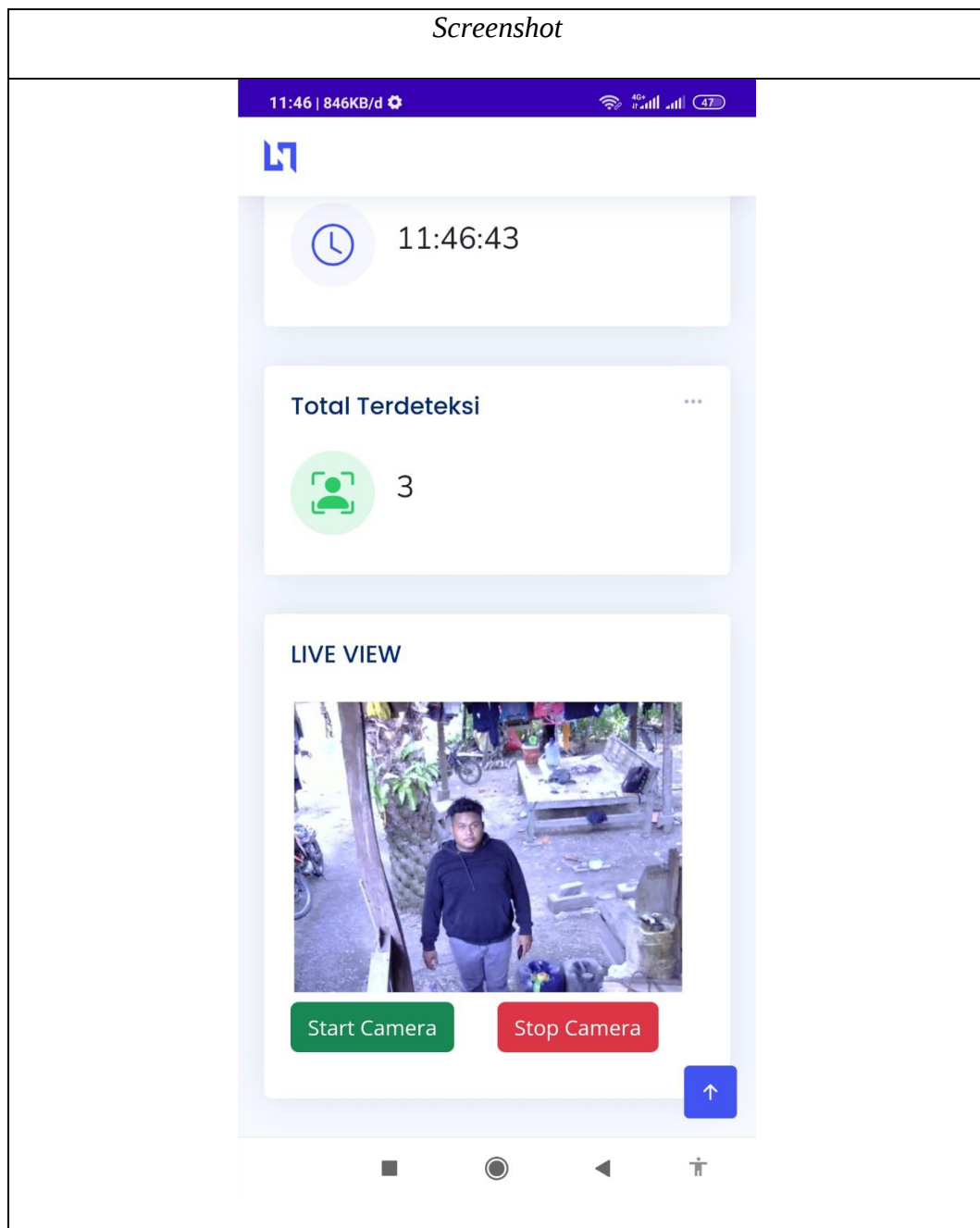
Text Faktor	Hasil	Kesimpulan
Tampilan awal aplikasi	✓	Berhasil, berhasil menampilkan tampilan awal aplikasi
<i>Screenshot</i>		
		

Tabel 4.4 *Blackbox* tampilan home

Text Faktor	Hasil	Kesimpulan
Tampilan home aplikasi	✓	Berhasil, berhasil menampilkan tampilan <i>home</i> aplikasi
<i>Screenshot</i>		
		

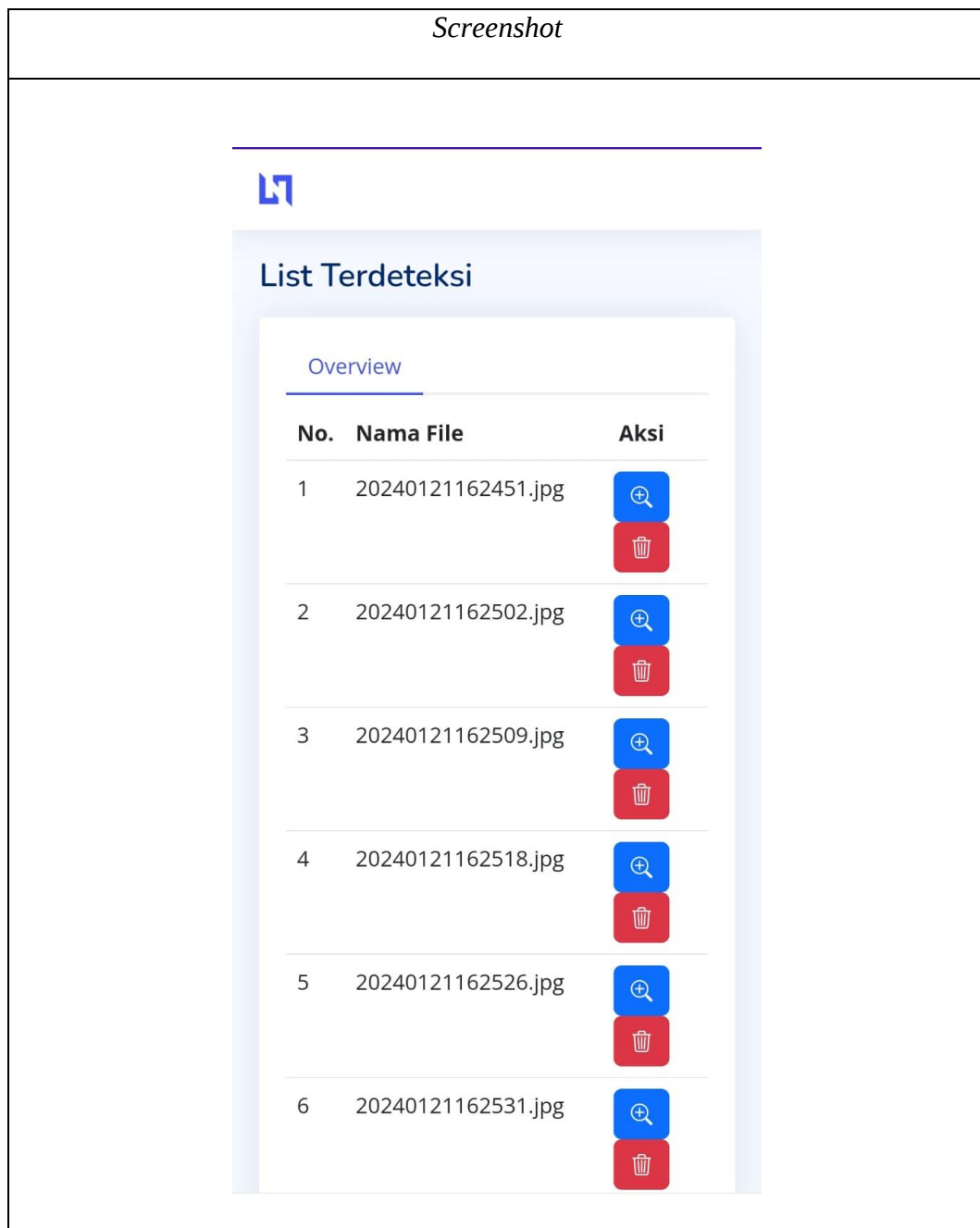
Tabel 4.5 *Blackbox* tampilan live stream

Text Faktor	Hasil	Kesimpulan
Tampilan <i>live cam</i>	✓	Berhasil, berhasil menampilkan livecam dari <i>webcam</i> pada alat



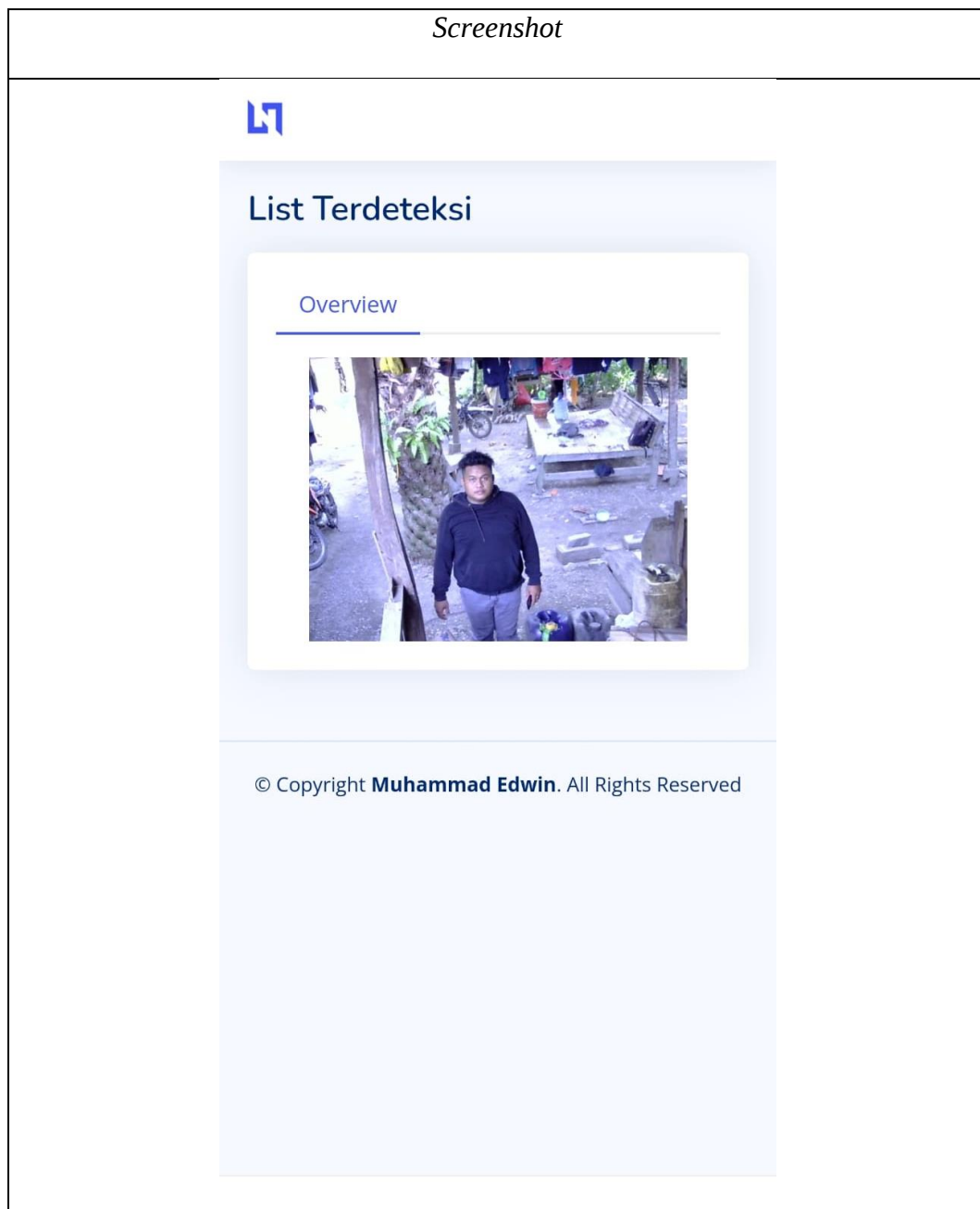
Tabel 4.6 *Blackbox* tampilan menu *list* deteksi

Text Faktor	Hasil	Kesimpulan
Tampilan menu <i>list</i> deteksi	✓	Berhasil, berhasil menampilkan list hasil deteksi alat



Tabel 4.7 *Blackbox* tampilan zoom image

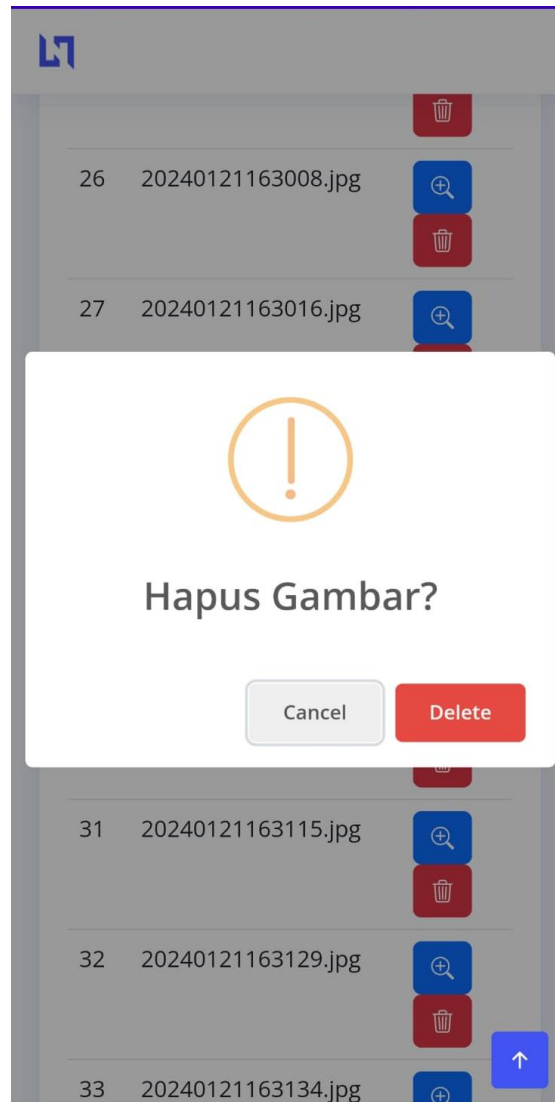
Text Faktor	Hasil	Kesimpulan
Tampilan zoom image	✓	Berhasil, berhasil menampilkan zoom image yang dipilih pada list hasil deteksi alat



Tabel 4.8 *Blackbox* tampilan hapus *image*

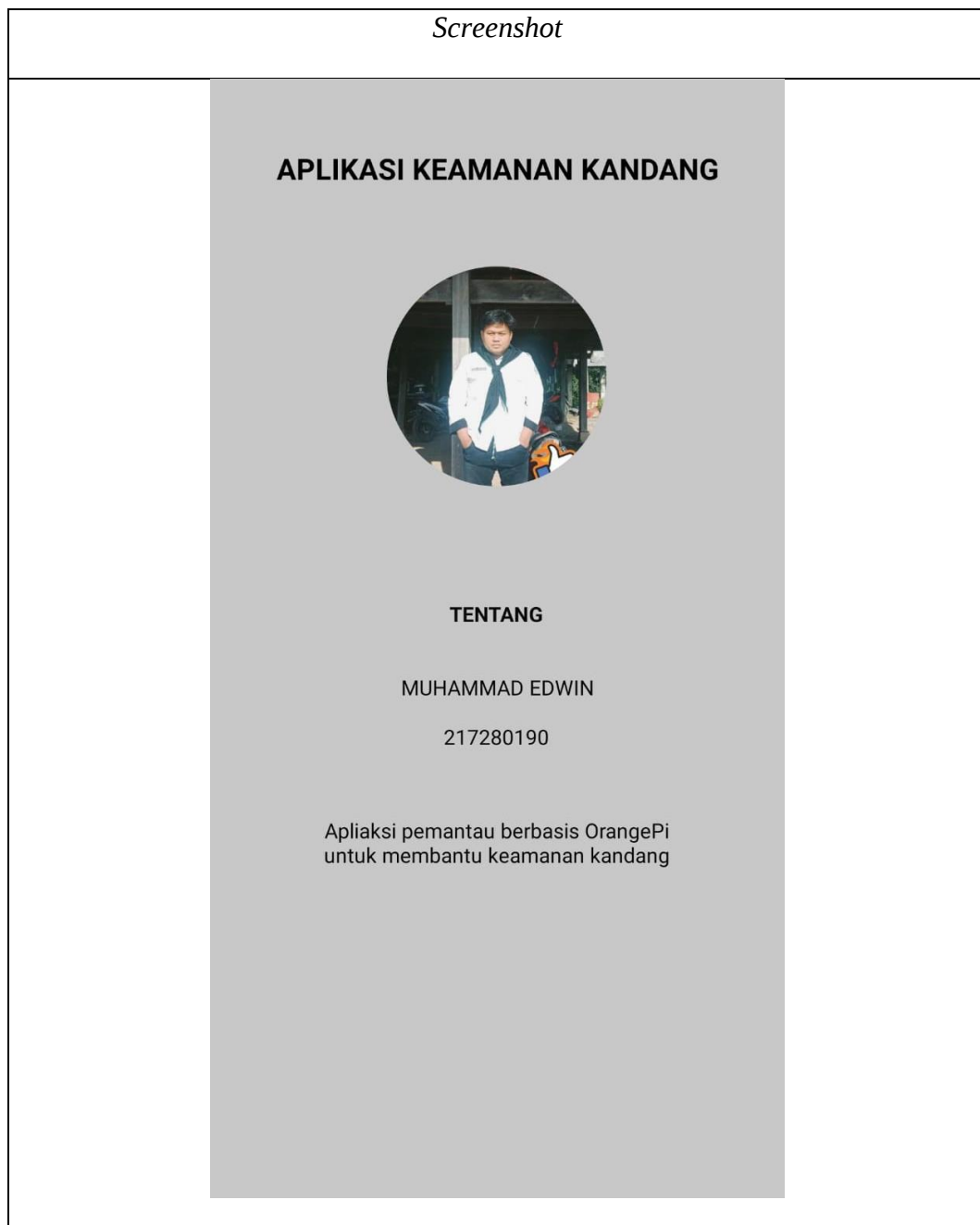
Text Faktor	Hasil	Kesimpulan
Tampilan hapus <i>image</i>	✓	Berhasil, berhasil menampilkan hapus <i>image</i> yang dipilih pada list hasil deteksi alat

Screenshot



Tabel 4.9 Blackbox tampilan menu tentang

Text Faktor	Hasil	Kesimpulan
Tampilan menu tentnag	✓	Berhasil, berhasil menampilkan halaman tentang aplikasi



Aplikasi sistem keamanan ini dibuat menggunakan bahasa pemrograman kotlin dan *javascript* dari sisi webnya, aplikasi ini memerlukan koneksi internet dalam menghubungkan sisi alat dan sisi aplikasi. Berdasarkan pengujian diatas, aplikasi ini berhasil dijalankan sesuai sintaks dan logika programnya.

b. *Blackbox* Alat

Pengujian blackbox alat bertujuan untuk mengetahui berfungsi atau tidaknya alat yang telah dibuat. Pengujian dilakukan untuk melihat hasil deteksi yang dilakukan alat berdasarkan intruksi sensor yang digunakan.

1) Detail alat



Gambar 4. 15. Alat sistem keamanan kandang

2) Pengujian alat

Pengujian alat dilakukan untuk mengetahui fungsi mikrokontroller, webcam dan jarak sensor pir dalam mendeteksi.

Tabel 4.10 Pengujian jarak sensor PIR

No	Jarak (cm)	Respon Sensor PIR	Keterangan
1	60	terdeteksi	berhasil
Hasil capture			

No	Jarak (cm)	Respon Sensor PIR	Keterangan
			
2	90	terdeteksi	berhasil
Hasil capture			
			
3	120	terdeteksi	berhasil
Hasil capture			
			
4	150	terdeteksi	berhasil
Hasil capture			

No	Jarak (cm)	Respon Sensor PIR	Keterangan
5	180	terdeteksi	berhasil
Hasil capture			
6	210	terdeteksi	berhasil
Hasil caputre			
7	250	terdeteksi	berhasil dengan akurasi rendah
Hasil capture			

No	Jarak (cm)	Respon Sensor PIR	Keterangan
8	270	terdeteksi	Berhasil dengan akurasi rendah
Hasil capture			
			
9	300	tidak terdeteksi	tidak berhasil
Hasil capture			
			

Berdasarkan pengujian diatas, alat berhasil mendeteksi dan *mengcapture* gambar melalui webcam dengan jarak maksimal 270cm.



Gambar 4. 16. Alat dalam posisi *stanby* dan posisi *capture image*

BAB V

PENUTUP

A. Kesimpulan

Penelitian ini menghasilkan 2 *output* yaitu sebuah software dan hardware. Software berupa aplikasi android yang digunakan untuk memantau hasil deteksi dari *hardware* yang telah dibuat, sedangkan *hardware* berupa alat menggunakan mikrokontroller *Orange Pi 3b* yang dikombinasikan dengan sensor PIR (*passive infrared receiver*) HC-SR501 dan sebuah *webcam* yang berhasil dirangkai dan berjalan sesuai sintaks program. Berdasarkan hasil pengujian alat menggunakan sensor PIR HC-501, alat ini berhasil mendeteksi manusia dengan jangkauan jarak sejauh 270cm. semakin dekat objek deteksi dengan alat maka semakin akurat hasil deteksi yang dilakukan. Berdasarkan hasil pengujian aplikasi, aplikasi tersebut berhasil menampilkan gambar hasil dari jepretan yang diambil melalui alat.

B. Saran

saran untuk pengembangan dalam penelitian ini sebagai berikut:

1. Alat ini memerlukan koneksi internet untuk melakukan live stream melalui aplikasi android, peneliti menggunakan perangkat Mi-Fi (modem *WiFi mobile*) dengan kecepatan yang terbatas dalam transfer data, sehingga menyebabkan delay dalam *livestream* pada aplikasi. Diharapkan kedepannya untuk mengembangkan penelitian ini agar menggunakan

perangkat jaringan yang lebih cepat dan stabil agar mengurangi delay pada livestream kamera.

2. Peneliti menggunakan sensor PIR HC-SR501 sebagai sensor yang mendeteksi gerakan, tentu saja sensor ini memiliki batasan dalam jangkauan deteksinya yang dalam penelitian kali ini dapat menjangkau 270cm dengan hasil deteksi yang kurang akurat. Untuk kedepannya peneliti berharap penggunaan sensor dengan jarak jangkauan yang melebihi sensor yang kami gunakan saat ini, demi meningkatkan akurasi dan jarak deteksi yang lebih baik.
3. Dalam membangun aplikasi berbasis android ini, peneliti menggunakan metode *native* dan *webview*. Untuk pengembangan sistem ini peneliti berharap penggunaan *native* dengan *clean code* yang lebih baik demi meningkatkan kecepatan aplikasi.

DAFTAR PUSTAKA

- Arduino. (2018, februari 05). *Arduino What is Arduino*. Diambil kembali dari Arduino: <https://www.arduino.cc/en/Guide/Introduction>
- Bagye, W., & Azizah, T. (2018). Alat Pengaman Kandang Berbasis Mikrokontroler Arduino Uno. *JIRE (Jurnal Informatika & Rekayasa Elektronika) Volume 1, No 2*.
- Chidera, E. I. (2023, Maret 31). *Kotlin Vs Java - Whats the different?* Diambil kembali dari Freecodecamp: <https://www.freecodecamp.org/news/kotlin-vs-java-whats-the-difference/#:~:text=One%20of%20the%20most%20significant,to%20perform%20the%20same%20operations.&text=As%20you%20can%20see%2C%20the,readable%20than%20the%20Java%20version.>
- Darwin, Gadjah, U., Simarmata, E. M., & Laia, Y. (2019). Aplikasi Penyimpanan File Alternatif Bagi Pengguna Smartphone Berbasis Android. *Jurnal Sistem Informasi dan Ilmu Komputer Prima (JUSIKOM PRIMA) Vol. 3 No. 1*, 35-36.
- Dennis, A., Wixom., B. H., & Tegarden, D. (2014). *System Analysis and Design: An Object-Oriented Approach with UML*. Wiley. wiley.
- Dharwiyanti, S., & Wahono, R. S. (2003). Pengantar Unified Modelling Language (UML). *IlmuKomputer.Com*.
- Galliers, R. D., & Currie, W. L. (2017). *Information Systems Development: Reflections, Challenges and New Directions*. New York: Spriger.
- Pradipta, G. M., Nabilah, N., Islam, H. I., Saputra, D. H., Said, S., Kurniawan, A., . . . Irzaman. (2016). Pembuatan Prototipe Sistem Keamanan Laboratorium Berbasis Arduino Mega. *Prosiding Seminar Nasional Fisika (E-Journal)* .
- Rahman, S., Sembiring, A., Siregar, D., Khair, H., Prahmana, I. G., Puspadini, R., & Zen, M. (2023). *Python: Dasar Dan Pemrograman Berorientasi Objek*. Sokoharjo: Tahta Media Group.
- Rony Setiawan. (2021, November 17). *Black Box Testing Untuk Menguji Perangkat Lunak*. Diambil kembali dari Dicoding: <https://www.dicoding.com/blog/black-box-testing/> - Black Box Testing Untuk Menguji Perangkat Lunak
- Setiawan, Rony. (2021, November 20). *White Box Testing untuk Menguji Perangkat Lunak*. Diambil kembali dari Dicoding:

<https://www.dicoding.com/blog/white-box-testing/> - White Box Testing untuk Menguji Perangkat Lunak

Shenzhen Xunlong Software Co., Ltd. (2023). *Orange Pi User Manual*. Shenzhen: Shenzhen Xunlong Software Co., Ltd.

Theengineeringprojects:introduction to hc-sr501. (2019, januari 30). Diambil kembali dari theengineeringprojects: <https://www.theengineeringprojects.com/2019/01/introduction-to-hc-sr501.html>

Toyib, R., Bustami, I., Abdullah, D., & Onsardi. (2019). Penggunaan Sensor Passive Infrared Receiver (Pir) Untuk Mendeteksi Gerak Berbasis Short Message Service Gateway. *Jurnal Pseudocode, Volume VI Nomor 2*.

Universitas Muhammadiyah Parepare. (2021). *Panduan Penulisan Proposal Dan Skripsi*. Parepare: Fakultas Teknik Universitas Muhammadiyah Parepare.