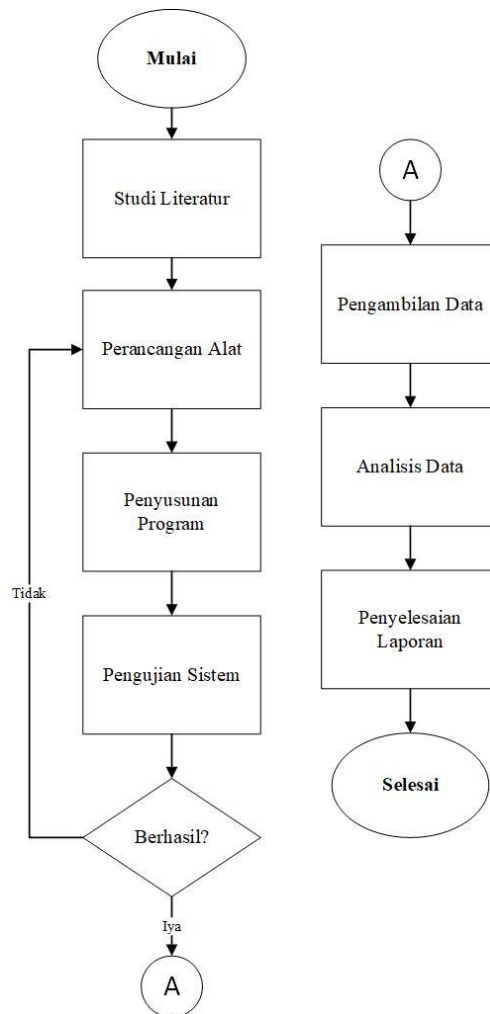


### BAB III

## METODE PENELITIAN

Pada bab ini, akan menjelaskan mengenai bagaimana merancang alat sistem *monitoring* kendaraan guna pengecekan bawah kendaraan berbasis IoT. Dalam membuat alat ini dibagi menjadi dua kategori perangkat berupa perangkat keras dan perangkat lunak. Perangkat keras yang digunakan antara lain ESP32-CAM, sensor ultrasonik HCSR04, transistor NPN, resistor 1k Ohm, lampu LED 5V. Sedangkan, untuk perangkat lunak yang digunakan antara lain *software* Arduino IDE, Python, Telegram, dan Wireshark. Untuk *flowchart* alur penelitian seperti terlihat pada Gambar 3.1 :



**Gambar 3.1** *Flowchart* Alur Penelitian

Penjelasan mengenai *flowchart* alur penelitian sebagai berikut:

1. Tahap pertama : Studi literatur, yakni mencari informasi terkait data yang bermanfaat untuk menunjang penelitian. Terkait sumber diperoleh dari jurnal ilmiah, buku, dan lain sebagainya. Studi literatur berfungsi sebagai rujukan sehingga dapat menciptakan konsep yang dapat digunakan untuk penelitian.
2. Tahap kedua : Melakukan perancangan alat sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT, tahap ini ialah penentuan perlengkapan atau komponen yang digunakan untuk membuat alat tersebut. Untuk pembuatan sistem menggunakan komponen berupa sensor ultrasonik HCSRFB-04, lampu LED 5V, transistor NPN, resistor 1k Ohm, dan ESP32-CAM.
3. Tahap ketiga : Melakukan pembuatan program sistem, kode program akan menggunakan kombinasi dari Arduino IDE dan Python di mana pada Arduino IDE akan melakukan perintah terhadap sensor untuk mendeteksi benda, lalu akan direkam oleh ESP32-CAM, lalu hasil penangkapan gambar pada pengiriman datanya dari alat ke server dan dari server dikirim ke komputer untuk dilakukan proses *image stitching* secara otomatis di mana jika berhasil maka akan disimpan baik data kumpulan foto dan hasil penggabungan gambar untuk dikirim ke telegram.
4. Tahap keempat : Implementasi, kode program serta pengujian sistem yang akan diimplementasikan pada ESP32-CAM. Dalam tahap ini dapat dinilai bahwa kode program yang telah dirancang berhasil atau gagal.
5. Tahap kelima : Pengecekan keberhasilan pada alat, jika pengimplementasian berhasil dapat dilanjutkan ketahap berikutnya, namun jika tidak berhasil maka alat tersebut perlu dilakukan perbaikan dengan penyesuaian kembali.
6. Tahap keenam : Pengambilan data alat sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT yang dihasilkan dari proses sensor Ultrasonik HCSRFB-04 dan pengujian sensor pada sistem, untuk tahap tersebut peneliti mengambil data dengan melakukan percobaan melewatkan benda keatas sensor di mana apakah gambar yang didapat dapat diproses dengan baik dari segi mendeteksi benda, mengambil gambar hingga memproses gambar menjadi bentuk panoramik.

7. Tahap ketujuh: Pada tahap ini berisi analisis data, pada tahap ini data-data yang diperoleh akan dianalisa, selanjutnya hasil dari data tersebut dibuat dalam bentuk perbandingan terhadap efektivitas dari alat.
8. Tahap kedelapan: Penyelesaian laporan, tahap ini merupakan tahap untuk menggabungkan keseluruhan hasil dari data yang sudah diambil, dianalisis akan digabungkan menjadi satu serta pada tahap ini merupakan tahap akhir dalam alur penelitian tersebut.

### **3.1 ALAT DAN BAHAN**

Sub bab ini merupakan penjelasan mengenai perangkat dan bahan yang nantinya diaplikasikan untuk membuat sistem ini, beserta fungsinya akan dijelaskan pada sub-sub bab 3.1.1 Perangkat Keras (*Hardware*) dan 3.1.2 Perangkat Lunak (*Software*).

#### **3.1.1 Perangkat Keras (*Hardware*)**

Perangkat keras merupakan bagian utama dalam penelitian ini maka dari itu sangatlah diperlukan untuk menjalankan instruksi dari perangkat lunak (*software*), maka perangkat yang digunakan untuk penelitian ini sebagai berikut:

1. ESP32-CAM  
Berfungsi untuk mengambil gambar saat melakukan pengujian dan memproses data.
2. Sensor Ultrasonik HCSR04  
Digunakan sebagai mendeteksi benda yang akan melewati alat.
3. Transistor NPN  
Berfungsi sebagai saklar atau penguat. Ini memungkinkan kontrol perangkat berdaya tinggi seperti LED dengan sinyal berdaya rendah dari mikrokontroler seperti ESP32-CAM. Tanpa transistor, menghubungkan langsung LED ke ESP32-CAM dapat menyebabkan arus berlebih yang dapat merusak komponen karena perbedaan kebutuhan daya mereka.
4. Lampu LED 5V  
Berfungsi untuk menerangi bagian bawah objek saat dilakukan uji agar hasil gambar yang dihasilkan cukup terang.

5. Resistor 1k Ohm

Berfungsi untuk membatasi daya yang masuk pada lampu LED 5V sebesar 1000 Ohm supaya tidak terlalu terang sehingga lebih bagus dalam pengambilan gambar bawah mobil.

6. Komputer dengan spesifikasi *Hardware* Intel Core i3-6100 @ 3.7GHz 16GB DDR4 2133MHz *Software* Endeavour OS RAM 16 GB

Berfungsi untuk memproses penggabungan gambar, menyimpan gambar, dan media untuk uji QoS menggunakan *software* Wireshark,.

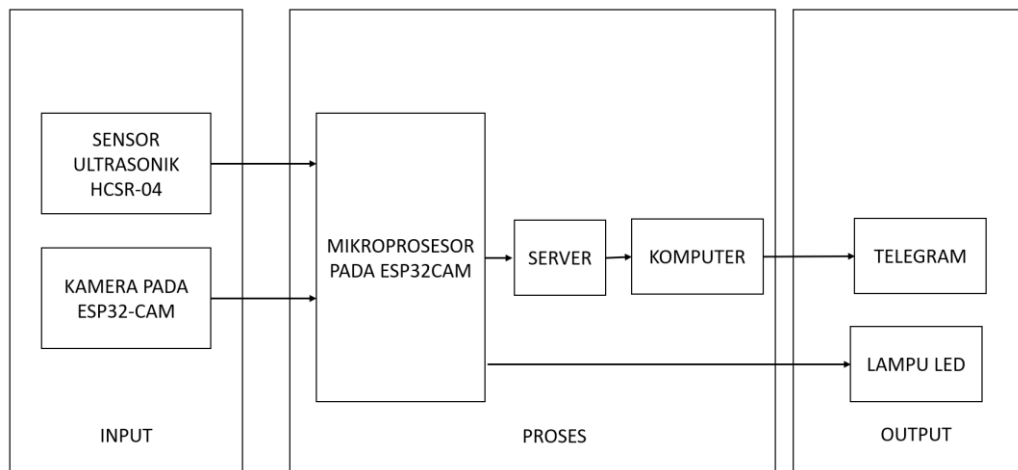
### 3.1.2 Perangkat Lunak (*Software*)

*Software* atau perangkat lunak digunakan sebagai sarana interaksi dan penghubung antara pengguna dengan perangkat keras (*hardware*). Pada penelitian ini menggunakan *software* Arduino IDE dan Python. Arduino IDE merupakan sebuah aplikasi yang berguna sebagai pemrograman untuk mikrokontroler, bagian dari penyusunan program tersebut merupakan kode program dari sensor Ultrasonik HCSRFB-04 dan pengambilan gambar menggunakan ESP32-CAM. Python digunakan sebagai pengolah gambar yang telah ditangkap untuk menjadikannya gambar memanjang (panorama). Telegram digunakan untuk menampilkan hasil dari nilai sensor dan gambar yang ditangkap agar dapat dilihat dari aplikasi pengguna. Pada pengujian terkait *Quality of Service* (QoS) memakai *software* Wireshark.

## 3.2 PERANCANGAN SISTEM

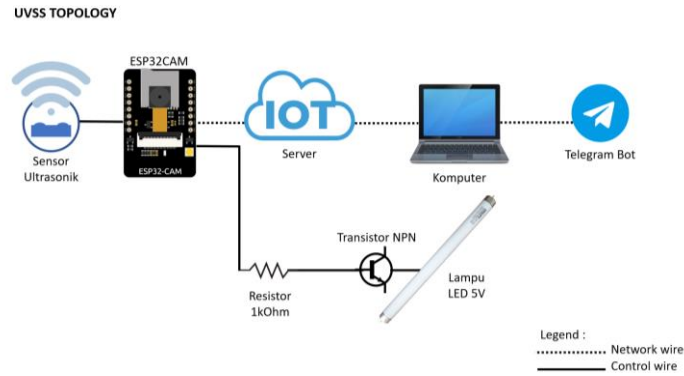
Pada perancangan sistem alat sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT, dijelaskan mengenai diagram blok sebagai gambaran untuk sistem kerja alat serta pemodelan sistem tersebut. Hal ini bertujuan untuk membantu peneliti melakukan analisis dari permasalahan yang ada. Dalam setiap perencanaan dan pembuatan suatu alat, diperlukan sebuah blok diagram yang berfungsi untuk mempermudah dalam menentukan alur kerja alat. Selain itu, blok diagram juga berguna untuk mengetahui bagian-bagian sistem dari suatu alat tersebut secara lebih rinci dan terstruktur. Disamping penggunaan blok diagram, dibutuhkan juga topologi sistem. Topologi sistem memiliki fungsi yang sangat

penting, yakni membantu mengidentifikasi hubungan antara elemen-elemen sistem, membantu memodelkan sistem secara keseluruhan, membantu mengidentifikasi kelemahan-kelemahan sistem, serta membantu mengembangkan strategi perbaikan yang efektif. Pada Gambar 3.2, ditampilkan blok diagram dari sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT. Gambar tersebut memberikan visualisasi yang jelas mengenai bagaimana sistem ini bekerja dan bagaimana setiap komponen saling berinteraksi. Pada Gambar 3.2 juga akan menampilkan blok diagram sistem *monitoring* bawah kendaraan guna pengecekan keamanan berbasis IoT, yang menunjukkan detail lebih lanjut mengenai bagian-bagian spesifik dari sistem tersebut.



**Gambar 3.2 Blok Diagram Sistem *Monitoring* Bawah Kendaraan Guna Pengecekan Keamanan Berbasis IoT**

Pada blok diagram sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT menjelaskan bahwa inputan terdiri dari tangkapan gambar dari kamera pada ESP32CAM, dan tangkapan jarak dari sensor ultrasonik HCSR-04. Pada bagian proses terdiri dari pemrosesan pada mikrokontroler dari ESP32-CAM, pengiriman gambar ke server lalu ke komputer, dan pemrosesan penggabungan gambar pada komputer. Pada *outputnya* yakni tampilan hasil penggabungan gambar pada bot telegram, dan cahaya yang dikeluarkan dari lampu LED. Pada Gambar 3.3 akan menampilkan topologi sistem *monitoring* bawah kendaraan guna pengecekan keamanan berbasis IoT



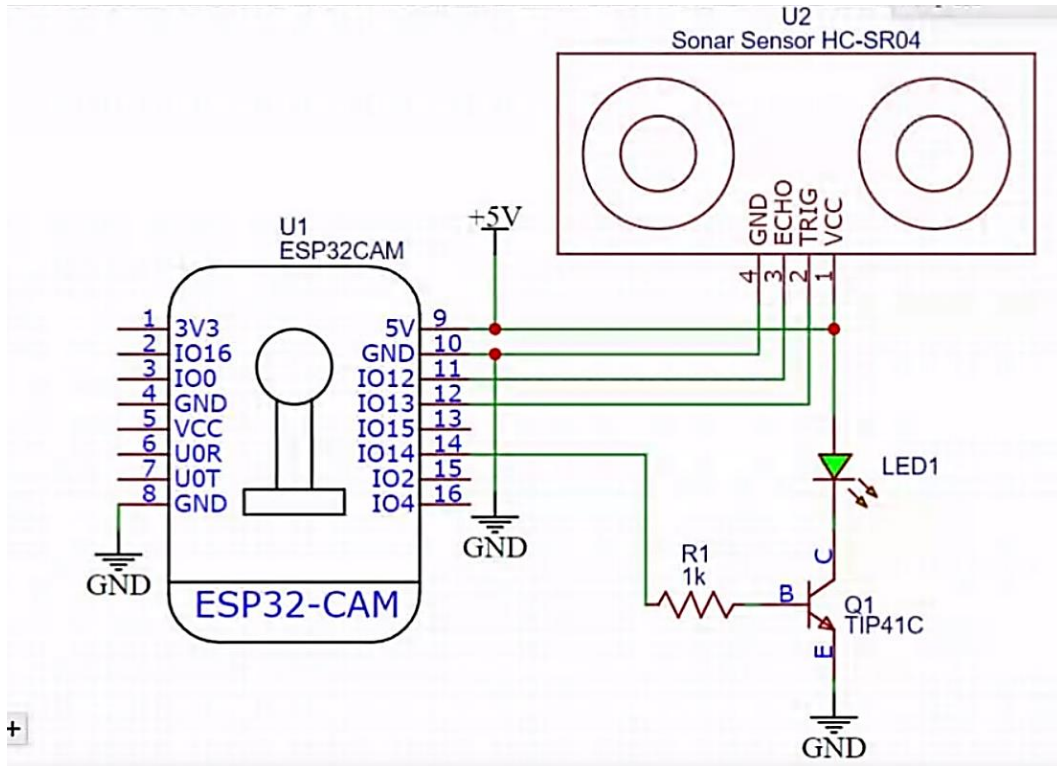
**Gambar 3.3 Topologi Sistem *Monitoring* Bawah Kendaraan Guna Pengecekan Keamanan Berbasis IoT**

Pada Gambar 3.3 merupakan topologi sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT. Pada Gambar 3.3 merupakan topologi sistem di mana berisi koneksi lebih lanjut mengenai hubungan antar komponen. Pada topologi sistem memberikan gambaran tentang alur kerja sistem tersebut. Pada saat sensor ultrasonik HCSR-04 mendeteksi benda maka ESP32-CAM akan melakukan pengambilan gambar dan menyalakan lampu LED selama masih mendeteksi benda. Saat sensor ultrasonik HCSR-04 tidak lagi mendeteksi benda maka akan berhenti melakukan penangkapan gambar dan menonaktifkan lampu. Pada koneksi dari lampu LED dan sensor ultrasonik HCSR-04 ke ESP32-CAM terkoneksi dengan kabel sedangkan pada ESP32-CAM menuju server lalu menuju komputer dan dikirim ke telegram memiliki koneksi secara *wireless*. Hasil dari penangkapan gambar akan dikirim ke server secara otomatis dan dari server akan diambil oleh komputer agar diproses menggunakan python untuk menggabungkan gambar-gambar yang ditangkap untuk diproses menjadi gambar panorama/memanjang yang dapat menampilkan sisi depan hingga belakang dari benda/kendaraan. Jika berhasil proses penggabungan gambar berhasil maka akan disimpan baik data kumpulan foto dan hasilnya akan dikirim ke telegram. Sedangkan, jika proses penggabungan gambar gagal dikarenakan gambar yang ditangkap tidak padu, buram ataupun tepi nya tidak saling terkait maka gambar tidak disimpan di penyimpanan internal komputer dan tidak dikirim ke telegram.

### 3.3 PERANCANGAN PERANGKAT KERAS

Pada perancangan perangkat keras menjelaskan mengenai skema hubungan antar komponen satu dengan lainnya secara rinci. Pada Gambar 3.4 akan

menampilkan *wiring diagram* dari perangkat sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT.



**Gambar 3.4 Wiring Diagram**

Pada *wiring diagram* menunjukkan koneksi di mana ESP32-CAM tersambung dengan sensor ultrasonik HCSR04 pada 5V, GND, GPIO 11, dan GPIO 12. Lalu tersambung dengan lampu LED pada GPIO 14 dan 5V di mana juga tersambung dengan resistor sebagai penurun daya dan transistor NPN sebagai saklar untuk menyalakan lampu LED. Pada Tabel 3.1 akan menunjukkan konfigurasi pin perangkat keras.

**Tabel 3.1 Konfigurasi Pin Perangkat Keras**

No	Sensor Ultrasonik HCSR04	ESP32-CAM	Transistor	Lampu LED 5V	Sumber daya
1	VCC	5V	C	5V	5V
2	GND	GND	E		
3	TRIG	GPIO11			
4	ECHO	GPIO12			
5		GPIO14	B		

Komponen *hardware* atau perangkat keras yang digunakan sebagai berikut :

1. ESP32-CAM

Berfungsi untuk mengambil gambar saat melakukan pengujian dan memproses data.

2. Sensor Ultrasonik HCSRFB-04

Digunakan sebagai mendeteksi benda yang akan melintas.

3. Transistor NPN

Berfungsi sebagai saklar atau penguat. Ini memungkinkan kontrol perangkat berdaya tinggi seperti LED dengan sinyal berdaya rendah dari mikrokontroler seperti ESP32-CAM. Jika tanpa transistor lalu menghubungkan langsung LED ke ESP32-CAM dapat menyebabkan arus berlebih yang dapat merusak komponen karena perbedaan kebutuhan daya mereka.

4. Lampu LED 5V

Berfungsi untuk menerangi bagian bawah objek saat dilakukan uji agar hasil gambar yang dihasilkan cukup terang.

5. Resistor 1k Ohm

Berfungsi untuk membatasi daya yang masuk pada lampu LED 5V sebesar 1000 Ohm supaya tidak terlalu terang sehingga bagus dalam pengambilan gambar bawah mobil.

Pada Gambar 3.4 dan Tabel 3.1 di atas dijelaskan fungsi dari masing-masing pinoutnya. Sensor Ultrasonik HCSRFB-04 adalah suatu sensor yang digunakan untuk mendeteksi jarak objek dengan memanfaatkan gelombang ultrasonik. Sensor ini memiliki 4 pin, yaitu VCC, GND, *trigger*, dan *echo*. Pin *trigger* digunakan untuk memicu sensor agar mengirimkan sinyal gelombang ultrasonik. Sedangkan pin *echo* digunakan untuk menerima sinyal pantul dari objek yang dideteksi. Dalam penggunaannya, ketika kita memberikan tegangan positif pada pin *trigger* selama 10 $\mu$ S, maka sensor akan mengirimkan 8 step sinyal ultrasonik dengan frekuensi 40kHz. Kemudian, gelombang yang terbentuk dipancarkan mengenai target. Hasil pantulan gelombang tersebut nantinya akan diterima oleh *receiver piezoelectric* untuk dikalkulasikan waktu pengiriman dan waktu diterimanya gelombang pantul tersebut. Hasil pengalkulasian itulah nanti yang akan kita peroleh sebagai nilai jarak.



VCC dan GND adalah istilah yang sering digunakan dalam dunia elektronik. VCC adalah singkatan dari *Voltage Collector Collector* dan GND adalah singkatan dari *Ground*. Pada sensor ultrasonik HCSRFB-04, VCC dan GND digunakan untuk memberikan tegangan positif dan negatif. Tegangan positif pada VCC tidak berfungsi untuk mengirim data sensor ke mikrokontroler. Fungsi VCC dan GND pada sensor ultrasonik HCSRFB-04 adalah untuk memberikan catu daya pada sensor. *Grounding* pada suatu arus dilakukan untuk menghindari terjadinya noise atau gangguan pada sinyal yang dihasilkan oleh sensor. Dengan *Grounding*, maka sinyal yang dihasilkan oleh sensor akan lebih stabil dan akurat. Selain itu, *Grounding* juga berfungsi untuk melindungi komponen elektronik dari lonjakan tegangan yang tidak diinginkan.

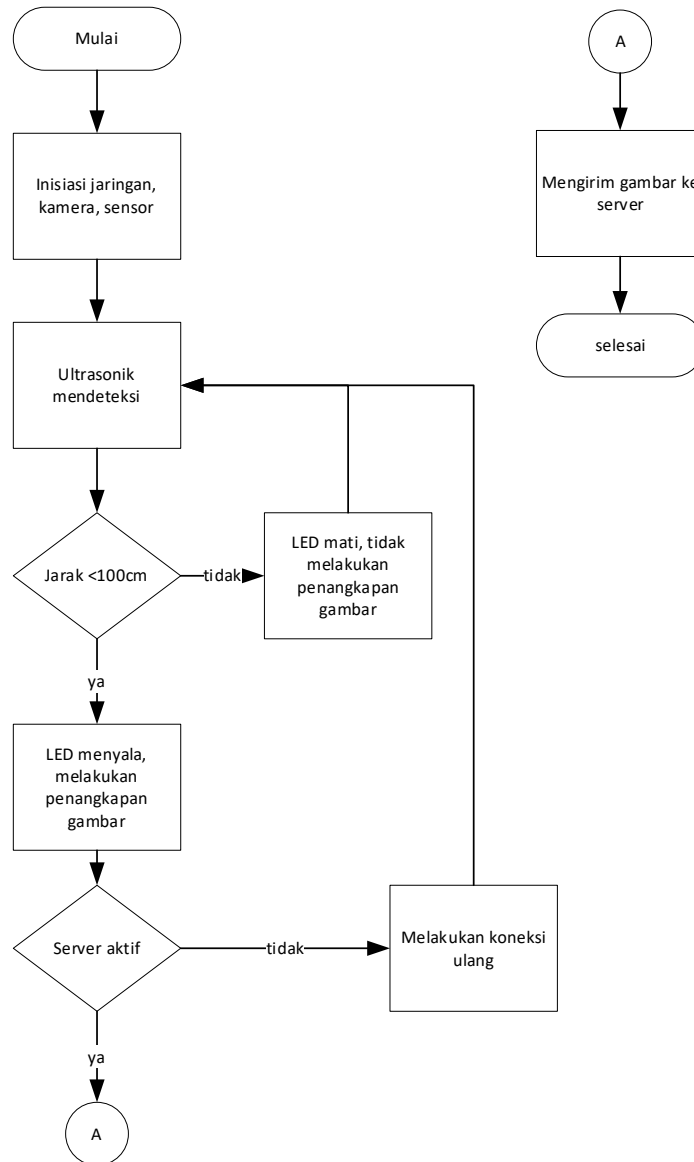
Pada ESP32-CAM memiliki 16 pin tetapi pada percobaan ini hanya memakai 4 pin yakni 5V, GND, IO11, IO12, IO14. Pin 5V digunakan untuk memberikan tegangan positif 5V pada sensor ultrasonik dan pada lampu LED 5V. GND digunakan untuk memberikan *Grounding*. GPIO 11, GPIO 12, dan GPIO 14 berfungsi sebagai pin *output*. Adapun pin lain yang tidak disebutkan di wiring diagram dikarenakan akan dilepas saat sudah ter-*upload* kode program yakni pin GPIO 0. GPIO 0 berfungsi digunakan untuk memilih mode booting. GPIO 0 akan dilakukan catu pada GND yakni pada ESP32-CAM, pin GPIO 0 digunakan untuk menentukan apakah ESP32 berada dalam mode *flashing* atau tidak. Ketika pin GPIO 0 dihubungkan ke GND, maka ESP32-CAM akan masuk ke dalam mode *flashing* dan kita dapat meng-*upload* kode ke dalam board.

Selain itu, penting untuk dicatat bahwa penggunaan pin GPIO pada ESP32-CAM sangat fleksibel dan dapat disesuaikan dengan kebutuhan proyek. Misalnya, pin GPIO lainnya dapat digunakan untuk menghubungkan sensor tambahan atau perangkat lain yang diperlukan dalam sistem *monitoring* kendaraan. Dengan demikian, ESP32-CAM menawarkan banyak kemungkinan untuk pengembangan lebih lanjut dan penyesuaian sistem sesuai dengan kebutuhan spesifik.

### **3.4 PERANCANGAN PERANGKAT LUNAK**

Pada bagian perancangan perangkat lunak terdapat 2 *flowchart* yang di mana berisikan alur kerja alat pada bagian pemrosesan ESP32-CAM dan alur kerja

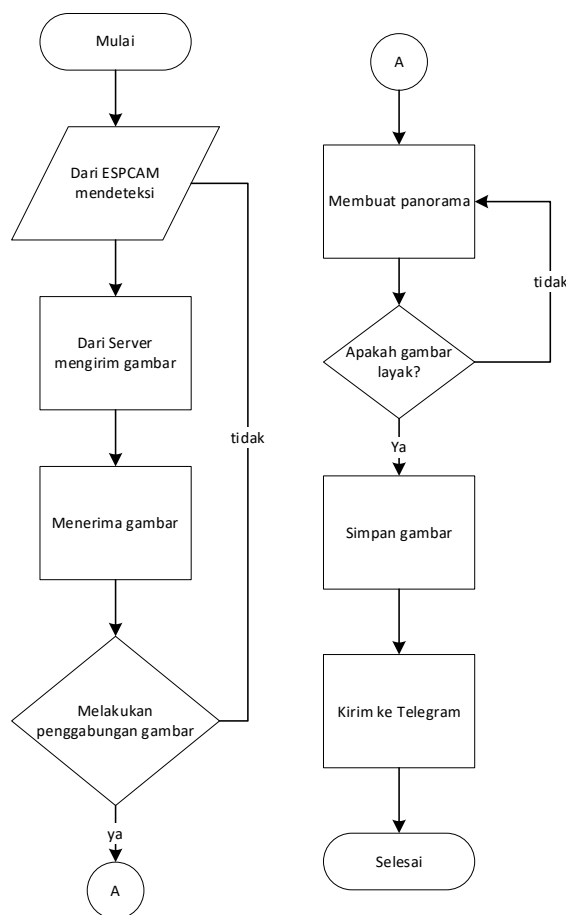
pemrosesan pada komputer. Untuk *flowchart* pada bagian ESP32-CAM akan dijelaskan pada gambar 3.5 dan *flowchart* mengenai pemrosesan pada komputer akan dijelaskan pada Gambar 3.6.



**Gambar 3.5 Flowchart Perangkat Lunak Pada Sisi ESP32-CAM**

Dari Gambar 3.5 *flowchart* perangkat lunak menjelaskan secara rinci alur sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT pada sisi pemrosesan ESP32-CAM, di mana tahap pertama merupakan inisiasi *library* jaringan, kamera dan sensor ultrasonik HCSRFB-04, tahap berikutnya yakni penentuan variabel terkait sensor ultrasonik HCSRFB-04.

Variabelnya yakni saat mendeteksi benda pada jarak kurang dari 100 cm maka akan menyalakan lampu dan melakukan penangkapan gambar secara terus menerus. Saat jarak sudah tidak memenuhi kurang dari 100 cm maka akan mematikan lampu dan berhenti melakukan penangkapan gambar. 100 cm didasarkan pada ketinggian mobil dari permukaan ke bagian kolongnya yang ada di rentang 60 – 80 cm. Saat melakukan penangkapan gambar dan menyalakan lampu akan menyalakan server dan mengirimkan gambar ke server. Gambar ditampung pada server untuk diambil oleh komputer untuk proses selanjutnya. Pada Gambar 3.6 menampilkan *flowchart* perangkat lunak pada sisi pemrosesan python di komputer



**Gambar 3.6 Flowchart Perangkat Lunak Pada Sisi Pemrosesan Python di Komputer**

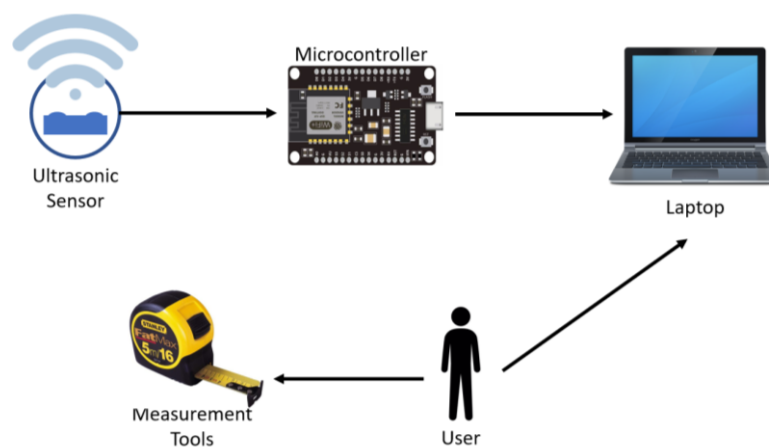
Dari Gambar 3.6 *flowchart* menjelaskan secara rinci alur sistem *monitoring* kendaraan guna pengecekan keamanan berbasis IoT pada sisi pemrosesan python

di komputer, di mana tahap pertama yakni saat mendeteksi ESP32-CAM, lalu dari server melakukan pengiriman gambar. Gambar tersebut akan diterima di komputer dan kemudian akan dilakukan penggabungan gambar (*image stitching*). Akan dilakukan pengambilan keputusan yakni jika dari hasil penggabungan tersebut menghasilkan gambar yang layak di mana sisi tepi dari gambar padu, gambar nya sesuai maka akan hasil panorama akan disimpan sedangkan jika hasil tidak sesuai maka tidak disimpan. Gambar hasil panorama tersebut lalu dikirimkan ke bot telegram.

### 3.5 PENGUJIAN SISTEM

#### 3.5.1 Validasi Nilai Sensor Ultrasonik

Validasi dilakukan guna mengetahui apakah nilai sensor yang didapatkan sudah sesuai dengan pengukuran yang sebenarnya. Pada pengujian ini akan membandingkan nilai pengukuran nilai sensor yang akan ditampilkan pada *serial monitor* komputer dengan pengukuran menggunakan meteran konvensional (meteran *roll*). Telapak tangan akan ditempat sesuai dengan ketetapan jarak dari meteran roll dan dibandingkan dengan pengukuran dengan sensor ultrasonik HCSRFB-04. Skenario pengukuran nilai sensor ultrasonik dapat dilihat pada Gambar 3.7.

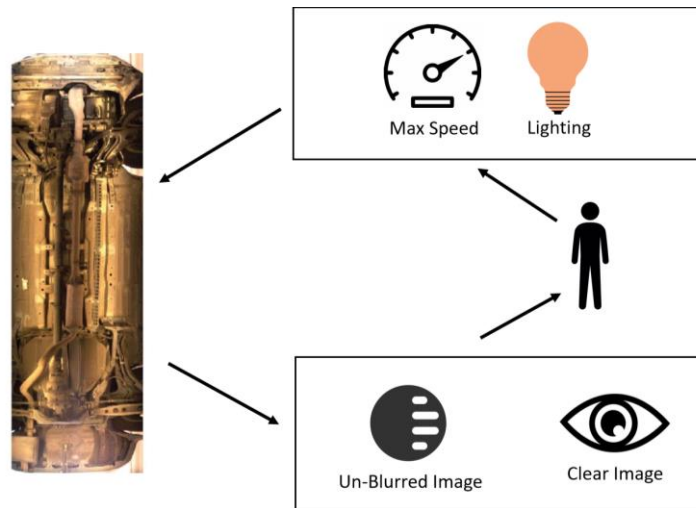


**Gambar 3.7 Skenario Pengukuran Nilai Sensor Ultrasonik**

#### 3.5.2 Validasi Gambar Yang Dihasilkan

Validasi dilakukan guna mengetahui apakah gambar yang didapatkan sudah memenuhi kriteria yakni cukup pencahayaan, dapat melingkupi sisi depan hingga

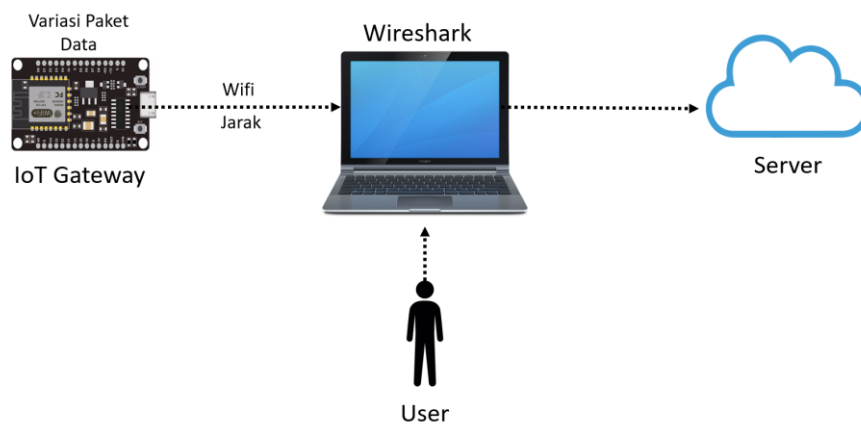
belakang dari bawah kendaraan, serta kejelasan gambar. Pada pengujian ini akan melakukan penyesuaian terkait lampu yang digunakan sebagai penerangan bawah mobil, penyesuaian maksimum kecepatan dari mobil guna menghasilkan gambar yang jelas. Skenario validasi gambar ada pada Gambar 3.8.



**Gambar 3.8 Skenario Validasi Gambar**

### 3.5.3 Pengujian Kualitas Layanan / *Quality of Service* (QoS)

Pada pengujian ini berfungsi untuk menilai performansi komunikasi pada kualitas layanan. QoS yang diukur antara lain *delay* dan *packet loss*. Pengukuran dilakukan dengan menggunakan komputer yang tersambung ke wifi yang sama-sama terkoneksi ke alat lalu akan dilakukan pengujian QoS menggunakan *software* Wireshark dengan menguji koneksi antara alat dengan server. Skenario pengujian QoS tertampil pada Gambar 3.9.



**Gambar 3.9 Skenario Pengujian Kualitas Layanan QoS**