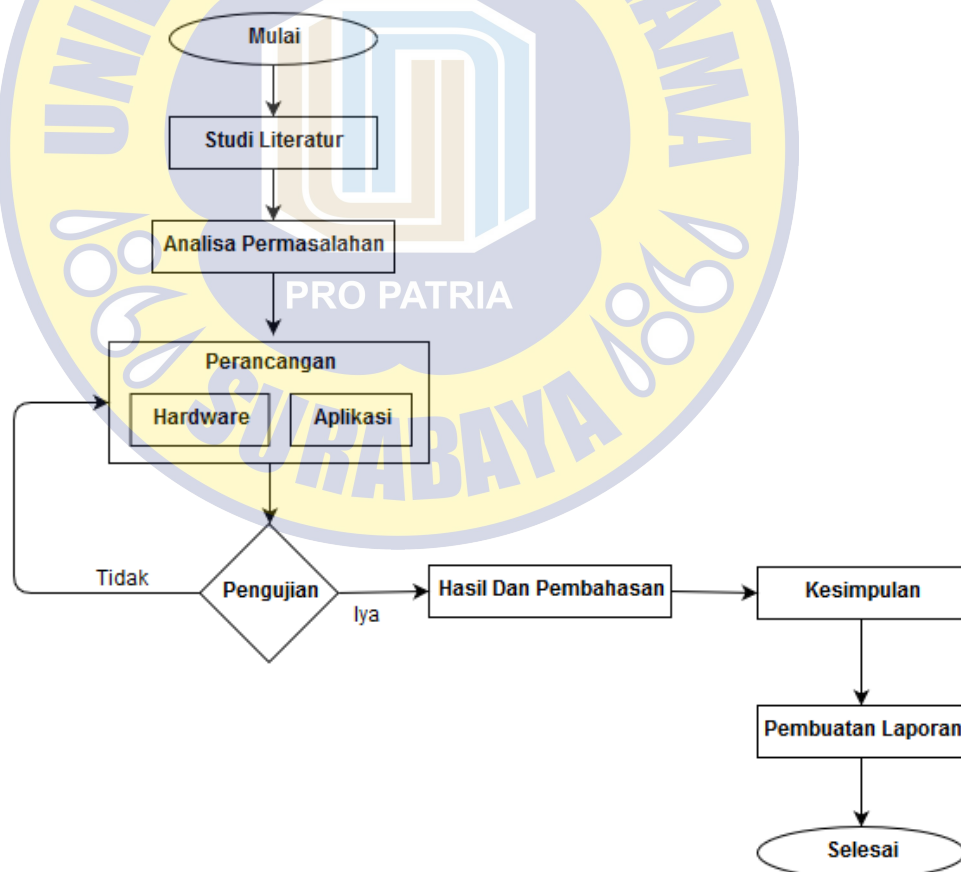


### BAB III

## METODOLOGI PENELITIAN

Metodologi atau tahapan pengerjaan merupakan hal yang sangat diperlukan untuk melakukan suatu penelitian, sehingga pada alur pengerjaan penelitian teratur, terarah, dan sistematis. Metode yang dilakukan dalam penelitian ini meliputi studi literatur, analisa permasalahan, perancangan *hardware* dan aplikasi, pengujian, hasil dan pembahasan, kesimpulan, pembuatan laporan.



Gambar 3.1 *Flowchart* Alur Penelitian.

### 3.1 Studi Literatur

Proses studi literatur meliputi pencarian penelitian terdahulu yang mencakup tema yang hampir sama beserta dasar-dasar teori yang dibutuhkan dalam penelitian ini. Teori-teori yang terkait dengan permasalahan penelitian seperti dasar rangkaian elektronik, komponen-komponen pendukung, bahasa pemrograman Arduino, Android, dan teori pendukung lain yang terkait dengan penelitian. Studi literatur dilakukan dengan membaca langsung dari beberapa *e-book*, jurnal terdahulu dan artikel di internet, kemudian merangkumnya dan menulisnya kembali sesuai dengan kebutuhan penelitian.

Penelitian ini terpusat pada bagaimana cara mengetahui jumlah penumpang bus pada setiap rute melalui *smartphone*. Dalam studi literatur dilakukan pencarian informasi mengenai segala sesuatu yang berkaitan dengan penelitian ini, diantaranya adalah sebagai berikut:

1. ESP32 (IOT).
2. RFID.
3. Web.

### 3.2 Analisa Permasalahan

Dalam perancangan alat ini, diperlukan sebuah RFID yang terhubung dengan ESP32 (IOT). Setelah selesai, ESP32 (IOT) tersebut mengirim data ke database untuk kemudian memberikan notifikasi ke web.

### 3.3 Perancangan

Pada tahap ini peneliti melakukan beberapa tahapan-tahapan mengenai perancangan purwarupa sebuah alat untuk mengetahui jumlah penumpang bus pada setiap rute berbasis *IoT*. mulai dari blok diagram, desain rancangan *hardware* yang didalamnya adalah tentang kebutuhan *hardware* sehingga alat dapat bekerja dengan baik. Terdapat juga perancangan didalam program meliputi *flowchart* program penelitian.

#### 3.3.1 Kondisi Existing Bus Berdasarkan Rute



Gambar 3.2 Kondisi Terminal Purabaya

Jika penumpang bus ingin masuk atau keluar dari bus, penumpang harus menempelkan kartu RFID ke sensor RFID yang terdapat pada pintu masuk dan keluar bus. Kemudian sensor RFID mengirimkan data tersebut kepada ESP32 yang kemudian dilanjutkan oleh ESP32 untuk mengirimkan data ke website untuk mengetahui jumlah penumpang bus.

## Jalur Rute Bus Surabaya

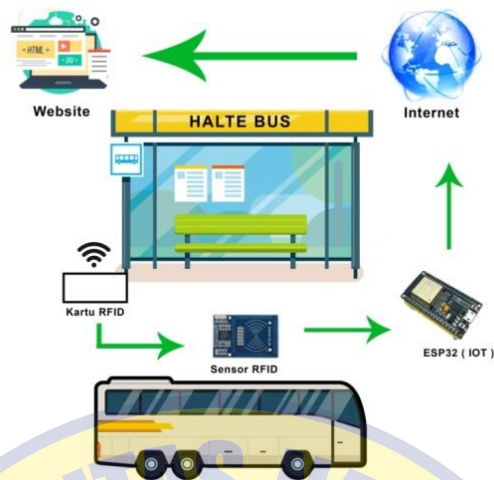


Gambar 3.3 Jalur Rute Bus Yang Ada Di Kota Surabaya

(Sumber : <https://surabayastory.com/> )

### 3.3.2 Perancangan Sistem Bus Berbasis *IoT*

Pada sebuah penelitian diperlukan sebuah rancangan awal seperti layaknya sebuah desain atau gambar ilustrasi yang menerangkan alur sistem penelitian.

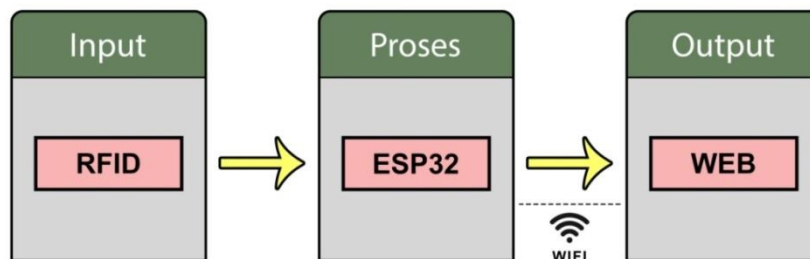


Gambar 3.4 Ilustrasi Prototype Bus Berbasis *IoT*.

Terdapat sebuah kondisi dalam penelitian ini yakni ketika sensor mendeteksi adanya kendaraan yang parkir. Berikut penjelasan secara runtut tentang sebuah kondisi tersebut:

- ✓ Kartu RFID di pasang pada Halte A/ B / C. Kemudian RFID sensor mengirim data pada ESP32, yang berfungsi untuk memproses data dan dikirimkan melalui website.

### 3.3.3 Blok Diagram



Gambar 3.5 Blok Diagram Proses Prototype Bus.

#### A. Blok *Input*

Pada blok *input* ini terdapat RFID yang berfungsi untuk mengirim data pada ESP32. Data yang dikirim adalah jika ada penumpang bus yang masuk atau keluar dari bus.

#### B. Blok Proses

Pada blok proses terdapat ESP32 (IOT) yang berfungsi sebagai pusat *prototype*. Jika ada penumpang yang masuk atau keluar, maka *input* yang terdapat pada RFID dan Arduino Uno menghubungkan dan mengaktifkan ESP32 (IOT).

#### C. Blok *Output*

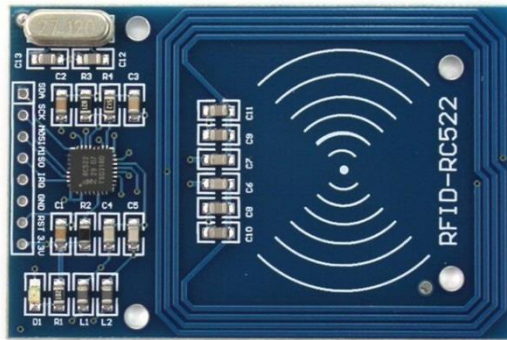
Blok *output* atau hasil keluaran setelah diproses oleh ESP32 (IOT) akan mengirimkan data kepada *website* server untuk mengetahui jumlah penumpang bus yang masuk atau keluar.

### 3.3.4 Rancangan Hardware Yang Dibutuhkan

Berikut desain rancangan *hardware* yang dibutuhkan untuk membangun atau merancang sebuah *prototype* bus berbasis *IoT*.

#### 3.3.3.1 RFID





Gambar 3.6 RFID

. Modul RFID pada penelitian ini berfungsi untuk mengirim data ke ESP32 yang kemudian akan mengirimkan data tersebut ke *internet*.

### 3.3.3.2 ESP32



Gambar 3.7 ESP32

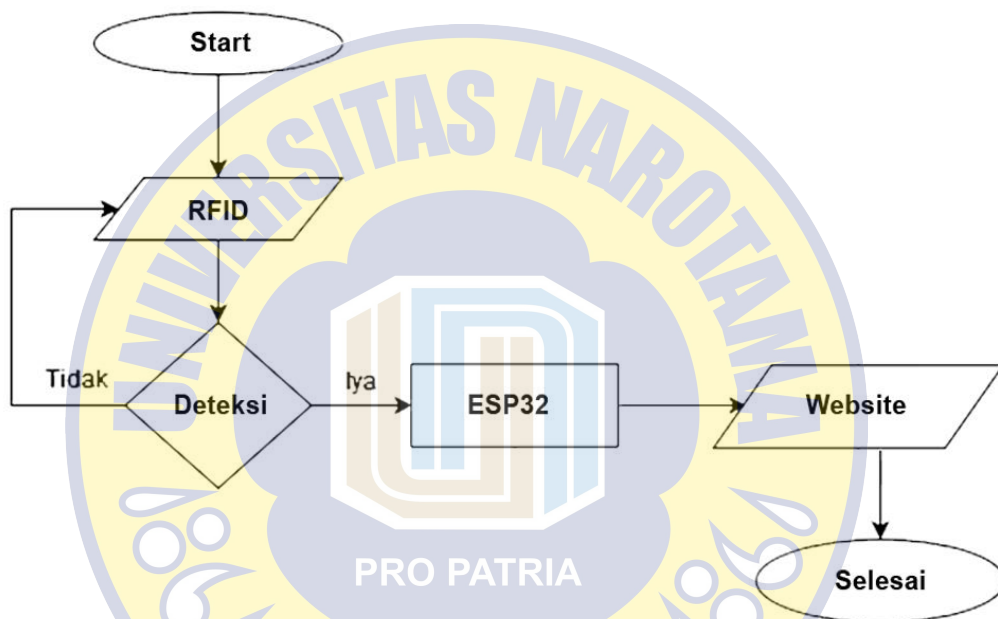
ESP32 pada penelitian ini berfungsi sebagai *IoT* yang penerima data dari sensor RFID, nantinya diproses lalu mengkoneksikan dengan *wifi* atau *hotspot* yang tersedia lalu akan mengirim data ke internet ( *website*).

### 3.3.4 Perancangan Software

Berikut *flowchart* perancangan program penelitian yang menjelaskan alur perancangan *prototype* bus berbasis *IoT*. *Flowchart* ini berguna untuk

menjelaskan alur sistem secara terurut dan memudahkan kita dalam memahami cara kerja dari suatu program.

### 3.3.4.1 Flowchart Sistem



Gambar 3.8 *Flowchart* Prototype Bus Berbasis *IoT*.

Gambar 3.6 merupakan urutan proses cara kerja program penelitian. Langkah pertama adalah RFID mendeteksi bila penumpang yang akan masuk mendekati kartu pada sensor. Kemudian data dari RFID akan diterima dan diproses oleh ESP32. Setelah itu, ESP32 mengirim data tersebut pada Website yang nantinya berfungsi untuk menampilkan jumlah keseluruhan penumpang bus yang masuk atau keluar di setiap haltenya.



### 3.3.5 Implementasi

Pada tahap implementasi sistem alat, implementasi yang pertama adalah implementasi semua komponen *input* yang harus bisa bekerja sebagaimana mestinya dan bisa memberikan data *input* kepada RFID. Kemudian ESP32 (*IoT*) harus bisa memproses *input* yang diterima. Serta semua komponen *output* dapat menerima dengan baik perintah yang telah diberikan oleh RFID.

### 3.3.6 Dokumentasi

Pada tahap ini dilakukan pengumpulan dokumen-dokumen dengan bukti yang konkrit. Tahap ini dilakukan setelah melalui beberapa proses pengujian dan pengelolaan terhadap sistem. Tujuannya adalah untuk mendapatkan dan memperlihatkan dokumen yang diperlukan sebagai bukti nyata atas suatu kegiatan yang telah dilakukan. Pendokumentasian berupa gambar atau foto alat yang digunakan dalam penelitian dan *source code* program yang telah diuji.