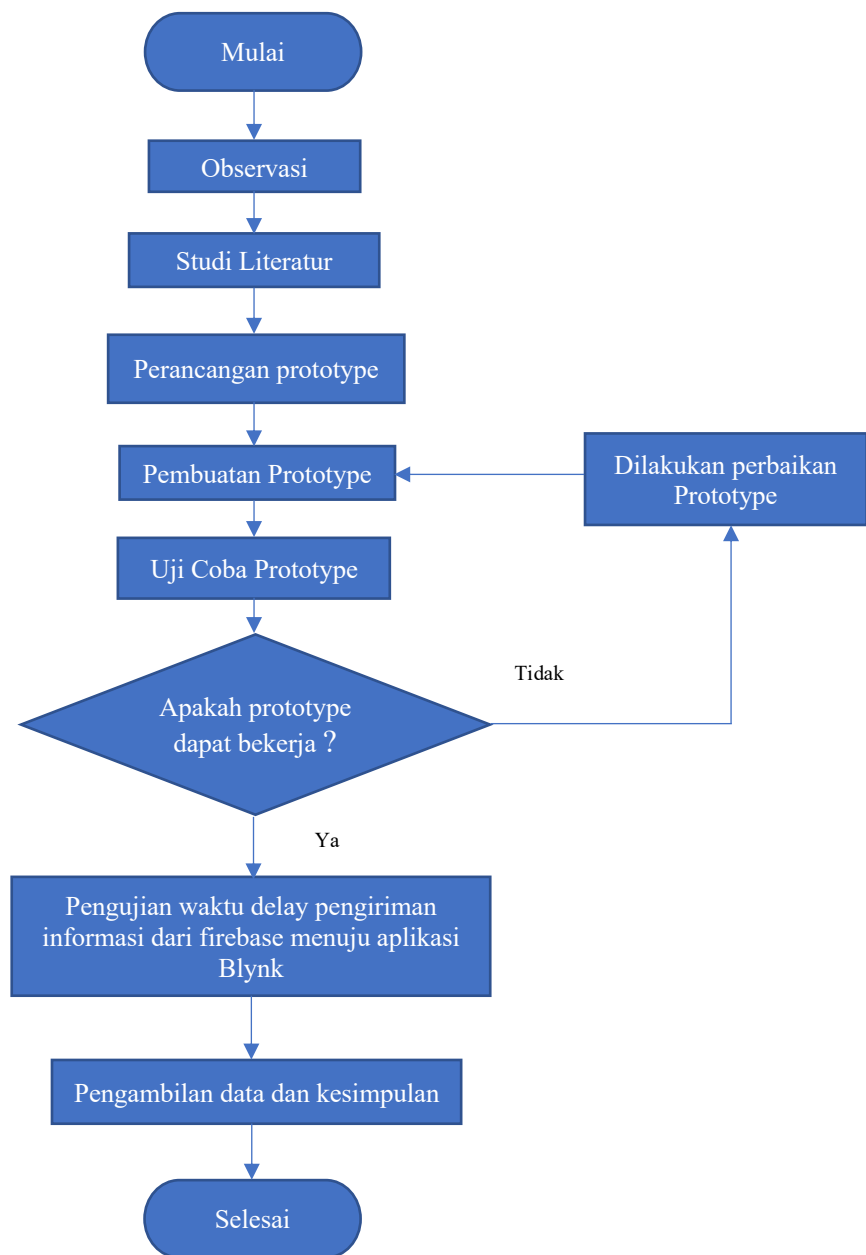


## BAB III METODE PENELITIAN

### 3.1. Prosedur Penelitian

Dalam perancangan Sistem Tempat Sampah Otomatis pada Gedung Bertingkat Berbasis *Internet of Things* (IoT), terdapat beberapa tahap yang harus dilalui untuk menghasilkan suatu alat yang dapat bekerja dengan baik. Tahap-tahap tersebut dijabarkan dalam bentuk *flowchart* pada Gambar 3.1.



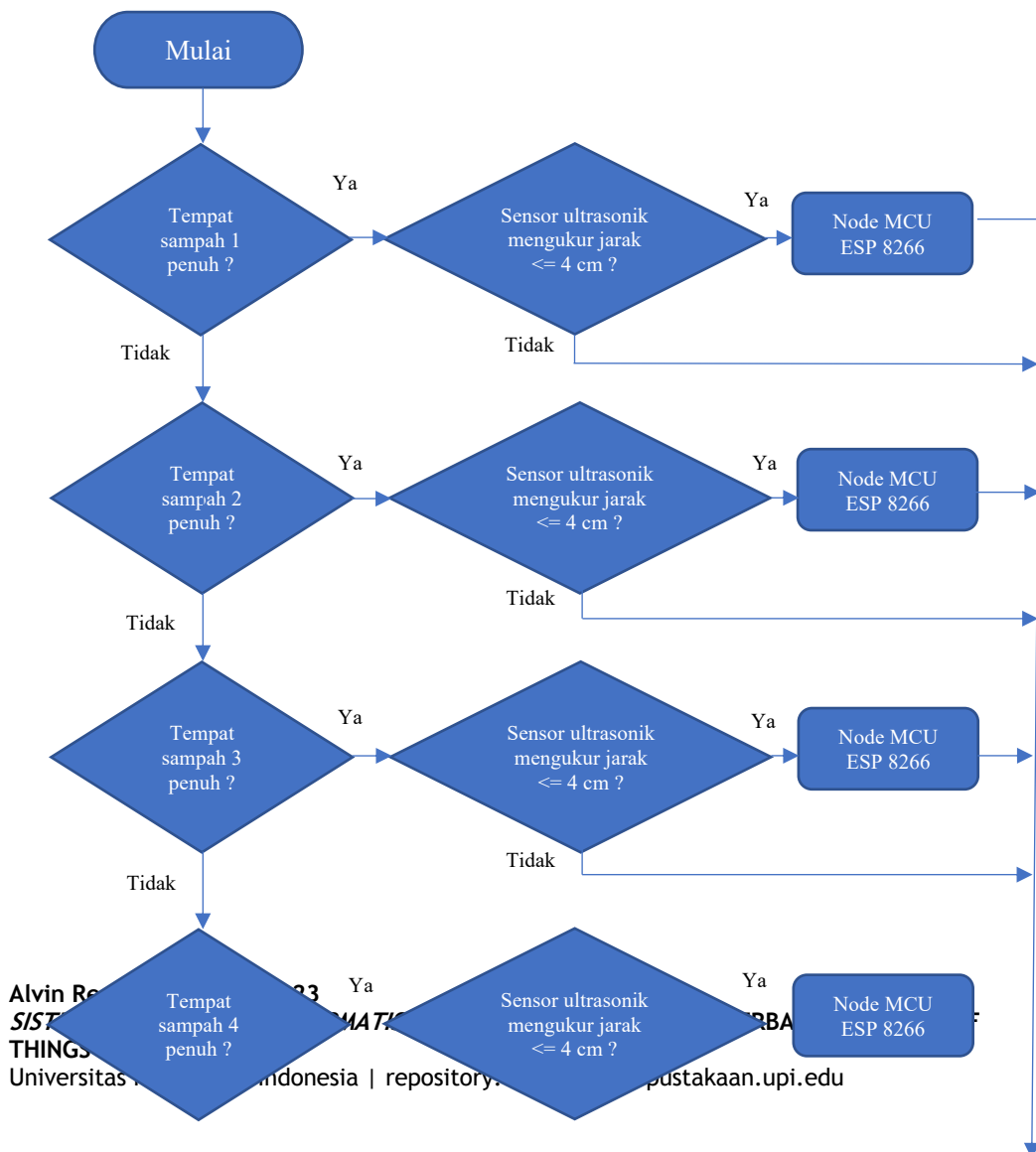
### Gambar 3.1. *Flowchart* Prosedur Penelitian

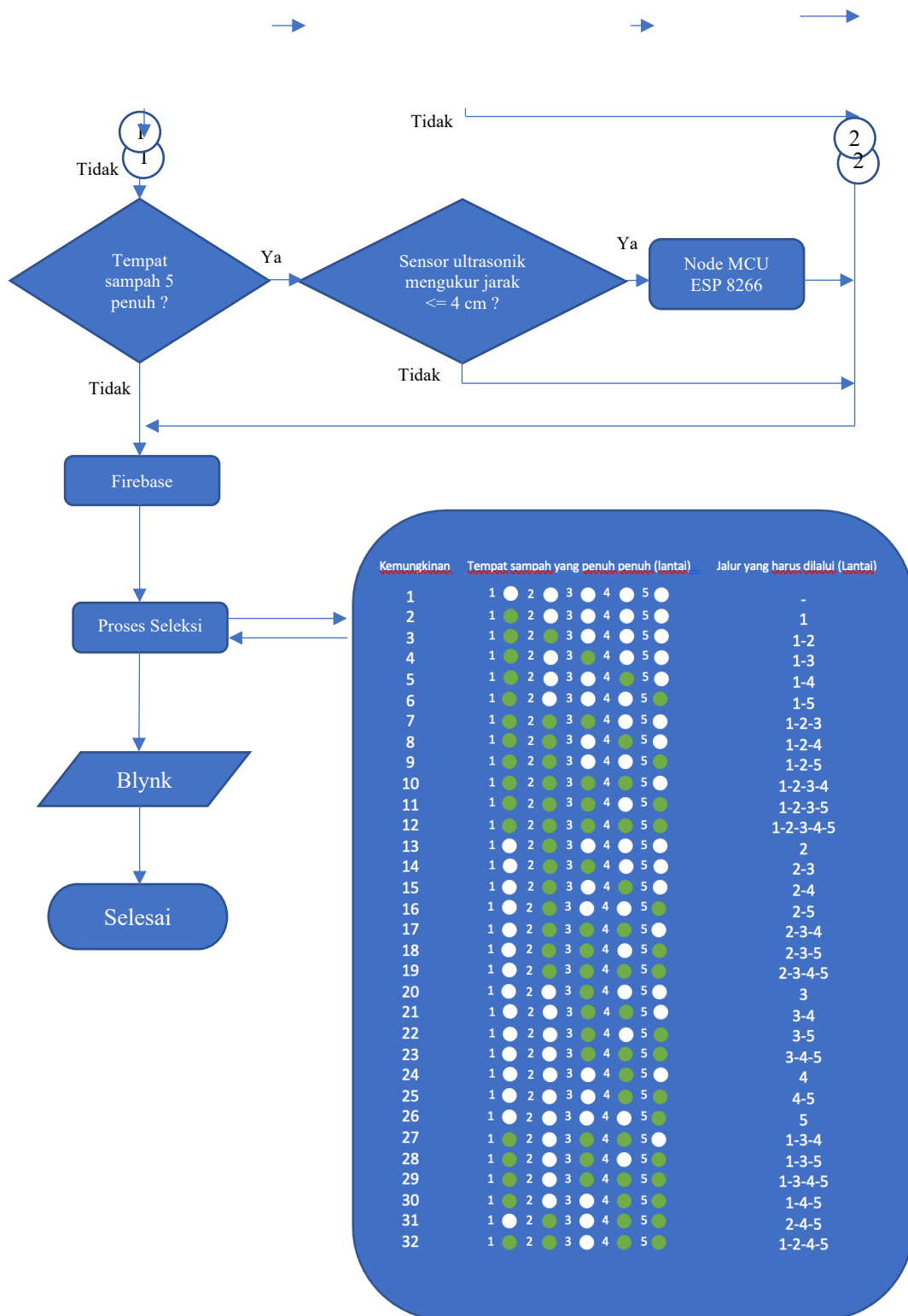
Dari *flowchart* prosedur penelitian yang diuraikan pada Gambar 3.1, Tahap pertama yang dilakukan yaitu tahap observasi. pada tahap ini, penulis mengamati masalah yang timbul pada pengelolaan sampah di gedung bertingkat sehingga dari permasalahan tersebut diharapkan dapat memberikan solusi melalui sistem pembuangan sampah yang akan dibuat. kemudian, pada tahap studi literatur, penulis mengumpulkan data dan informasi terkait yang berasal dari jurnal-jurnal yang dipublikasikan dari sumber yang kredibel seperti IEEE (*Institute of Electrical and Electronics Engineers*), *Scimago Journal and Country Rank*, DOAJ (*Directory of Open Access Journals*), dan jurnal-jurnal yang dipublikasikan oleh Universitas yang sudah terakreditasi baik dari dalam negeri maupun luar negeri. kemudian, pada tahap perancangan *prototype*, penulis membuat rancangan *schematic* dari *prototype* yang akan dibuat sebelum diaplikasikan pada *prototype* sungguhan dengan komponen elektronik yang sebenarnya. Lalu, pada tahap pembuatan *prototype*, dilakukan pembuatan *prototype* yang sebenarnya sesuai dengan perancangan yang sudah dilakukan sebelumnya. selanjutnya, pada tahap uji coba *prototype*, akan dilakukan proses pengujian alat untuk memastikan apakah *prototype* dapat bekerja dengan baik atau tidak. jika *prototype* tidak bekerja dengan baik, maka akan dilakukan perbaikan dan pengujian ulang pada *prototype*. Jika *prototype* dapat bekerja dengan baik, maka akan dilakukan tahap selanjutnya, yaitu pengujian *delay* pengiriman informasi dari situs Firebase menuju aplikasi Blynk. Pada tahap ini, dilakukan uji coba untuk mengetahui pengaruh jarak antara *prototype* tempat sampah otomatis dengan petugas kebersihan terhadap keberhasilan penerimaan informasi yang didapatkan pada aplikasi Blynk yang terdapat pada *smartphone* petugas. Dalam kasus ini, petugas kebersihan berada pada lantai pertama dan bertugas untuk membersihkan lima buah tempat sampah yang masing-masing berada pada lima lantai yang berbeda. Pada tahap ini juga, penulis melakukan uji coba untuk mengetahui waktu *delay* dari informasi yang diterima oleh aplikasi Blynk dari lima buah tempat sampah dengan jarak yang berbeda. Untuk mengetahui waktu *delay* yang terjadi pada pengiriman informasi dari Firebase

hingga mencapai aplikasi Blynk pada *smartphone*, penulis menggunakan fitur *serial monitor* yang terdapat pada aplikasi Arduino IDE sehingga hasil pengukuran waktu *delay* dapat diketahui secara akurat. Setelah melakukan uji coba, tahap terakhir yaitu pengambilan data dan kesimpulan. Pada tahap ini, penulis melakukan pengambilan data dari *prototype* tempat sampah pintar yang sudah dinyatakan dapat bekerja dengan baik dan mengambil kesimpulan dari data-data yang sudah didapatkan dari *prototype* tersebut.

### 3.2. Alur Kerja *Prototype*

Pada rancangan Sistem Tempat Sampah Otomatis pada Gedung bertingkat Berbasis *Internet of Things* (IoT), terdapat beberapa tahap yang harus dilalui agar rangkaian bekerja sesuai dengan yang diharapkan. Adapun tahap-tahap yang dilalui dijabarkan pada Gambar 3.2.



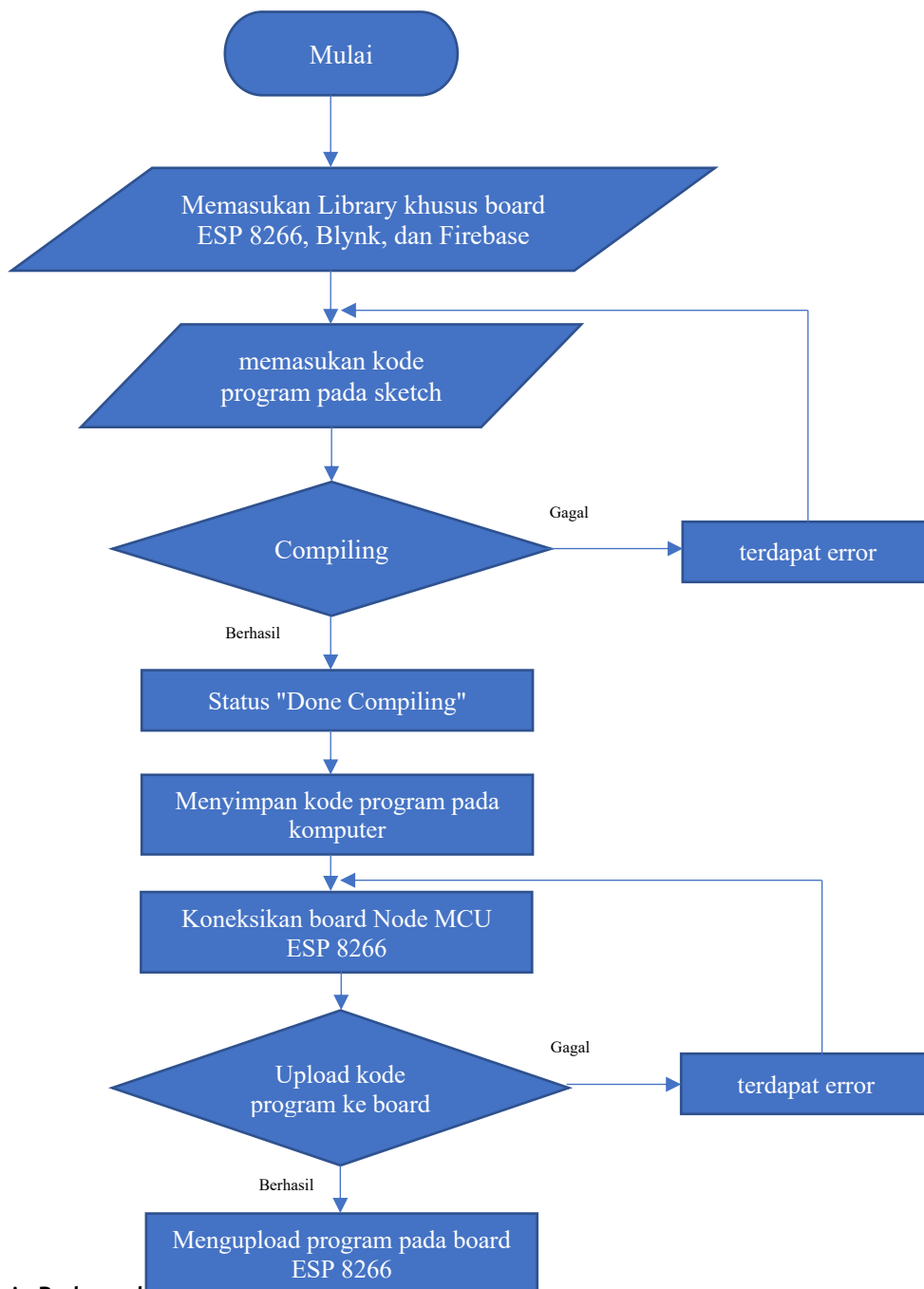


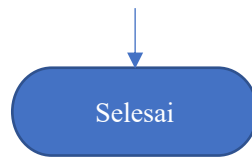
Gambar 3.2. Flowchart Alur Kerja pada Prototype

Pada masing-masing *prototype* tempat sampah, terdapat sensor HC-SR04 yang dipasang pada rangkaian. Pada sensor HC-SR04, jika jarak sensor terhadap objek sampah yang diukur ( $S$ ) kurang dari sama dengan 4 cm ( $S \leq 4$  cm), maka tempat sampah dinyatakan sudah penuh oleh sistem. Jika nilai ( $S$ ) lebih dari 2 cm ( $S > 4$  cm), maka proses tidak akan dilanjutkan hingga sensor HC-SR04 mendapatkan nilai ( $S \leq 4$  cm) karena tempat sampah dinyatakan belum penuh. selanjutnya, Node MCU ESP 8266 akan mengirimkan data ke *cloud server* Firebase untuk menyimpan data keterisian tempat sampah dalam bentuk persentase bersamaan dengan data keterisian tempat sampah dari tempat sampah lainnya. Kemudian, pada Firebase, data keterisian seluruh tempat sampah baik yang sudah hampir penuh (80%), penuh (100%), maupun yang belum penuh ( $< 80\%$ ) akan disimpan dalam *realtime database* pada Firebase dan akan digunakan kembali untuk melakukan proses seleksi agar dapat menentukan keputusan tempat sampah mana saja yang akan diambil oleh petugas kebersihan dalam bentuk rute yang akan ditampilkan pada aplikasi Blynk. Pada proses seleksi juga, terdapat 32 kemungkinan jalur rute yang akan dibuat secara otomatis sesuai dengan lokasi tempat sampah mana saja yang sudah dinyatakan penuh. Dengan mengambil data dari *realtime database* pada Firebase, sistem hanya akan mengambil data dari tempat sampah yang keterisiannya diatas 80% saja (Styawan, 2019) yang akan digunakan untuk menentukan rute tempat sampah yang harus dilalui secara otomatis. Selain itu, data keterisian tempat sampah akan ditampilkan pada aplikasi Blynk dengan menampilkan informasi keterisian tempat sampah dengan tampilan *virtual gauge*. Selain itu. pada aplikasi Blynk juga dapat menampilkan rute tempat sampah yang harus dilalui oleh petugas kebersihan yang disajikan dalam bentuk virtual LCD display sehingga hanya mengambil tempat sampah yang sudah penuh saja. Setelah itu, dilakukan proses perhitungan waktu *delay* pengiriman data dari saat sistem mendapatkan data dari Firebase hingga berhasil menampilkan informasi pada aplikasi Blynk baik informasi keterisian tempat sampah pada *virtual gauge* maupun membuat rute tempat sampah yang harus dilalui melalui tampilan *virtual LCD display*.

### 3.3. Alur Kerja *Software* Pada *Prototype*

Pada pembuatan *prototype* tempat sampah pintar, terdapat beberapa langkah yang dilakukan pada *software* Arduino IDE untuk memasukan kode program ke *board* Node MCU ESP 8266 agar *hardware* pada rangkaian *prototype* dapat berfungsi dengan semestinya. adapun alur kerja dari *software* tersebut digambarkan dalam *flowchart* pada Gambar 3.3.





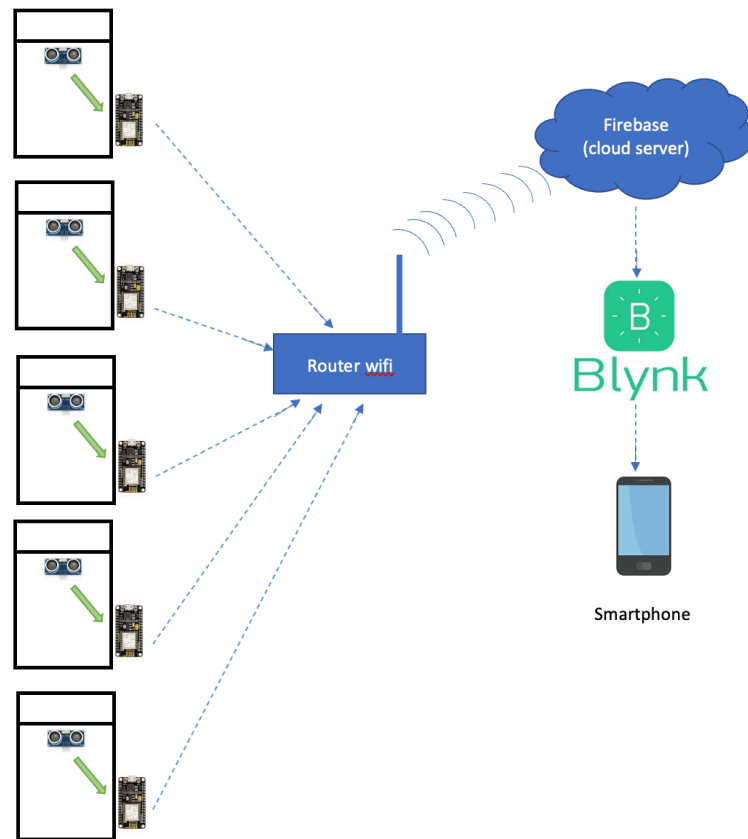
Gambar 3.3. *Flowchart* Alur Kerja *Software* pada *Prototype*

Pada *flowchart* yang digambarkan pada Gambar 3.3, Tahap pertama yang dilakukan dalam software Arduino IDE yaitu menambahkan *library* khusus untuk *board* Node MCU ESP 8266, aplikasi Blynk dan Firebase agar kode program pada Arduino IDE dapat dimasukkan ke dalam *board* Node MCU ESP 8266 dan aplikasi Blynk. kemudian, dilakukan penulisan kode program pada Arduino IDE agar *hardware* dapat berfungsi sesuai dengan perintah yang telah diberikan melalui kode program. Setelah melakukan penulisan kode program pada Arduino IDE, dilakukan proses *compiling* untuk memeriksa apakah kode program pada Arduino IDE sudah benar atau belum. Jika proses *compiling* gagal, maka dilakukan penulisan ulang pada kode program hingga proses *compiling* sudah dinyatakan berhasil. Jika proses *compiling* berhasil, maka akan muncul pesan "*Done compiling*" pada Arduino IDE. Jika sudah berhasil, maka tahap selanjutnya yaitu menyimpan *file* kode program yang sudah dibuat pada komputer agar dapat digunakan kembali. kemudian, dilakukan proses mengkoneksikan Node MCU ESP 8266 ke komputer melalui kabel USB agar kode program yang sudah dibuat sebelumnya dapat diupload ke *board* tersebut. setelah itu, dilakukan proses upload kode program dari Arduino IDE ke Node MCU ESP 8266. Jika proses gagal, maka akan muncul pesan error pada Arduino Uno dan harus melakukan proses upload ulang. Jika proses sudah berhasil, maka kode program dapat diupload pada Node MCU ESP 8266 hingga Node MCU ESP 8266 tersebut dapat menjalankan fungsinya dengan baik.

### 3.4. Rancangan *Prototype*

Pada penelitian ini, rancangan diperuntukan untuk gedung bertingkat yang memiliki lima lantai. Dimana, satu buah *prototype* tempat sampah pintar

mewakiliakan satu lantai pada gedung bertingkat. Pada kasus ini, terdapat satu orang petugas kebersihan yang bertugas untuk membersihkan lima buah tempat sampah dari lima lantai yang berbeda pada gedung bertingkat. Rancangan tersebut diuraikan pada Gambar 3.4.



Gambar 3.4. Ilustrasi Rancangan *Prototype* Tempat Sampah Otomatis untuk Gedung Lima Lantai.

Pada rancangan *prototype* ini, terdapat beberapa komponen yang digunakan untuk menciptakan *prototype* Sistem Tempat Sampah Otomatis pada Gedung Bertingkat Berbasis *Internet of Things* (IoT) baik dalam bentuk *hardware* maupun *software*. Untuk *hardware*, digunakan 5 buah sensor ultrasonik HC-SR04 dan lima buah node MCU ESP 8266. *Software* yang digunakan untuk rancangan ini yaitu Firebase yang digunakan sebagai *realtime database* untuk menyimpan data keterisian dari setiap tempat sampah dan



aplikasi Blynk yang digunakan untuk menampilkan informasi keterisian tempat sampah dan rute lantai tempat sampah yang harus dilalui melalui tampilan virtual. Pada rancangan *prototype*, semua komponen dan *software* terhubung melalui jaringan *wifi* yang disediakan oleh *router wifi*. Pada setiap tempat sampah, sensor ultrasonik HC-SR04 akan mengukur keterisian tempat sampah yang kemudian akan dikonversi secara otomatis dari satuan sentimeter (cm) menjadi bentuk persentase dengan skala 0%-100%. Kemudian, data keterisian tempat sampah ini akan dikirimkan oleh Node MCU ESP 8266 melalui jaringan *wifi* menuju Firebase untuk disimpan. Setelah itu, data keterisian seluruh tempat sampah akan digunakan untuk melakukan proses seleksi. Pada proses seleksi, sistem hanya akan menyeleksi tempat sampah yang sudah terisi diatas 80% saja dimana data ini akan digunakan untuk menentukan jalur yang harus diambil oleh petugas kebersihan untuk membersihkan tempat sampah. Kemudian, data keterisian tempat sampah dari setiap lantai akan ditampilkan pada aplikasi Blynk melalui tampilan *virtual gauge* dengan persentase keterisian tempat sampah yang tertera pada tampilan tersebut. Selain itu, pada aplikasi Blynk juga menampilkan rute yang harus dilalui oleh petugas kebersihan untuk membersihkan tempat sampah melalui tampilan *virtual LCD* pada aplikasi Blynk. Aplikasi Blynk dapat diakses dengan mudah oleh petugas kebersihan melalui *smartphone* sehingga petugas kebersihan dapat dengan mudah *memonitoring* keterisian tempat sampah dan membersihkannya hanya sesuai jalur yang tertera pada *virtual LCD* pada aplikasi Blynk.