

BAB I

PENDAHULUAN

Pada bab ini berisi latar belakang dari dibuatnya penelitian dengan judul “Pengembangan Platform IoT Cerdas Dengan *Trigger Action Programming* dan Canvas API Berbasis Android”. Pembahasan dari bab ini mencakup latar belakang, rumusan masalah, tujuan, batasan masalah dan manfaat dari penelitian ini. Pada latar belakang dijelaskan alasan yang mendasari pembuatan penelitian ini, pada rumusan masalah dijelaskan masalah yang ingin diselesaikan pada tujuan penelitian ini, pada tujuan penelitian dijelaskan tujuan yang ingin dicapai pada penelitian ini dan manfaat menjelaskan manfaat yang akan diberikan oleh penelitian ini.

1.1. Latar Belakang

Internet of things atau sering disebut IoT adalah sebuah teknologi yang dapat menghubungkan satu benda dengan benda yang lain melalui internet (Santo et al, 2022). IoT cukup masif digunakan saat ini karena dukungan perkembangan teknologi informasi yang semakin pesat (Santo et al, 2022). Berbagai permasalahan masyarakat saat ini banyak yang telah diselesaikan dengan memanfaatkan teknologi IoT seperti permasalahan dibidang pertanian dan peternakan (Susanto et al, 2022). IoT digunakan karena efisiensi dalam penggunaannya, pengguna dapat memonitor dan mengendalikan perangkat tertentu dari jarak jauh melalui internet dengan mudah tanpa mengeluarkan banyak sumber daya. Oleh karena itu semua membuat pemanfaatan teknologi IoT saat ini semakin luas pada segala bidang (Abdur et al, 2022).

Pemanfaatan IoT dapat dipadukan dengan berbagai bidang disiplin ilmu salah satunya *machine learning*. Beberapa aplikasi IoT yang telah mengimplementasikan *machine learning* saat salah satunya adalah (Liu et al, 2020) penelitian tersebut menggunakan *machine learning* untuk melakukan prediksi anomali iklim dalam implementasi *vertical plant wall* di dalam ruangan menggunakan algoritma *machine learning*. Pada penelitian yang dirujuk algoritma *machine learning* digunakan untuk mendeteksi anomali nilai gas CO₂ dan suhu dalam ruangan,

pemanfaatan *machine learning* dan IoT pada penelitian tersebut membuktikan bahwa IoT dapat dikombinasikan dengan *machine learning* dengan baik.

Cognitive IoT adalah kombinasi dari IoT dan *machine learning* sehingga IoT dapat melakukan pengambilan keputusan secara cerdas tanpa intervensi manusia (Alipio, 2020) *Cognitive IoT* berasal dari *cognitive computing*, *cognitive computing* sendiri adalah salah satu cabang dalam bidang ilmu kecerdasan buatan yang berusaha untuk membuat komputer memiliki kemampuan bernalar selayaknya manusia, *cognitive computing* menggabungkan berbagai bidang ilmu seperti *natural language processing*, *machine learning*, *image recognition* dan masih banyak lagi. *Cognitive IoT* adalah lanjutan dari IoT di mana ekosistem IoT dipadukan dengan kemampuan bernalar berdasarkan data yang didapat dari berbagai perangkat IoT untuk selanjutnya diolah menjadi sebuah keputusan cerdas berbasis data yang telah dikumpulkan sebelumnya (Shrivastava et al, 2023).

Pengembangan *cognitive IoT* ini cukup memakan waktu karena pengguna harus melakukan pengembangan *machine learning* dan juga pengembangan perangkat IoT keduanya secara bersamaan (Alipio, 2020). Pengembangan *machine learning* dilakukan untuk dapat melakukan prediksi pada data, pengembangan ini mencakup proses-proses pengembangan sebuah model seperti *data collection*, *preprocessing*, *training* model, *evaluation* sedangkan pengembangan IoT dilakukan untuk mendapatkan data dari berbagai perangkat elektronik, pengembangan IoT mencakup pengembangan perangkat elektronik, menghubungkan perangkat elektronik dengan MQTT broker, menghubungkan data dari perangkat IoT untuk disajikan dalam bentuk *dashboard* yang dapat dilihat oleh pengguna. Saat *machine learning* dan IoT selesai dikembangkan pengguna harus menghubungkan model *machine learning* dan IoT. Menghubungkan model dengan IoT dapat dilakukan dengan *deployment* model *machine learning* pada perangkat yang sama dengan *dashboard* IoT seperti Web dan Android atau dapat juga dilakukan *deployment* pada sisi server. Banyaknya langkah tadi membuat pengembangan solusi cerdas berbasis IoT lebih lama oleh karena itu dibutuhkan automasi integrasi antara *machine learning* dan IoT untuk memangkas langkah-langkah dari pembuatan solusi cerdas berbasis IoT tadi.

1.2. Rumusan Masalah

Dari latar belakang yang sudah dijelaskan, didapati rumusan masalah yang merupakan dasar dilakukannya penelitian ini:

1. Bagaimana merancang dan membuat platform IoT berbasis android ?
2. Bagaimana integrasi *machine learning* pada platform IoT berbasis Android ?
3. Bagaimana merancang dan membuat sistem dapat menangani banyak model?
4. Bagaimana *trigger action programming* diimplementasikan pada platform IoT berbasis Android untuk mengendalikan perangkat IoT ?

1.3. Tujuan Penelitian

Adapun tujuan yang hendak dicapai pada penelitian ini berdasarkan rumusan masalah yang telah dirumuskan yaitu :

1. Merancang dan membuat platform IoT berbasis Android.
2. Mengintegrasikan *machine learning* pada platform IoT berbasis Android.
3. Merancang dan membuat sistem dapat menangani banyak model.
4. Mengimplementasikan *trigger action programming* pada platform IoT berbasis Android untuk mengendalikan perangkat IoT.

1.4. Manfaat Penelitian

Penelitian ini memiliki beberapa manfaat untuk berbagai pihak, seperti untuk penelitian ke depannya dan pengguna, berikut manfaat dari penelitian ini yaitu:

1. Tersedianya referensi penelitian tambahan terkait pengembangan aplikasi berbasis android yang memanfaatkan Canvas API dan *trigger action programming*.
2. Tersedianya referensi implementasi otomatisasi untuk integrasi model *machine learning* pada platform IoT berbasis Android yang dapat dipelajari dan dikembangkan oleh semua orang dikarenakan sifat kode sumbernya yang terbuka.
3. Tersedianya alternatif aplikasi IoT berbasis Android yang menyediakan cara baru untuk mengimplementasikan solusi cerdas pada IoT dengan lebih mudah.

1.5. Batasan Masalah

Untuk menghindari pembahasan yang terlalu luas, berikut adalah batasan-batasan yang ditetapkan oleh penulis dalam penelitian ini:

1. Aplikasi terdiri atas 2 bagian, yaitu bagian *backend* dan *frontend*.
2. *Stack* teknologi yang dipilih adalah sebagai berikut:
 - *Frontend*: Kotlin;
 - *Backend*: Nest.js, FastApi;
 - Basis data: PostgreSQL;
3. Kode sumber aplikasi bersifat terbuka dan dapat diakses melalui platform GitHub.
4. Hanya terdapat 4 komponen platform IoT yaitu *switch*, *line graph*, *gauge*, dan *text*.
5. Pengguna hanya dapat memasukkan model yang memiliki format .sav, .h5.
6. Aplikasi ini hanya terdapat satu jenis pengguna yaitu pengguna umum tidak memiliki pengguna dengan jenis administrator.
7. Sistem operasi Android yang dapat menggunakan aplikasi ini hanya yang memiliki versi 7.0 ke atas.