P-ISSN2614-5499
E-ISSN2614-5164

Perancangan Sistem Cerdas untuk Keamanan dan Pemantauan Pintu Rumah Berbasis IoT

Marcellus Denta Widyapramana[1], Gunawan Dewantoro[2], dan Handoko[3]
[1,2,3]Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga, Jawa Tengah 50711
e-mail: [1]denta.marcellus@gmail.com, [2]gunawan.dewantoro@staff.uksw.edu, [3]handoko@uksw.edu

***Abstrak*—Angka pencurian dan kehilangan barang yang terjadi di dalam rumah masih terbilang tinggi dan belakangan semakin meningkat. Hal ini disebabkan oleh banyak faktor, baik dari faktor kejahatan atau faktor kelalaian manusia. Tentu kejadian ini sangat merugikan dan dibutuhkan peningkatan keamanan dalam rumah. Perangkat pintar dapat memudahkan manusia dalam banyak hal termasuk peningkatan keamanan. Hal ini bisa dilakukan dengan adanya konsep *Internet of Things* dan *Machine Learning*. Pada penelitian ini dirancang sebuah sistem keamanan yang terdiri dari aplikasi Android dan perangkat keras yang menggunakan mikrokontroler untuk menghubungkan berbagai sensor dan aktuator. Tiap perangkat tersebut merupakan node dalam konsep *Internet of Things* dan dikoneksikan ke *smartphone* dengan internet. Hasil pengujian menunjukkan bahwa kontrol dan pemantauan akses pintu yang dipasang perangkat keras dapat dilakukan dengan menjalankan perintah Unlock atau Lock untuk mengendalikan solenoid pengunci baik secara manual atau otomatis melalui *set time* serta *set date*. Fitur GPS digunakan untuk mengakses kunci melalui posisi pengguna dan fitur NFC *smartphone* berfungsi sebagai proteksi tambahan untuk mengakses pintu. Semua data dikomunikasikan melalui Internet menggunakan protokol *Message Queuing Telemetry Transport* (MQTT) dan akan disimpan untuk kebutuhan pemantauan serta digunakan untuk data training dalam mempelajari pola kebiasaan penggunaan pintu.**

Kata kunci: **Keamanan, Internet of Things, Machine Learning, Message Queuing Telemetry Transport, NFC**

***Abstract*— The rate of robbery and lost goods in residential places is still high yet increasing recently. It is due to many factors, including crime factors and owner's ignorance. This is definitely undesirable and need some improvements in terms of home security. Smart devices can assist human in many aspects including home security. This can be done by means of Internet of Things and Machine Learning. This study designed a security system that consisted of Android application and device which use microcontroller to connect sensors and actuators. Each device served as node in the concept of Internet of Thing and be connected to smartphone through internet. The results show that control and monitoring can be conducted by executing Unlock or Lock command for actuating lock solenoid both manually and automatically by setting time and date. The GPS feature was used to actuate the key by detecting user's position and NFC smartphone feature served as additional protection. Each data is transferred through internet by means of Message Queuing Telemetry Transport (MQTT) protocol and will be saved for monitoring and also used as training data to learn the habit pattern of accessing door.**

Keywords: **Security, Internet of Things, Machine Learning, Message Queuing Telemetry Transport, NFC**

I. PENDAHULUAN

Sistem keamanan rumah konvensional masih sering dijumpai kekurangan sehingga tidak sedikit rumah mengalami pencurian karena kelalaian pemilik rumah sendiri atau dibobol oleh pencuri. Salah satu solusi untuk masalah ini adalah penggunaan konsep *Internet of Things* (IoT) yang makin marak penggunaannya di masa sekarang.

Berbagai penelitian yang memanfaatkan teknologi dan tergolong terjangkau telah dilakukan untuk meningkatkan keamanan di dalam rumah. Seperti pada penelitian sebelumnya yaitu sebuah implementasi penggunaan teknologi NFC untuk memberi akses membuka dan menutup pintu dan menyimpan data penggunaan pintu. Basis data pengguna dapat disimpan di Arduino [1] - [2], ataupun web server [3]. Teknologi RFID di dalam E-KTP juga telah dimanfaatkan sebagai kunci akses untuk keamanan pintu [4]. Pada penelitian [5], ditambahkan kunci akses tambahan sehingga terdapat pilihan kunci berbasis RFID atau bluetooth. IoT juga telah diintegrasikan dalam sistem keamanan pintu untuk meningkatkan kemudahan penggunanya. Sistem keamanan pintu menggunakan sensor magnetik dan *Passive Infra Red* (PIR) untuk memantau rumah pengguna. Jika sensor-sensor mendeteksi, maka pemilik rumah mendapatkan notifikasi ke aplikasi ponsel [6], ataupun ke platform IoT seperti Blynk dan Thingspeak [7]. Hasil notifikasi juga dapat berupa foto yang dikirimkan oleh Raspberry Pi ke Telegram Messenger [8]. Selain untuk pengiriman notifikasi, IoT juga dapat berguna untuk mengunci dan membuka kunci pintu. Pada penelitian [9], digunakan aplikasi Blynk untuk memantau kondisi pintu dan juga dapat untuk membuka dan mengunci pintu secara manual menggunakan interface yang dibuat pada Blynk.

P-ISSN2614-5499
E-ISSN2614-5164

Selain menggunakan platform IoT, *web database* juga dapat digunakan untuk menyimpan hasil pemantauan berupa gambar [10].

Pada penelitian yang telah dilakukan, masih dapat disisipkan kecerdasan dengan memanfaatkan *machine learning* untuk membuat sistem keamanan menjadi sistem cerdas yang mampu mempelajari pola tertentu untuk meningkatkan kemanan dan kenyamanan pengguna pintu. Pada penelitian ini, dibuat sebuah alat dan aplikasi berbasis IoT sebagai penyusun utama sistem keamanan pintu rumah dalam upaya peningkatan keamanan dan kenyamanan pemilik rumah. Hal ini dilakukan dengan memanfaatkan teknologi NFC yang sekarang sudah terpasang juga pada *smartphone*, dan juga kunci sekunder dengan memanfaatkan GPS pada *smartphone*. Selain itu, pengguna dapat membuka dan mengunci pintu secara manual dengan memanfaatkan aplikasi yang dibangun pada smartphone. Ditambah lagi dengan adanya *machine learning* yang akan mempelajari pola kebiasaan penggunan pintu dapat meningkatkan kenyamanan pemilik rumah dan menjadikan perangkat menjadi *smart device*.

II. STUDI PUSTAKA

A. *Internet of Things* (IoT)

IoT merupakan sebuah konsep di mana suatu objek (barang fisik) memiliki kemampuan untuk berkomunikasi melalui jaringan internet tanpa memerlukan bantuan interaksi antar manusia atau manusia ke komputer. Karena konsep IoT sendiri memerlukan internet untuk melakukan transfer data, maka objek tersebut perlu memiliki perangkat sehingga bisa terkoneksi [11]. IoT terbentuk dari 3 elemen dasar yaitu:

1. Perangkat keras, yang memiliki modul-modul koneksi ke jaringan Internet.
2. Jaringan internet, untuk media komunikasi data.
3. *Cloud server service*, atau layanan server awan sebagai tempat data disimpan dan tempat pertukaran data.

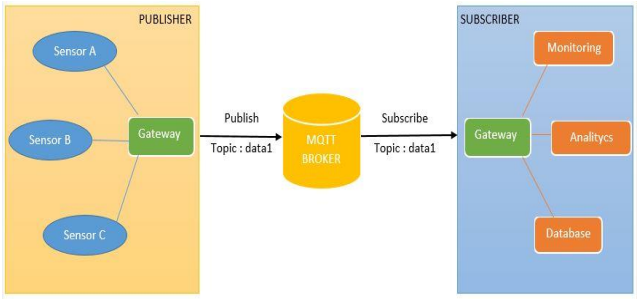
Tiap perangkat keras (disebut *node*) akan terhubung ke jaringan Internet. *Node* juga tersambung ke *cloud server* di mana data akan disimpan. Data tersebut dapat diakses dan diproses melalui komputer atau *smart phone*.

B. *Protokol MQTT*

Message Queuing Telemetry Transport (MQTT) adalah protokol yang berjalan di atas *stack* TCP/IP. Protokol ini dirancang untuk *machine to machine* yang tidak memiliki alamat khusus, seperti Arduino dan Raspberry, serta cukup mengirim nilai *string*. MQTT yang menerapkan sistem kerja *Publish-Subscribe* data, memiliki 4 komponen utama dalam menjalankan proses pengiriman dan penerimaan data seperti berikut:

1. *Broker*, sebagai server yang memiliki alamat IP khusus yang bertugas untuk mengatur data *publish-subscribe* dari device.
2. *Publish*, merupakan cara perangkat mengirimkan data ke *broker* dan akan diteruskan ke *subscriber*.
3. *Subscribe*, merupakan cara suatu perangkat menerima data dari *publisher* melalui *broker*.
4. *Topic*, berfungsi untuk pengelompokan data berdasarkan kategori tertentu. *Topic* tertentu wajib dimiliki untuk pertukaran data.

Gambar 1 menjelaskan implementasi dari penggunaan protokol dan komponen-komponen MQTT [12].

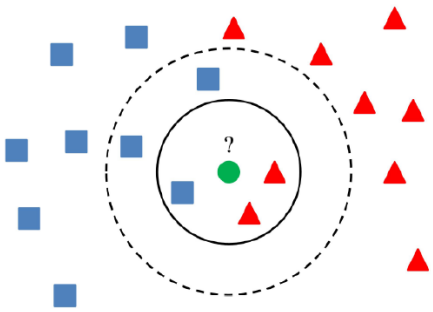


Gambar 1. Implementasi protokol MQTT

C. *K-Nearest Neighbor (KNN)*

KNN adalah suatu metode dalam *machine learning* yang termasuk dalam jenis *supervised learning* di mana tipe pembelajaran yang memiliki variabel input dan variabel output, yang menggunakan satu algoritma atau lebih untuk mempelajari relasi antara input dan output [13]. Tipe model ini dapat memperkirakan fungsi pemetaannya, sehingga ketika memiliki input baru kita dapat memprediksi output untuk input tersebut. Penjelasan singkat KNN terdapat pada Gambar 2.

Gambar 2. Implementasi algortima K-Nearest Neighbor



Algoritma KNN berkerja berdasarkan jarak terpendek *query instant* ke *training sample* agar mendapatkan nilai KNN-nya. *Training sample* akan diproyeksikan ke ruang berdimensi banyak, di mana ruang dikelompokkan berdasarkan klasifikasi *training sample* dan tiap dimensi merepresentasikan variabel dari data. Dekat dan jauhnya jarak biasanya dihitung menggunakan *euclidian distance*. Berikut beberapa langkah yang digunakan dalam metode KNN:

1. Menentukan parameter *K* atau jumlah tetangga terdekat.
2. Hitung kuadrat jarak *euclid* tiap objek terhadap data *sample* yang dipakai.
3. Mengelompokkan objek pada kelompok dengan jarak terpendek.
4. Kumpulkan *Y* (klasifikasi *nearest neighbor*).
5. Dengan kategori *nearest neighbor* terbanyak, dapat diprediksi nilai *query instant* yang telah dihitung.

Model KNN diprogram menggunakan library Sklearn atau Scikit-learn yang membantu dalam *training* serta *processing* dan ditulis dengan bahasa Python.

P-ISSN2614-5499
E-ISSN2614-5164

D. Wemos D1 Mini

ESP8266 adalah sebuah perangkat tambahan (modul) yang berfungsi sebagai sarana komunikasi dan kontrol melalui Internet. Perangkat ini dapat digunakan secara s*tandalone* atau digabungkan dengan mikrokontroler lain untuk membuat koneksi TCP/IP. Tetapi di pasaran juga sudah ditemukan ESP8266 yang sudah tertanam pada mikrokontroler, contohnya ESP12-E. Chip ESP12-E ini juga banyak bentuk integrasinya untuk menambah variasi kanal dan fitur untuk menambah keringkasan atau fungsi contohnya Wemos D1, NodeMcu dan jenis Arduino ESP. Tabel 1 menunjukkan spesifikasi Wemos D1 mini yang digunakan pada penelitian ini.

Tabel 1. Spesifikasi Wemos D1 mini

1	Wifi Modul	ESP8266 (didalam chip)
2	Prosesor	Tensilica 32bit RISC CPU Xtensa LX106
3	Tegangan Kerja	3,3 V
4	Tegangan Masukan	7-12V
5	Digital Pin	11
6	Analog Pin	1
7	Memory Flash	4 Mb
8	SRAM	64 kB
9	Kecepatan Clock	80 MHz ~ 160 MHz

E. Near Field Communication Reader (PN532)

Near Field Communication (NFC) merupakan teknologi *wireless* jarak pendek yang memungkinkan *smart phone* untuk berkomunikasi dan bertukar data. Penggunaan NFC dapat meningkatkan keamanan dan kenyamanan akses bertukar data.

NFC bekerja pada frekuensi 13,56 MHz dan memiliki kecepatan dalam transfer data sampai 424 kbit/s dengan perkiraan jarak sekitar 10 cm dan memiliki cara kerja yang hampir sama dengan RFID, serta komunikasi antara device yang memiliki NFC dapat berupa aktif-pasif dan *peer-to-peer*. Pada komunikasi aktif-pasif seperti pada NFC *smartphone* menggunakan sinyal pembawa satu arah 13,56 MHz dari perangkat *polling* sebagai sumber energi. Sedangkan pada *peer-to-peer* kedua arah dimodulasi dan diberi kode seperti perangkat *polling*, tetapi dalam sistem *peer-to-peer* daya yang digunakan lebih sedikit karena kedua perangkat menggunakan catu daya sendiri dan sinyal pembawa dimatikan setelah transmisi berakhir. Spesifikasi NFC PN532 ditunjukkan oleh Tabel 2.

Dengan memanfaatkan medan elektromagnet NFC dapat melakukan pertukaran data. Konsep yang digunakan mirip dengan prinsip transformator. Medan elektromagnet di antara perangkat yang memiliki koil konduktor berfungsi untuk memasangkan perangkat pengirim *(inisiator)* dan perangkat penerima (*target*). Data tersebut akan diubah menjadi bentuk biner dan dikirim berupa sinyal digital oleh koil. Data dikirim secara serial dengan teknik modulasi, di mana data tersebut akan ditumpangi dengan sinyal modulasi agar tidak mengalami gangguan.

Tabel 2. Spesifikasi PN532

1	Komunikasi yang didukung	SPI, I^2C dan UART
2	Tegangan kerja	2,7–5,5 V
3	RAM/ROM	1 kB/40 kB
4	Frekuensi kerja	13,56 MHz

F. Real Time Clock (DS3231)

Salah satu penyebab sering terjadinya pembobolan dan pencurian adalah tidak adanya *record* data yang jelas waktu pintu tersebut digunakan atau terbuka. Sehingga tidak ada yang tau secara pasti akses pintu tersebut digunakan. Untuk itu dalam sistem ini memakai DS3231 untuk memberi parameter data waktu dan tanggal pintu tersebut digunakan.

DS3231 memiliki catu daya cadangan sendiri sehingga apabila catu daya dari sumber utama putus, maka data pada modul ini akan tetap berjalan dan tersimpan serta tidak akan mencatat dari awal lagi. Selain itu DS3231 dilengkapi dengan osilator kristal eksternal yang membuat keakuratan data terjamin. DS3231 ini menggunakan pin SDA/SCL untuk komunikasi mengirim data dan kerja pada tegangan 5 V.

G. Sensor Getaran (SW420)

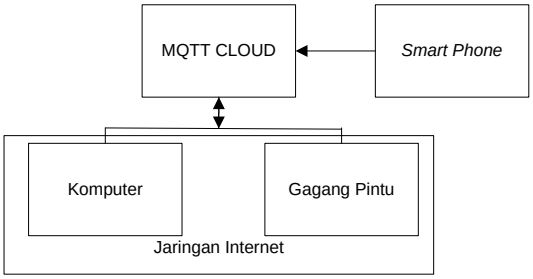
Sensor getaran akan digunakan untuk mendeteksi gataran yang dialami oleh pintu. Dengan memanfaatkan getaran yang mempengaruhi mekanisme sensor getaran dan menghasilkan perbedaan tegangan, sinyal tersebut selanjutnya dibandingkan dengan tegangan referensi oleh komparator untuk menghasilkan sinyal digital. Tegangan kerja sensor ini adalah 5 V dan menggunakan pembacaan sinyal digital.

Dengan menghitung jumlah *Pulse-IN* atau jumlah nilai pulsa tinggi yang dihasilkan dari tiap getaran sensor, dapat digunakan untuk sebagai monitoring dan pemberitahuan jika terjadi pembobolan paksa pada pintu.

III. METODE

A. Diagram Kerja Sistem

Dalam sistem yang dirancang terdiri dari minimal tiga buah *node* utama yaitu gagang pintu, sebuah komputer dan *smart phone*. Gambar 3 menunjukkan diagram blok sistem keseluruhan.



Gambar 3. Diagram blok sistem keseluruhan

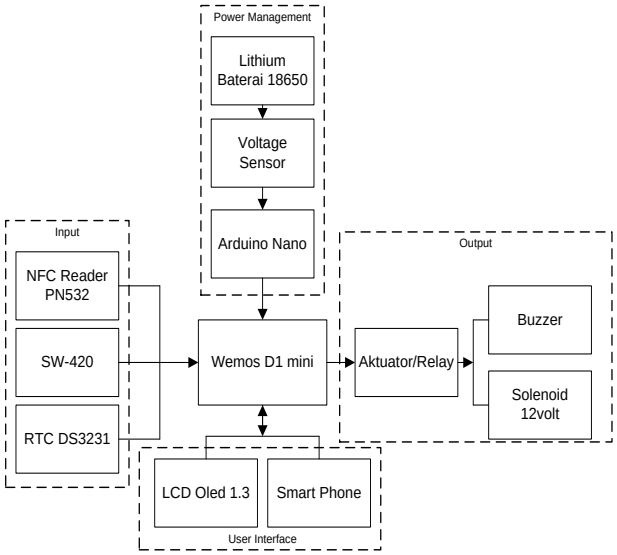
Dengan menggunakan jaringan WiFi, perangkat keras (gagang pintu) dan komputer akan terhubung ke internet lalu akan terhubung ke MQTT cloud sebagai sarana pertukaran data. Komputer di sini digunakan untuk *training* sistem mempelajari pola kebiasaan. Sedangkan *smartphone* akan terhubung langsung ke MQTT *cloud* untuk mengirim data dan menerima data.

P-ISSN2614-5499
E-ISSN2614-5164

B. *Perancangan Gagang Pintu*

Gagang pintu merupakan komponen utama dalam sistem keamanan terutama sebagai parameter kontrol, pemantauan maupun pola kebiasaan. Gagang pintu ini menggunakan sumber tegangan 12 VDC. Pusat pemrosesan data menggunakan Wemos D1 mini yang dipasang dengan sensor NFC *Reader* PN5322, sensor getar SW420 dan RTC DS3231, seperti ditunjukkan Gambar 4. Untuk output dari pemrosesan data sendiri menggunakan aktuator/relay untuk menghidupkan buzzer maupun solenoid 12 V. Selain itu juga terpasang LCD Oled sebagai *interface* tambahan pada gagang pintu. Diagram blok gagang pintu ini ditunjukkan oleh Gambar 5.

Gambar 4. Bagian alat pada gagang pintu



Gambar 5. Diagram blok gagang pintu

C. *Diagram Alur Perangkat Lunak*

Ketika perangkat keras memulai rutinitasnya, pertama-tama alat akan mengambil data inisialisasi terlebih dahulu. Dua proses inisialisasi terdiri dari dua macam yaitu inisialisasi komunikasi data yang berhubungan dengan sarana pertukaran data dan inisialisasi user yang berhubungan dengan *security permission* untuk user.

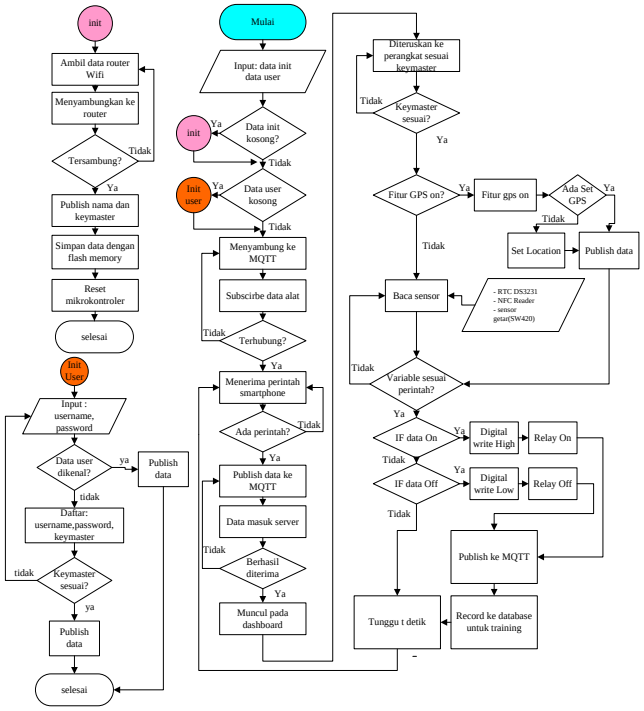
Dari kedua proses inisialisasi tersebut adalah agar perangkat dapat:

1. Mendapat koneksi internet dari router WiFi lokal yang terpasang pada rumah.
2. Menghubungkan ke MQTT *Cloud* untuk menjadi protokol pertukaran data.
3. Menentukan topik yang akan di-*subscribe*, untuk membedakan jenis data baik kontrol, monitoring, *keymaster*, *training*, GPS dan NFC.
4. Mengirim *keymaster* yang digunakan user untuk login ke aplikasi

Setelah proses inisialisasi selesai, alat akan terhubung ke internet dan alat terhubung ke *broker* serta melakukan *subscribe* untuk sinkronisasi data dengan *smartphone* yang digunakan. Setelah itu, perangkat akan menunggu perintah dari *smartphone*.

Setelah semua selesai, perangkat akan melakukan perulangan sesuai dengan rutinitasnya yaitu menerima perintah dan data dari *smartphone*, yang kemudian akan dibandingkan dengan data dari pembacaan sensor untuk menghasilkan perintah membuka atau menutup kunci secara manual maupun otomatis.

Perangkat juga akan menerima data *state* untuk mengaktifkan fitur-fitur mulai dari fitur GPS, NFC dan penggunaan pola kebiasaan. Semua data akses yang dikirim *smart phone* dan diterima oleh perangkat akan di-*publish* ke MQTT *cloud* dan disimpan pada database untuk pemantauan dan data *training*. Gambaran alur disajikan pada Gambar 6.



Gambar 6. Diagram alur perangkat lunak

D. *Aplikasi Android*

Aplikasi terdiri dari Inisialisasi, Kontrol dan Monitoring, Sebelum menggunakan aplikasi *user* perlu *login* terlebih dahulu. Setelah berhasil *login* dengan *user* dan *password* yang sudah terdaftar, akan dilanjutkan ke laman menu utama. Sebelum melakukan fitur kontrol dan monitoring *user* perlu memilih pintu yang akan digunakan yang telah didaftarkan sebelumnya dengan memasukkan *Keymaster* yang dimiliki gagang pintu.

Untuk fitur kontrol terdiri dari manual, otomatis, GPS, NFC dan *smart*. Untuk fitur manual pintu bisa dikunci dan dibuka secara langsung dengan menekan tombol pada aplikasi, untuk fitur otomatis perlu melakukan *set* data berupa jam, tanggal dan status dari pintu.

P-ISSN2614-5499
E-ISSN2614-5164

Pada fitur GPS *user* perlu melakukan *set* lokasi dahulu dengan cara berdiri pada titik untuk deteksi kemudian *user* melakuka *set* status fitur *On/Off*. Setelah selesai melakukan *set*, data pintu akan dapat terbuka selama 8 detik dan menutup kembali setelah user berada pada titik deteksi. Sedangkan untuk fitur NFC dapat digunakan setelah mengaktifkan melalui *Icon* NFC pada bagian menu utama aplikasi. Setelah diaktifkan perangkat keras akan mendeteksi NFC Tag pada *smarphone*.

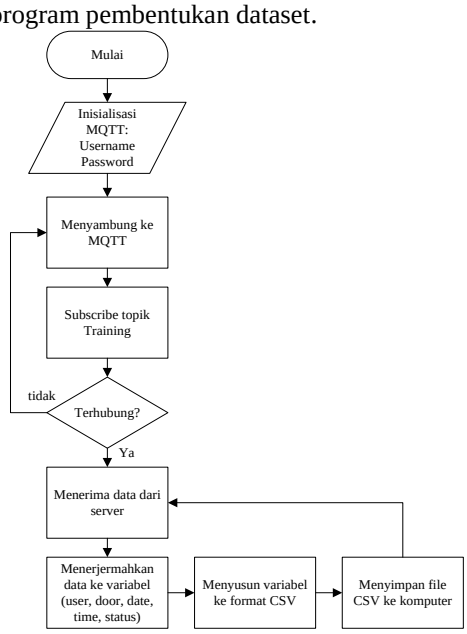
Untuk fitur monitoring menampilkan secara realtime kondisi pintu yang telah dikontrol. Poin yang ditampilkan berupa nama pintu yang dipilih, user yang mengakses, status pintu, waktu dan tanggal akses. Semua poin tersebut juga tersimpan dan terekam pada database untuk bisa diakses dan digunakan untuk *training* sistem.

Untuk fitur *smart* yang memprediksi kebiasaan pengguna pintu, user perlu mengaktifkannya terlebih dahulu di menu utama pada bagian *smart mode*. Pada mode ini user perlu memasukkan hari di mana mode *smart* akan diaktifkan dan status yang diaktifkan baik *Lock* atau *Unlock*. Setelah di-*Set* aplikasi akan mengirimkan variabel input berupa JSON file ke MQTT *Cloud* dan kemudian akan diterima komputer. Setelah itu komputer akan menghasilkan output yang merupakan hasil prediksi dari sistem yang sudah dilatih sebelumnya.

E. Pembuatan Dataset Training

Komponen utama untuk melatih sistem adalah *dataset*. Informasi *dataset* berisi data user, tanggal akses, waktu akses, nama pintu dan status pintu.

Data tersebut diterima oleh komputer dalam format JSON lalu diterjermahkan kemudian dimasukkan ke setiap variabel oleh komputer lalu disusun dan disimpan dalam format file *Comma Separated Values* (CSV). File inilah yang akan digunakan sebagai data *training* sistem dan semua pembuatan *dataset* menggunakan bahasa pemrograman Python. Gambar 7 menunjukkan alur program pembentukan dataset.



Gambar 7. Diagram alur program penyusun *dataset*

F. Prediksi Penggunaan Pintu

Untuk prediksi penggunaan pintu, komputer menggunakan metode KNN yang dibuat dengan bahasa pemrograman python dengan library Scikit-learn atau Sklearn. Lalu untuk proses normalisasi dan pengolahan *dataset* juga menggunakan library Pandas dan Numpy.

Sebelum digunakan, *dataset* yang berisi informasi akses pintu harus dinormalisasi terlebih dahulu agar stabil saat diproses lalu dibagi menjadi data latih dan data test. Dalam proses pelatihan, data latihlah yang akan diinputkan. Di sini akan dimasukkan nilai *K* terbaik untuk memperoleh error yang sangat kecil dan akurasi yang tinggi. Setelah itu akan digunakan data test untuk melihat hasil performa dengan model yang dipakai.

Setelah mendapat nilai *K*, akan dibuat model KNN dengan memasukkan nilai K terbaik yang memiliki tingkat error rendah. Jika dirasa model sudah cukup baik, dapat disimpan dalam sebuah file agar dapat langsung digunakan saat dijalankan tanpa harus melatih model lagi.

Ketika sistem sudah mengelompokkan pola pemakaian pintu maka sistem sudah dapat memprediksi pemakaian pintu dengan memasukkan beberapa variabel-variabel yang diperlukan seperti nama user, nama pintu, hari yang akan diprediksi dan status yang akan diprediksi. Variable tersebut diterima dari aplikasi pada pilihan *smart mode.*

IV. HASIL DAN PEMBAHASAN

A. Pengujian Perintah Aplikasi ke Alat

Pengujian pada Tabel 3 ini akan menguji pengiriman perintah dari aplikasi ke alat. Pengujian diawali dengan menyiapkan keseluruhan alat dan *smartphone* yang masing-masing sudah terkoneksi ke internet. Pengujian masing-masing dilakukan sebanyak 10 kali. Kemudian diukur persentase keberhasilan aplikasi dalam mengirim dan meneruskan perintah ke alat. Berikut tabel pengujian:

Tabel 3. Hasil pengujian perintah dari *smartphone* ke alat

Topik	Perintah	Value pada Server	Keberhasilan
kontrol/	Manual, pintu dikunci, oleh denta,bagian pintu utama	{“Date”:”10-10-2020”,”Door”:”pintu utama”,”Name”:”denta”,”Status”:”Locked”,”Time”:”1:37:30”}	100%
kontrol/	Manual, pintu dibuka, oleh denta,bagian pintu utama	{“Date”:”10-10-2020”,”Door”:”pintu utama”,”Name”:”denta”,”Status”:”Unlocked”,”Time”:”1:38:46”}	100%
kontrolA/	otomatis, pintu dibuka, oleh denta,bagian pintu utama, tanggal 10-10-2020 jam 3:3:1	{“Date”:”10-10-2020”,”Door”:”pintu utama”,”Name”:”denta”,”Status”:”Unlocked”,”Time”:”3:3:1”}	100%
kontrolA/	otomatis, pintu dikunci, oleh denta,bagian pintu utama, tanggal 10-10-2020 jam 3:25:1	{“Date”:”10-10-2020”,”Door”:”pintu utama”,”Name”:”denta”,”Status”:”Locked”,”Time”:”3:25:1”}	100%
NFC/	Fitur NFC, pintu dibuka, oleh denta,bagian pintu utama, tanggal 10-10-2020 jam 1:41:7	{“Date”:”10-10-2020”,”Door”:”pintu utama”,”Name”:”denta”,”Status”:”Unlocked”,”Time”:”1:41:7”}	100%
gpsSet/	Fitur NFC, pintu diaktifkan, oleh denta,bagian	{”DOOR”:”pintu utama”,”Latitude”:”110.8100679”,”Longitud	100%

	pintu utama, Latitude:110.8100679, Longitude:7.6038204	e":"7.6038204", "Name":"denta","Status":"Unlocked","Tipe":"SET"}	
gpsSet/	Fitur NFC, pintu diaktifkan, oleh denta,bagian pintu utama, Latitude:110.8082052, Longitude:7.6042482	{"DOOR":"pintu utama","Latitude":"110.8082052","Longitude":"7.6042482", "Name":"denta","Status":"Unlocked","Tipe":"Update"}	100%

Data dikirim ke server dalam format JSON file, dengan *Topic* yang berbeda untuk membedakan fungsi dari data. Data tersebut akan diteruskan dan diterjemahkan ke alat menjadi sebuah variabel.

B. Pengujian Deteksi GPS

Pengujian dilakukan untuk mengetahui kesesuaian nilai *latitude* dan *longitude* yang dibaca aplikasi, dikirim ke server dan yang diterima sebagi *pin point* pada perangkat keras. Nilai latitude dan longitude didapat dengan membaca sensor geolokasi pada *smartphone*, di mana nilai ini akan digunakan sebagai daerah deteksi membuka pintu.

Pada aplikasi		Pada server mqtt		Pada perangkat		Status
Longi-tude	Lati-tude	Longi-tude	Lati-tude	Longi-tude	Lati-tude	
-7,60603	110,8116767	-7,60603	110,8116767	-7,6060	110,8116	Set
-7,60422	110,8082183	-7,60422	110,8082183	-7,6042	110,8082	Update
-7,60436	110,8081792	-7,60437	110,8081792	-7,6043	110,8081	Update
-7,60459	110,8080401	-7,60459	110,8080401	-7,6045	110,8080	Update
-7,60642	110,8111245	-7,60642	110,8111245	-7,6064	110,8111	Update
-7,60633	110,8111441	-7,60633	110,8111441	-7,6063	110,8111	Update
-7,60652	110,8110569	-7,60652	110,8110569	-7,6065	110,8110	Update
-7,60698	110,8108582	-7,60698	110,8108582	-7,6069	110,8108	Update
-7,60723	110,8108552	-7,60723	110,8108552	-7,6072	110,8108	Update
-7,60735	110,8108537	-7,60735	110,8108537	-7,6073	110,8108	Update

Tabel 4 menunjukkan pengambilan data dan nilai latitude serta longitude yang dibaca pada aplikasi, server dan perangkat keras (COM6). Nilai Update diambil dengan berpindah tempat dari lokasi Set dengan masing-masing perpindahan berjarak ± 2 meter.

C. Pengujian Sensor Getar

Pengujian dilakukan dengan cara memberi goncangan pada pintu dan memberi goncangan pada gagang pintu, di mana getaran akan menggerakkan pelampung logam di dalam tabung yang berisi elektroda yang menghasilkan perbedaan tegangan. Berikut pengujian respon sensor saat disimulasikan pembukaan paksa. Untuk pengujian dilakukan simulasi pembobolan dengan mengacu pada nilai pembacaan sensor getar mulai dari alat dicoba dibongkar / dilepas paksa dan pintu saat didobrak. Pengujian dilakukan sebanyak 10 kali untuk deteksi. Berikut data pengujian.

Tabel 5. Pengujian sensor getar

Kondisi Pintu	Keberhasilan
Alat dicoba dibongkar / dilepas paksa	100%
Alat dicoba dibongkar / dilepas paksa	100%
Alat dicoba dibongkar / dilepas paksa	100%
Alat dicoba dibongkar / dilepas paksa	100%
Alat dicoba dibongkar / dilepas paksa	100%
Pintu saat didobrak	100%
Pintu saat didobrak	100%
Pintu saat didobrak	100%
Pintu saat didobrak	100%
Pintu saat didobrak	100%

Tabel 5 menjelaskan tingkat keberhasilan alat dalam mendeteksi percobaan pembobolan pintu dan mengirimkan notifikasi ke aplikasi Android serta membunyikan alarm.

D. Pengujian Prediksi Penggunaan

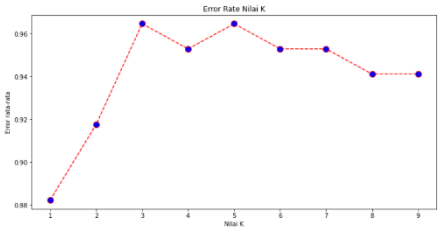
Pada pengujian, pertama-tama disediakan *dataset* yang sudah dibuat sebelumnya, *dataset* tersebut dinormalisasi dan dikelompokkan menjadi *input output* asli. *Input* ini berisi nama user, nama pintu, hari dan status pintu sedangkan *output* berisi waktu akses pintu. Lalu *input output* tadi dipisahkan masing-masing menjadi data tes sebanyak 20% dari jumlah *dataset* dan data latih sebanyak 80% dari jumlah *dataset*. Lalu untuk memperoleh hasil yang baik dan data agar bisa diproses, data diberi skala yang sama menggunakan *Standard Scaler* yang terdapat pada *library* Sklearn.

Dikarenakan agar data tidak membuat klasifikasi yang terlalu banyak, data latih untuk jam diubah menjadi rentang waktu. Hal ini dikarenakan satuan jam memiliki satuan yang sangat banyak jika digunakan untuk memprediksi tepat jam yang diinginkan akan dibutuhkan *class* untuk tiap jam, tiap detik dan tiap menit. Sehingga hal ini ditakutkan dapat mempengaruhi dalam akurasi dari model *Machine Learning* dan model dapat menjadi *overfitting* atau bisa disebut model yang dibuat terlalu fokus pada data *training* tertentu sehingga tidak bisa melakukan prediksi dengan tepat. Dataset waktu yang diubah menjadi rentang waktu memiliki rentang selama 1 jam.

Setelah *dataset* diproses, untuk mendapat model yang bagus perlu mencari nilai *K* yang merupakan tetangga terdekat yang digunakan untuk menentukan klasifikasi. Pada pengujian dicari model terbaik menggunakan data latih dengan mengganti nilai *K* dengan nilai *K* dari 1 sampai 10. Berikut nilai error rata-rata yang didapat.

P-ISSN2614-5499
E-ISSN2614-5164

Gambar 9. Grafik *Error Rate*



Dari Gambar 9 dapat dilihat nilai error paling rendah berada pada nilai *K*=1. Tetapi proses pencarian tidak dilakukan 1 kali saja, proses pencarian *K* dengan melihat model terbaik dilakukan sebanyak 10 kali. Berikut hasil pencarian nilai *K* dengan nilai error terendah:

Tabel 6. Nilai K dengan rata-rata error terendah

Percobaan	Nilai K error terendah	Nilai error
1	1	0,88
2	1	0,88
3	2	0,93
4	1 dan 2	0,85
5	2	0,9
6	2	0,99
7	1	0,8
8	1	0,88
9	1	0,83
10	1	0,88

Dari Tabel 6 dapat disimpulkan bahwa nilai *K* terbaik adalah 1 untuk melatih klasifikasi. Selanjutnya nilai *K* tadi akan digunakan untuk membuat dan melatih model baru dengan memasukkan nilai *K* menggunakan *input output* asli pada *dataset* tanpa dibagi menjadi data latih dan data tes.

Setelah membuat dan melatih model, model dapat disimpan dengan menggunakan *library* pickle pada komputer sehingga bisa diload dan digunakan tanpa dilatih lagi. Berikut hasil final pelatian model dengan menggunakan 2 user berbeda dan menggunakan pintu yang sama.

Tabel 7 terdapat informasi bahwa persentase *accuracy* dari model adalah sebanyak 0.21 atau sebanyak 21%. Hasil ini menggunakan *dataset* dari 2 user berbeda dengan waktu satu minggu dan hanya menggunakan 1 jenis pintu saja. Tetapi begitu menggunakan *dataset* dengan 4 user berbeda dengan jumlah pengambilan satu minggu serta jenis pintu yang sama memperloleh data sebesar 0.30 atau sekitar 30%.

Tabel 7. Hasil latih model 2 *user*

No	Precision	Recall	F1-Score	Support
0	0,00	0,00	0,00	1
1	0,00	0,00	0,00	1
2	0,12	0,31	0,18	13
3	0,33	0,32	0,33	34
4	0,23	0,058	0,36	19
5	0,14	0,04	0,06	27
6	0,25	0,10	0,14	21
7	0,16	0,29	0,21	14
8	0,14	0,04	0,06	28
9	0,15	0,57	0,24	7
10	0,25	0,09	0,13	22
11	0,18	0,27	0,22	15
12	0.00	0,00	0,00	20
13	0,00	0,00	0,00	14
14	0,00	0,00	0,00	2
Accuracy			0,21	238
Macro avg	0,13	0,19	0,13	238
Weighted avg	0,18	0,21	0,16	238

Pada Tabel 8 dapat disimpulkan bahwa semakin banyak variasi data dari *dataset* akan membuat akurasi model semakin tinggi dalam memprediksi *output* yang sesuai dengan pola pengelompokkan kebiasaan user.

Selanjutnya dilakukan pengujian untuk memprediksi kebiasaan user dengan memasukkan nama user, nama pintu, hari yang diprediksi dan status yang diprediksi. Pengujian dilakukan sebanyak 5 kali untuk tiga user untuk memprediksi kebiasaan dari tiap user pada hari tertentu dan status tertentu.

Tabel 8. Hasil latih model 4 *user*

No	Precision	Recall	F1-Score	Support
0	49	0,05694	0,04236	22
1	59	0,05486	0,04722	28
2	50	0,27	0,35	22
3	0,14	0,11	0,12	46
4	0,33	0,05	0,09	40
5	0,13	0,07	0,10	27
6	0,17	0,38	0,23	21
7	0,16	0,30	0,21	20
8	0,25	0,04375	0,35	49
9	0,33	0,31	0,32	13
10	0,00	0,00	0,00	28
11	0,32	0,21	0,25	39
12	0,14	0,05	0,07	20
13	1,00	0,18	0,30	17
14	0,50	0,04306	0,56	8
15	0,50	0,11	0,18	18
16	0,50	1,00	0,04653	3
Accuracy			0,3	421
Macro avg	0,36	0,35	0,3	421
Weighted avg	0,31	0,30	0,26	421

Tabel 9. Pengujian prediksi

User	Input	Hasil Prediksi	Data Pada *Record*	Keberhasilan
denta	denta, Pintu kamar, hari Senin, Unlock	07.00.00-08.00.00	07.00.00	Berhasil
denta	denta, Pintu kamar, hari Selasa, Unlock	20.00.00-21.00.00	20.00.00	Berhasil
denta	denta, Pintu kamar, hari Jumat, Lock	07.00.00-08.00.00	07.20.00	Berhasil
denta	denta, Pintu kamar, hari Sabtu, Lock	07.00.00-08.00.00	07.20.00	Berhasil
denta	denta, Pintu kamar, hari Minggu, Unlock	10.00.00-11.00.00	10.00.00	Berhasil
Andi	andi, Pintu kamar, Hari Senin,Unlock	07.00.00-08.00.00	07.15.00	Berhasil
Andi	andi, Pintu kamar, Hari Selasa,Unlock	07.00.00-08.00.00	07.15.00	Berhasil
Andi	andi, Pintu kamar, Hari Rabu,Lock	21.00.00-22.00.00	21.00.00	Berhasil
Andi	andi, Pintu kamar, Hari Jumat,Lock	21.00.00-22.00.00	21.00.00	Berhasil
Andi	andi, Pintu kamar, Hari Minggu,Unlock	10.00.00-09.00.00	10.15.00	Berhasil
Yoyo	Yoyo,Pintu kamar, hari Senin,Unlock	08.00.00-09.00.00	08.00.00	Berhasil
Yoyo	Yoyo,Pintu kamar, hari Rabu,Unlock	08.00.00-09.00.00	08.00.00	Berhasil
Yoyo	Yoyo,Pintu kamar, hari Kamis,Lock	10.00.00-11.00.00	10.15.00	Berhasil
Yoyo	Yoyo,Pintu kamar, hari Sabtu,Lock	10.00.00-11.00.00	10.15.00	Berhasil
Yoyo	Yoyo,Pintu kamar, hari Minggu,Unlock	17.00.00-18.00.00	17.00.00	Berhasil

Tabel 9 menunjukkan pengujian dilakukan dengan memasukkan beberapa inputan untuk diprediksi, dan data hasil prediksi dibandingkan dengan data *record* yang telah diambil sebelumnya. Data record ini berisi data yang paling sering muncul dan stabil dalam rentan waktu pengambilan data. Hasil prediksi didapat dengan kemiripan kekerabatan

P-ISSN2614-5499
E-ISSN2614-5164

data *input* dengan atribut data yang dimiliki *dataset* yang telah disusun.

V. KESIMPULAN

Keberhasilan GPS dapat dalam menentukan nilai *Latitude* dan *Longitude* sebagai Set lokasi maupun Update lokasi sebesar 80%. Kegagalan dalam fitur terdapat pada *lagging* update lokasi GPS sehingga ada beberapa lokasi yang tidak terbaca atau terlewat dan terdapat ralat error sebesar 2 meter. Keberhasilan alarm keamanan dapat berkerja dalam mendeteksi percobaan pembobolan melalui pembacaan sensor getar, dan mengirim notifikasi ke aplikasi sebesar 100%. Keberhasilan sensor NFC dalam membaca nilai UID (*Unique Identifier*) pada *smartphone* sebesar 100% dengan pengujian sebanyak 10 kali untuk tiap sensor.

Dari hasil *training, dataset* sebanyak 239 data, dengan 2 user berbeda memperoleh akurasi sebesar 21%, tetapi dengan *dataset* sebanyak 422 data dengan 4 user berbeda memperoleh akurasi sebesar 30%. Sehingga jumlah dan variasi variable dataset dapat mempengaruhi hasil klasifikasi. Dalam pengujian aplikasi untuk mengirim data dan perintah dari *smartphone* ke cloud MQTT serta alat dapat menerima perintah dengan baik mencapai 100% dengan total topik yang digunakan adalah 7 topik serta percobaan masing-masing 10 kali untuk tiap topik.

REFERENSI

[1] D. Kurnianto, E. S. Nugraha, dan V. K. Ekaristi, "Penerapan kartu elektronis berbasis near field communication (NFC) pada sistem keamanan pintu rumah cerdas," *Jurnal Infotel*, vol. 9, no. 1, pp.122-129, Februari 2017.

[2] Zulhelmi dan R. Nandika, "Perancangan kunci pintu menggunakan NFC ring dan password logger berbasis Arduino Mega 2560," *Sigma Teknika*, vol. 1, no. 2, pp. 139-148, November 2018.

[3] A. S. Djamar, S. R.U. A Sompie, dan M. D. Putro, "Implementasi teknologi NFC untuk akses pintu masuk dan keluar", *Jurnal Teknik Informatika*, vol.11, no. 1, 2017.

[4] W. Wendanto, D. N. R. Salim, dan D. W. T. Putra, "Rancang bangun sistem keamanan smart door lock menggunakan E-KTP (Elektronik Kartu Tanda Penduduk) dan personal identification number berbasis Arduino Mega R3," *GO INFOTECH: Jurnal Ilmiah STMIK AUB*, vol. 25, no. 2, pp. 133-142, Desember 2019.

[5] S. Mulyati dan S. Sadi, "IoT pada prototipe control keamanan pintu berbasis RFID dan Bluetooth," *Jurnal Teknik: Universitas Muhammadiyah Tangerang*, vol. 8, no. 2, pp. 9-14, Juli-Desember 2019.

[6] W. Kurniasih, A. Rakhman, dan I. Salamah, "Sistem keamanan pintu dan jendela rumah berbasis IoT," *Jurnal Riset Sistem Informasi Dan Teknik Informatika*, vol. 5, no. 2, pp. 266-274, Agustus 2020.

[7] J. Waworundeng, L. D. Irawan, dan C. A. Pangalila, "Implementasi sensor PIR sebagai pendeteksi gerakan untuk sistem keamanan rumah menggunakan platform IoT," *CogiTo Smart Journal*, vol. 3, no. 2, Desember 2017.

[8] M. I. Kurniawan, U. S, dan R. Tulloh, "Internet of Things: Sistem kemanan rumah berbasis Raspberry Pi dan Telegram Messenger," *ELKOMIKA*, vol. 6, no. 1, pp. 1-15, Januari 2018.

[9] Arafat, "Sistem pengamanan pintu rumah berbasis Internet of Things (IoT) dengan ESP8266," *Technologia*, vol. 7, no. 4, pp. 262-268, Oktober – Desember 2016.

[10] I. T. Putra, W. K. Raharja, dan M. Karjadi, "Push button sistem keamanan pintu rumah menggunakan Raspberry Pi berbasis IoT," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 23, no. 3, pp. 166-176, Desember 2018

[11] A. Junaidi, "Internet of Things, sejarah, teknologi, dan penerapannya: Review," *Jurnal Ilmiah Teknologi Informasi Terapan*, vol. 1, no. 3, pp. 62-66, Agustus 2015.

[12] Mengenal MQTT Protokol untuk IoT [Online]. Available: http://reslab.sk.fti.unand.ac.id/index.php?option=com_k2&view=item&id=229:mengenal-mqtt-protokol-untuk-iot&Itemid=303

[13] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, "Learning k for kNN classification," *ACM Transactions on Intelligent Systems and Technology*, vol. 43, pp. 1-19, January 2017.