**BAB III**

**ANALISIS DAN PERANCANGAN**

* 1. **Analisis**

Analisis adalah proses identifikasi, menganalisis dan menspesifikasikan kebutuhan-kebutuhan umum dari suatu aplikasi yang sifatnya spesifik, biasanya agar suatu saat dapat digunakan pada berbagai proyek lain.

Analisis akan menghasilkan spesifikasi-spesifikasi dari karakteristik-karakteristik operasional yang akan dimiliki oleh perangkat lunak yang akan dikembangkan, dimana hal ini mungkin mengidentifikasi antarmuka perangkat lunak dengan elemen-elemen sistem yang lain, dan juga menetapkan batasan-batasan yang harus dihadapi sistem atau perangkat lunak.

1. **Analisis Sistem**

Analisis sistem bertujuan untuk melakukan identifikasi masalah-masalah yang muncul dalam pembuatan sistem, hal ini dilakukan saat tahap analisa, perancangan sistem, mengimplementasikan sistem serta menguji dan menganalisis hasil implementasi dari sistem sehingga sistem dapat berjalan dengan baik dan selesai tepat pada waktu yang telah ditentukan.

Dalam tahapan analisa sistem ini akan dibahas mengenai proses atau alur kerja dari sistem keamanan rumah dan algoritma / metode yang akan diterapkan.

1. **Analisis Kebutuhan**

Analisis kebutuhan sistem adalah beberapa kebutuhan implementasi sistem berupa perangkat lunak dan perangkat keras.

1. **Analisis Kebutuhan Non Fungsional**

Analisis kebutuhan non-fungsional adalah bagian yang akan mendukung proses baik dari pembuatan sistem sampai implementasi aplikasi pengontrolan lampu tersebut.

1. **Kebutuhan Perangkat Keras**

Spesifikasi perangkat keras yang digunakan penulis untuk membangun aplikasi adalah sebagai berikut:

1. Processor Intel Core i3-2328
2. Memory 4 Gb
3. Harddisk 500 Gb

Perangkat keras yang penulis gunakan untuk mengimplementasikan aplikasi adalah sebagai berikut:

1. Perangkat Kontroler
2. Arduino ATMega 2560
3. Ethernet Shield
4. Rangkaian Sensor PIR (*Passive Infra Red*)
5. Rangkaian *Magnetic Switch*
6. Sensor *Ultrasonic* HC-SR04
7. *Membrane Keypad*
8. *Buzzer*
9. Perangkat Smartphone Android
10. CPU Min : 1GHz
11. Ram Min 512Mb
12. Free Space Min Internal 20Mb
13. **Kebutuhan Perangkat Lunak**

Kebutuhan perangkat lunak ini meliputi kebutuhan pembuatan aplikasi dan implementasi aplikasi tersebut.

1. Kebutuhan perangkat lunak pembuatan aplikasi terdiri dari:
2. Sistem operasi Window 7
3. Rational Rose
4. Java Develepment Kit 8
5. Android Studio
6. Android SDK version 18 sampai 21
7. Emulator Genymotion
8. Kebutuhan perangkat lunak implementasi aplikasi terdiri dari:
9. Sistem Operasi Android Min : Jelly Bean
10. **Analisis Kebutuhan Fungsional**

Analisis kebutuhan fungsional disini adalah langkah-langkah atau tahapan yang dilakukan oleh sistem.

1. Proses penerimaan perintah yang dikirim langsung oleh perangkat arduino yang terhubung melalui jaringan internet. Proses ini berada pada kondisi jika pengguna ingin menyalakan atau mematikan sistem melalui perangkat androidnya.
2. Proses pembacaan kondisi lingkungan. Proses ini dimulai ketika pengguna menyalakan sistem otomatis untuk pembacaan lingkungan dengan sensor yang ada.
3. Proses kontrol sensor menggunakan metode *fuzzy logic*. Proses ini adalah tahapan dari hasil pembacaan kondisi disekitar dengan beberapa sensor yang kemudian diproses menggunakan metode *fuzzy logic* yang digunakan untuk menentukan ada atau tidaknya pergerakan manusia di depan rumah dengan menggunakan sensor yang kemudian dihitung jaraknya oleh sensor ultrasonic.
4. **Metode Penyelesaian Masalah**

Dalam penelitian ini metode penyelesaian masalah yang diterapkan adalah *fuzzy logic*. Tahap penerapan *fuzzy logic* pada penelitian ini pemrosesan nilai input dari sensor PIR (*Passive Infra Red*) yang menghasilkan nilai digital dari deteksi pergerakan manusia dan *counter* (sudah berapa kali sensor mendeteksi). Fungsi dari perangkat sensor tersebut sebagai parameter terhadap keadaan halaman rumah. Berikut tahapan proses *fuzzy logic* yaitu *fuzzification*, *inference* dan *defuzzification*.

1. ***Fuzzification***

Langkah dari proses *fuzzification* adalah mengkonversi masukan-masukan nilai kebenaran dari perangkat sensor menjadi bentuk nilai linguistik atau *fuzzy input*.

1. Rancangan fungsi keanggotaan sensor PIR yang merupakan data yang berbentuk nilai crips atau nilai awal (*analog value*) dari sensor yang diubah menjadi *fuzzy input*.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  |  |
| 1 |  |  |  | |  |  |
|  |  |  |  | |  |  |
| 0 |  |  |  | | 1 |  |
|  | | 0 | |

**Gambar 3.0 Fungsi Sensor PIR**

**Tabel 3.0 Keterangan Fungsi PIR**

|  |  |
| --- | --- |
| Nilai Crips | Nilai Linguistik |
| 0 | Undetected (U) |
| 1 | Detected (D) |

1. Rancangan fungsi keanggotaan counter yang merupakan data yang berbentuk nilai crips atau nilai awal (*analog value*) sebagai acuan terhadap sensor PIR yang diubah menjadi *fuzzy input*.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  |  | |  |  |
| 1 |  |  |  | |  |  | |  |  |
|  |  |  |  | |  |  | |  |  |
| 0 |  |  |  | |  |  | |  |  |
|  | | < 3 | | >=3 - <=5 | | | >5 | | | |

**Gambar 3.1 Fungsi Keanggotaan Counter**

**Tabel 3.1 Keterangan Fungsi Keanggotaan Counter**

|  |  |
| --- | --- |
| Nilai Crips | Nilai Linguistik |
| < 3 kali | Caounter 1 |
| >=3 - <=5 kali | Caounter 2 |
| >5 kali | Caounter 3 |

1. ***Inference***

Berdasarkan proses *fuzzification* sensor yang ada, akan dibuat aturan dari nilai-nilai yang terbaca dari sensor tersebut. Dimana aturan disini merupakan perintah untuk mengirim notifikasi. Adapun aturan-aturannya sebagai berikut:

**Tabel 3.2 *Rule Inference* Aturan Fuzzy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Counter | Counter 1 | Counter 2 | Counter 3 |
| PIR |  |
| Undetected | | N | N | N |
| Detected | | N | W | S |

Keterangan :

N = Normal

W = Waspada

S = Awas

Dari uraian aturan *fuzzy / rule inference* yang berasal dari 3 kolom sensor *ultrasonic* dan 6 kolom sensor maka menghasilkan 18 aturan. Hasil dari aturan tersebut adalah sebagai berikut:

1. IF PIR = U AND Counter = < 3 THEN N
2. IF PIR = U AND Counter = >=3 - <=5 THEN N
3. IF PIR = U AND Counter = > 5 THEN N
4. IF PIR = D AND Counter = < 3 THEN N
5. IF PIR = D AND Counter = >=3 - <=5 THEN W
6. IF PIR = D AND Counter = > 5 THEN S
7. **Defuzzification**

Proses *defuzzification* pada pengontrolan ini menggunakan *Center of Gravity* atau dikenal sebagai Mamdani, dimana metode ini mengambil nilai yang memiliki nilai keanggotaan di antara nilai keanggota yang ada. Proses *defuzzification* disini akan mengubah aturan *fuzzy* atau *rule inference* menjadi nilai crips *output*.

1. S = Awas

Send.Notification ”Awas : Pergerakan Mendekati Depan Rumah”

Buzzer = PIN 30 (HIGH)

1. W = Waspada

Send.Notification ”Waspada : Terdeteksi Pergerakan di Sekitar Rumah”

Buzzer = PIN 30 (LOW)

1. N = Normal

Buzzer = PIN 30 (LOW)

* 1. **Perancangan Sistem**

Analisis perancangan sistem menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik. Analisis yang dilakukan dimodelkan dengan menggunakan *UML (Unified Modeling Language)*. Tahap-tahap pemodelan dalam analisis tersebut antara lain identifikasi aktor*, usecase diagram,* skenario*, activity diagram, sequence diagram, class diagram*.

1. **Flow Chart**

*Flowchart* atau “diagram alir adalah suatu standar untuk menggambarkan suatu proses” (Kadir A, 2012). Proses yang akan di gambarkan di sini merupakan proses langkah dari cara kerja dari metode algoritma yang digunakan di dalam pembuatan aplikasi *game* edukasi berbentuk *puzzle*. Berikut adalah algoritma yang di gunakannya.

**3.3.1.1 Flowchart Sistem**

Berikut adalah flowchart sistem yang digunakan pada aplikasi sistem keamanan rumah :



**Gambar 3.2 Flowchart Sistem**

Keterangan Flowchart Sistem :

1. Sistem keamanan rumah berbasis aplikasi android dimulai dengan proses daftar.
2. Pengguna harus memastikan dirinya sudah terfdaftar sebelum login dan masuk sistem keamanan rumah.
3. Sistem terdiri dari menu kontrol sistem, kelola pengguna, kelola profil, kelola password, monitoring, dan Log.

**3.3.1.2 Flowchart Kelola Sistem**

Berikut adalah flowchart menu kotrol sistem pada aplikasi sistem keamanan rumah :



**Gambar 3.3 Flowchart Menu Kelola Sistem**

Keterangan Flowchart Menu Kelola Sistem :

1. Menu kelola sistem berfungsi untuk menyalakan atau mematikan sistem.
2. Ketika pengguna memilih menyalakan / mematikan sistem sistem akan menerima perintah ke web service dan diteruskan ke arduino.
3. Ketika sistem dimatikan / dinyalakan arduino akan mengirim *feedback* status sistem yang akan ditampilkan pada aplikasi.

**3.3.1.3 Flowchart Kelola Pengguna**

Berikut adalah flowchart menu kelola pengguna pada aplikasi sistem keamanan rumah :



**Gambar 3.4 Flowchart Menu Kelola Pengguna**

Keterangan Flowchart Menu Kelola Pengguna :

1. Pada menu kelola pengguna, root dapat menambahkan pengguna baru yaitu koordinator keluarga dan anggota keluarga, sementara koordinator keluarga hanya dapat menambahkan anggota keluarga.
2. Terdapat pula fungsi edit dan delete sibling (anggota keluarga) yang hanya dapat dilakukan oleh root dan koordinator keluarga.

**3.3.1.4 Flowchart Kelola Profil**

Berikut adalah flowchart menu kelola profil pada aplikasi sistem keamanan rumah :



**Gambar 3.5 Flowchart Menu Kelola Profil**

Keterangan Flowchart Menu Kelola Profil :

1. Menu kelola profil berfungsi untuk mengedit data diri dari setiap pengguna.
2. Dimulai dengan memilih menu kelola profil kemudian pengguna memasukan data baru apabila data sesuai maka akan disimpan, jika tidak maka akan tampil peringatan

**3.3.1.5 Flowchart Kelola Password**

Berikut adalah flowchart menu kelola password pada aplikasi sistem keamanan rumah :



**Gambar 3.6 Flowchart Menu Kelola Password**

Keterangan Flowchart Menu Kelola Password :

1. Menu kelola profil berfungsi untuk mengedit password dari setiap pengguna.
2. Dimulai dengan memilih menu kelola password kemudian pengguna memasukan pssword lama dan baru, apabila password lama sesuai maka akan disimpan, jika tidak maka akan tampil peringatan.

**3.3.1.6 Flowchart Log**

Berikut adalah flowchart menu kelola password pada aplikasi sistem keamanan rumah :



**Gambar 3.7 Flowchart Menu Log**

Keterangan Flowchart Menu Log :

1. Menu log berfungsi untuk menampilkan data riwayat sensor hasil pembacaan sensor pada arduino.

**3.3.1.7 Flowchart Proses Fuzzy Logic**

Berikut adalah flowchart menu proses fuzzy logic yang digunakan pada aplikasi sistem keamanan rumah :



**Gambar 3.8 Flowchart Proses Fuzzy Logic Pada Sistem Keamanan Rumah**

1. Saat perangkat berjalan maka sensor akan akan bekerja sesuai fungsinya. Sensor PIR mendeteksi pergerakan manusia, sensor *ultrasonic* berfungsi untuk mengukur jarak objek.
2. Nilai hasil pengukuran sensor berupa nilai awa (nilai crips), kemudian nilai tersebut akan dirubah menjadi nilai linguistik melalui proses f*uzzyfication*.
3. Setelah diperoleh nilai linguistik (*fuzzyfication*), langkah selanjutnya adalah proses untuk menentukan aturan *fuzzy inferance.*
4. Proses terakhir yaitu mengubah nilai linguistik menjadi nilai pasti (*Defuzzyfication*) untuk mendapatkan nilai *output* untuk ditampilkan pada aplikasi android di *Smartphone*.
5. ***Use Case* Diagram**

Pemodelan use case adalah pemodelan sistem dari perspektif pandangan pemakai akhir (end user). Model use case adalah pandangan dari luar sistem, sementara model rancangan adalah pandangan dari dalam. Model use case menangkap penggunaan-penggunaan sistem, sedangkan model rancangan merepresentasikan pembangunan dari sistem. Berikut ini adalah gambar dari use case untuk sistem keamanan rumah.

1. *Use Case* Sistem



**Gambar 3.9 Use Case Diagram Pengontrolan**

1. **Skenario**

**Tabel 3.3 Skenario Kontrol Sistem**

|  |  |
| --- | --- |
| Nama Use Case : | Kontrol Sistem |
| Aktor : | Root, Anggota, dan Koordinator Keluarga, |
| Tujuan : | Agar pengguna dapat melakukan kelola password, profil, dan kelola sistem, |
|  |  |
| Pre Condition | Pengguna akan mematikan / menyalakan sistem keamanan rumah. |
| Aktor | Sistem |
| 1. Aktor memilih menu kontrol sistem. |  |
|  | 1. Sistem menampilkan submenu kelola password, kelola profil, dan kelola sistem. |
| 1. Aktor memilih submenu kelola password. |  |
|  | 1. Sistem menampilkan form ubah password. |
| 1. Aktor memasukan password lama dan baru. |  |
|  | 1. Sistem melakukan validasi password. |
|  | 1. Sistem menyimpan password baru. |
| 1. Aktor memilih sub menu kelola profil. |  |
|  | 1. Sistem menampilkan form ubah profil. |
| 1. Aktor mengisi form profil. |  |
|  | 1. Sistem menyimpan data profil. |
| 1. Aktor memilih sub menu kelola sistem. |  |
|  | 1. Sistem menampilkan form menyalakan / mematikan sistem. |
| 1. a. Aktor koordinator dan anggota memilih perintah menyalakan / mematikan sistem rumahnya sendiri.   b. Aktor root memilih perintah menyalakan / mematikan sistem salah satu rumah. |  |
|  | 1. Sistem mengirim perintah jika perangkat android terhubung ke internet, jika tidak maka ada peringatan (*alert*). |
|  | 1. Sistem pada perangkat arduino menerima perintah dan melakukan instruksi kedapa PIN digital sesuai perintah engguna. |
|  | 1. Sistem menampilkan status sistem nyala atau mati. |
| Post Condition | Password aktor berubah, profil aktor berubah, dan jika pengiriman perintah terkirim ke perangkat arduino, maka sistem akan nyala atau mati sesuai dengan perintah. |

**Tabel 3.4 Skenario Monitoring**

|  |  |
| --- | --- |
| Nama Use Case : | Monitoring |
| Aktor : | Root, Anggota, dan Koordinator Keluarga |
| Tujuan : | Agar aktor dapat melihat status terkini sensor melalui perangkat android. |
|  |  |
| Pre Condition | Sensor dalam keadaan nyala dan mengirim data. |
| Aktor | Sistem |
| 1. Aktor memilih menu monitoring. |  |
|  | 1. a. Untuk aktor koodinator dan anggota keluarga, sistem menampilkan status terkini dari setiap sensor yang ada pada rumah pengguna.   b. Untuk aktor root, sistem menampilkan status terkini dari setiap sensor yang ada pada setiap rumah keluarga yang ada. |
| Post Condition | Aplikasi akan menampilkan status sensor. |

**Tabel 3.5 Skenario Log**

|  |  |
| --- | --- |
| Nama Use Case : | Log |
| Aktor : | Root, Anggota, dan Koordinator Keluarga |
| Tujuan : | Agar pengguna dapat melihat riwayat sensor yang telah mendeteksi tanda bahaya pada rumah. |
|  |  |
| Pre Condition | Sensor dalam keadaan nyala dan mengirim data. |
| Aktor | Sistem |
| 1. Aktor memilih menu laporan. |  |
|  | 1. Sistem menampilkan riwayat dari setiap sensor yang mendeteksi tanda bahaya pada rumah pengguna. |
| Post Condition | Aplikasi akan menampilkan daftar riwayar sensor yang mendeteksi tanda bahaya pada rumah. |

**Tabel 3.6 Skenario Login**

|  |  |
| --- | --- |
| Nama Use Case : | Login |
| Aktor : | Pengguna |
| Tujuan : | Sebagai otorisasi bagi pengguna agar dapat masuk ke sistem. |
|  |  |
| Pre Condition | Pengguna akan masuk ke sistem. |
| Aktor | Sistem |
| 1. Aktor membuka aplikasi. |  |
|  | 1. Sistem menampilkan form Login. |
| 1. Pengguna memasukan username dan password. |  |
|  | 1. Sistem melakukan validasi username dan password |
|  | 1. Jika tidak sesuai maka sistem menampilkan alert, jika sesuai sistem menampilkan halaman utama |
| Post Condition | Aktor berhasil login dan masuk ke dalam sistem. |

**Tabel 3.7 Skenario Register**

|  |  |
| --- | --- |
| Nama Use Case : | Login |
| Aktor : | Koordinator keluarga dan root. |
| Tujuan : | Sebagai otorisasi bagi pengguna agar dapat masuk ke sistem. |
|  |  |
| Pre Condition | Pengguna akan masuk ke sistem. |
| Aktor | Sistem |
| 1. Aktor memilih menu registrasi. |  |
|  | 1. Sistem menampilkan form registrasi. |
| 1. Aktor mengisi form registrasi. |  |
|  | 1. Sistem melakukan validasi . |
|  | 1. Jika user sudah terdaftar maka sistem menampilkan *alert*. |
|  | 1. Akun baru berhasil disimpan di database |
| Post Condition | Aktor berhasil terdaftar dan dapat masuk ke dalam sistem dengan login. |

**Tabel 3.8 Skenario Kelola Pengguna**

|  |  |
| --- | --- |
| Nama Use Case : | Kelola Pengguna |
| Aktor : | Koordinator keluarga dan root. |
| Tujuan : | Agar koordinator keluarga dapat mengola anggota keluarganya dan root dapat mengelola seluruh pengguna. |
|  |  |
| Pre Condition | a. Ketika koordinator keluarga ingin mengelola anggota keluarganya.  b. Ketika root ingin mengelola seluruh pengguna. |
| Aktor | Sistem |
| 1. Koordinator keluarga memilih menu kelola pengguna. |  |
|  | 1. Sistem menampilkan form pengelolaan anggota keluarganya. |
| 1. Koordinator keluarga melakukan pengelolaan pengguna.. |  |
| 1. Root memilih menu kelola pengguna. |  |
|  | 1. Sistem menampilkan form pengelolaan seluruh pengguna. |
| 1. Root melakukan pengelolaan pengguna.. |  |
| Post Condition | Koordinator keluarga telah mengelola anggota keluarganya, dan root telah mengelola seluruh pengguna. |

1. **Activity Diagram**

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Berikut ini akan digambarkan aktivitas-aktivitas diagram dari masing-masing aktor :

1. Activity Diagram Register



**Gambar 3.10 Activity Diagram Register**

1. Activity Diagram Monitoring



**Gambar 3.11 Activity Diagram Monitoring**

1. Activity Diagram Kelola Sistem



**Gambar 3.12 Activity Diagram Kelola Sistem**

1. Activity Diagram Login



**Gambar 3.13 Activity Diagram Login**

1. Activity Diagram Kelola Profil



**Gambar 3.14 Activity Diagram Kelola Profil**

1. Activity Diagram Log



**Gambar 3.15 Activity Diagram Log**

1. Activity Diagram Kelola Password



**Gambar 3.16 Activity Diagram Kelola Password**

1. Activity Diagram Kelola Pengguna



**Gambar 3.17 Activity Diagram Kelola Pengguna**

1. **Class Diagram**

Diagram kelas (*class* diagram) sebuah spesifikasi yang jika diinstasiasikan akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Kelas memiliki apa yang disebut atribut dan metode atau operasi.



**Gambar 3.18 Class Diagram**

1. **Sequence Diagram**

*Sequence diagram* mendokumentasikan komunikasi/interaksi antar kelas-kelas. Diagram ini menunjukkan sejumlah obyek dan *message* (pesan) yang diletakkan diantara obyek-obyek didalam *use case.* *Sequence Diagram* menunjukan bagaimana detil operasi dilakukan – pesan apa yang dikirim dan kapan.

1. Log



**Gambar 3.19 Sequence Diagram Log**

1. Sequence Kelola Profil



**Gambar 3.20 Sequence Diagram Kelola Profil**

1. Sequence Login



**Gambar 3.21 Sequence Diagram Login**

1. Sequence Monitorng



**Gambar 3.22 Sequence Diagram Monitoring**

1. Sequence Diagram Register



**Gambar 3.23 Sequence Diagram Register**

1. Sequence Kelola Pengguna



**Gambar 3.24 Sequence Diagram Kelola Pengguna**

1. Sequence Kelola Password



**Gambar 3.25 Sequence Diagram Kelola Password**

1. Sequence Kelola Sistem



**Gambar 3.26 Sequence Diagram Kelola Sistem**

* 1. **Perancangan Hardware**

Perancangan *hardware* merupakan suatu tahap yang sangat penting dalam pembuatan suatu alat, sebab dengan menganalisa komponen yang digunakan maka alat yang akan dibuat dapat bekerja seperti yang diharapkan. Untuk mendapatkan hasil yang optimal, terlebih dahulu membuat rancangan yang baik yaitu dengan memperhatikan sifat dan karakteristik dari tiap-tiap komponen yang digunakan sehingga dapat menghindari kerusakan pada komponen yang digunakan dan mempermudah dalam pengerjaannya.

1. **Block Diagram**

Blok Diagram Sistem bertujuan untuk memberikan gambaran secara umum mengenai proses berjalannya sebuah sistem. Sehingga akan mempermudah dalam mengimplementasikan sistem tersebut.

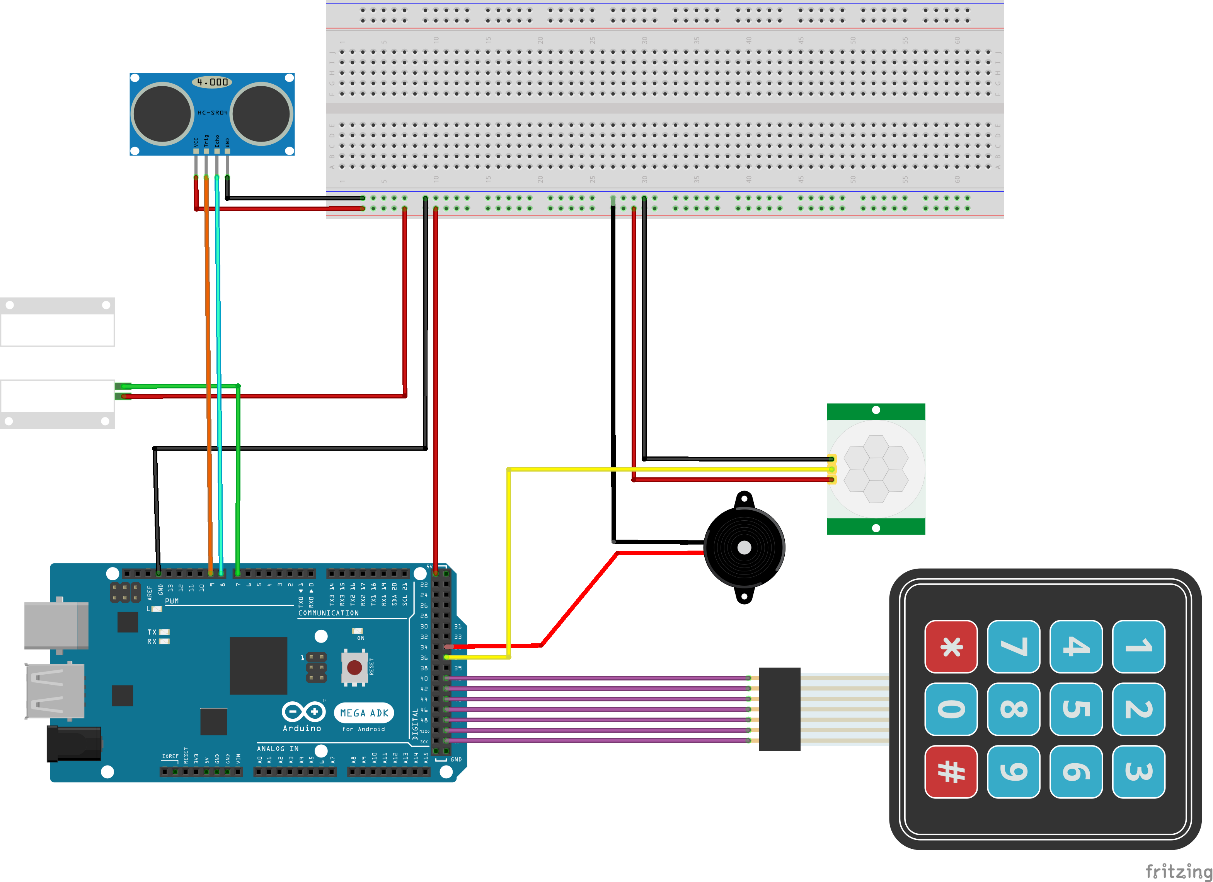


**Gambar 3.27 Blok Diagram Sistem Keamanan Rumah**

Berikut dalah fungsi dari masing-masing komponen penyusun sistem keamanan rumah :

1. Mikrokontroler atmega2560 berfungsi sebagai pusat pengendali. Semua data diolah dan cek koneksi sebelum data dikirim ke server.
2. Sensor PIR (*Passive Infra Red*) merupakan sensor yang berfungsi untuk mendeteksi manusia, keluaran nilai sensor ini berupa tegangan digital yang kemudian akan dikirim ke mikrokontroler sebagai pusat pengendali.
3. Sensor HC-SR04 merupakan sensor yang berfungsi untuk mendeteksi jarak objek yang sebelumnya sudah dideteksi oleh sensor PIR, keluaran nilai sensor berupa tegangan digital yang kemudian akan dikirim ke mikrokontroler sebagai pusat pengendali.
4. Sensor *Magnetic Switch* merupakan sensor yang berfungsi untuk mendeteksi ketika pintu terbuka atau tertutup , keluaran nilai sesnor berupa tegangan digital yang kemudian akan dikirim ke mikrokontroler sebagai pusat pengendali.
5. *Membrane Keypad* merupakan sebuah keypad yang berfungsi untuk menyalakan / mematikan sistem dengan memasukan PIN keamanan yang terintegrasi dengan aplikasi android.
6. *Buzzer* berfungsi sebagai alarm.
7. Server berfungsi untuk menyimpan serta melayani *request* dari client, aplikasi android.
8. User/Client adalah orang yang bertugas mengakses aplikasi.
   * 1. **Diagram pengkabelan Sistem Keamanan Rumah**

Perancangan wiring diagram merupakan tahap lanjut dari blok diagram yang dirancang. yang mana tahap ini merupakan perancangan system bagian hardware.untuk lebih jelasnya dapat dilihat pada gambar berikut :

****

**Gambar 3.28 Pengkabelan Arduino dan Sensor *Ultrasonic***

Keterangan Port :

a. Sensor Ultrasonic

1. VCC daya ultrasonik dengan label warna merah dihubungkan langsung ke input 5v di pin arduino.
2. Trig dengan kabel warna abu dihubungkan langsung ke pin digital 8
3. Echo dengan kabel warna biru dihubungkan langsung ke pin digital 9
4. GND dengan kabel warna hitam dihbungkan langsung ke pin Gnd pada arduino

b. Buzzer

* 1. I/O dengan kabel warna merah dihubungkan ke pin 35 pada arduino.
  2. GND dengan kabel warna hitam dihubungkan ke *breadboard*.

c. Keypad

1. 3 pin kolom terhubung ke arduino dari pin 41, 43, 45
2. 4 pin baris terhubung ke arduino dari pin 47, 49, 51, 53.

d. Sensor PIR

1. Trig pin dengan kabel warna kuning dihubungkan langsung ke pin digital 37.
2. Trig pin dari sensor PIR ke 2 dihubungkan langsung ke pin digital 33.
3. VCC daya PIR dengan label warna merah dihubungkan ke *breadboard* input 5v dari pin arduino.
4. GND sensor PIR dengan label Hitamdihubungkan ke *breadboard* GND dari pin arduino.

e. *Magnetic Switch*

1. Trig pin *magnetic switch* dengan kabel warna hijau dihubungkan langsung ke pin digital 7.
2. VCC daya *magnetic switch* dengan label warna merah dihubungkan ke *breadboard* input 5v dari pin arduino.
   1. **Perancangan Database**

Tabel yang penulis gunakan pada penelitian ini yaitu tabel kamus istilah komputer. Berikut ini uraian secara rinci dari tabel yang digunakan pada penelitian ini.

1. Tabel t\_user

Tabel ini merupakan tabel yang berisi data-data pengguna.

**Tabel 3.9 t\_user**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kolom** | **Jenis** | **Kosong** | **Primary Key** | **Auto Increment** |
| ***id*** | int(4) | Tidak | ya | ya |
| username | varchar(99) | Tidak | - | - |
| password | varchar(32) | Tidak | - | - |
| nama | varchar(199) | Tidak | - | - |
| alamat | text | Tidak | - | - |
| tipe | int(4) | Tidak | - | - |
| register\_datetime | datetime | Tidak | - | - |
| status | int(4) | Tidak | - | - |
| last\_login | datetime | Tidak | - | - |
| photo | varchar(199) | Tidak | - | - |
| parent | int(4) | Tidak | - | - |
| API\_KEY | varchar(32) | Tidak | - | - |
| secure\_key | varchar(4) | Tidak | - | - |

2. Tabel t\_ref\_user

Tabel ini merupakan tabel yang berisi data-data tipe user.

**Tabel 3.10 t\_ref\_user**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kolom** | **Jenis** | **Kosong** | **Primary Key** | **Auto Increment** |
| ***id*** | int(4) | Tidak | ya | ya |
| tipe | varchar(99) | Tidak | - | - |

3. Tabel t\_sensor

Tabel ini merupakan tabel yang berisi data-data tipe user.

**Tabel 3.11 t\_sensor**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kolom** | **Jenis** | **Kosong** | **Primary Key** | **Auto Increment** |
| ***id*** | int(9) | Tidak | ya | ya |
| state | varchar(19) | Tidak | - | - |
| outdoor | varchar(1) | Tidak | - | - |
| indoor | varchar(1) | Tidak | - | - |
| ussrf | varchar(5) | Tidak | - | - |
| magnetic | int(1) | Tidak | - | - |
| datetime | datetime | Tidak | - | - |
| API\_KEY | varchar(32) | Tidak | - | - |

* 1. **Perancangan Antarmuka**

Antarmuka pemakai merupakan media komunikasi antara pemakai dengan sistem komputer. Hasil perancangan antarmuka pemakai akan memudahkan proses implementasi.

Merupakan suatu bentuk tampilan dari program yang akan dibuat untuk kebutuhan *interface* dengan *user.* Berikut merupakan perancangan tampilan yang dibuat :

**3.6.1 Perancangan Antarmuka Aplikasi Android**

1. Tampilan Login



**Gambar 3.29 Tampilan Login**

**Tabel 3.12 Keterangan Tampilan Login**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewLogin | Caption | Login |
| ImageViewLogo | Caption | Logo |
| EditText1 | Caption | Username |
| EditText2 | Caption | Password |
| Button1 | Caption | Login |
| Button2 | Caption | Daftar |

1. Tampilan Registrasi/daftar



**Gambar 3.30 Tampilan Registrasi**

**Tabel 3.13 Keterangan Tampilan Registrasi**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewLogin | Caption | Registrasi |
| ImageViewLogo | Caption | Logo |
| EditText1 | Caption | Username |
| EditText2 | Caption | Password |
| EditText3 | Caption | Nama |
| EditText4 | Caption | Alamat |
| Button1 | Caption | Daftar |
| Button2 | Caption | Login |

1. Tampilan Dasboard



**Gambar 3.31 Tampilan Dashboard**

**Tabel 3.14 Keterangan Tampilan Dashboard**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewDashboard | Caption | Dasboard |
| TextView 1 | Caption | Total User |
| TextView 2 | Caption | User Block |
| TextView 3 | Caption | Data Hari Ini |
| TextView 4 | Caption | Total Data |

1. Tampilan Mode Kontrol Sistem



**Gambar 3.32 Tampilan Mode Kontrol**

**Tabel 3.15 Keterangan Tampilan Mode Kontrol**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| ImageButtonManualMode | Caption | Image Manual |
| ToggleButton Sensor | Caption | Image Sensor |

1. Tampilan Monitoring

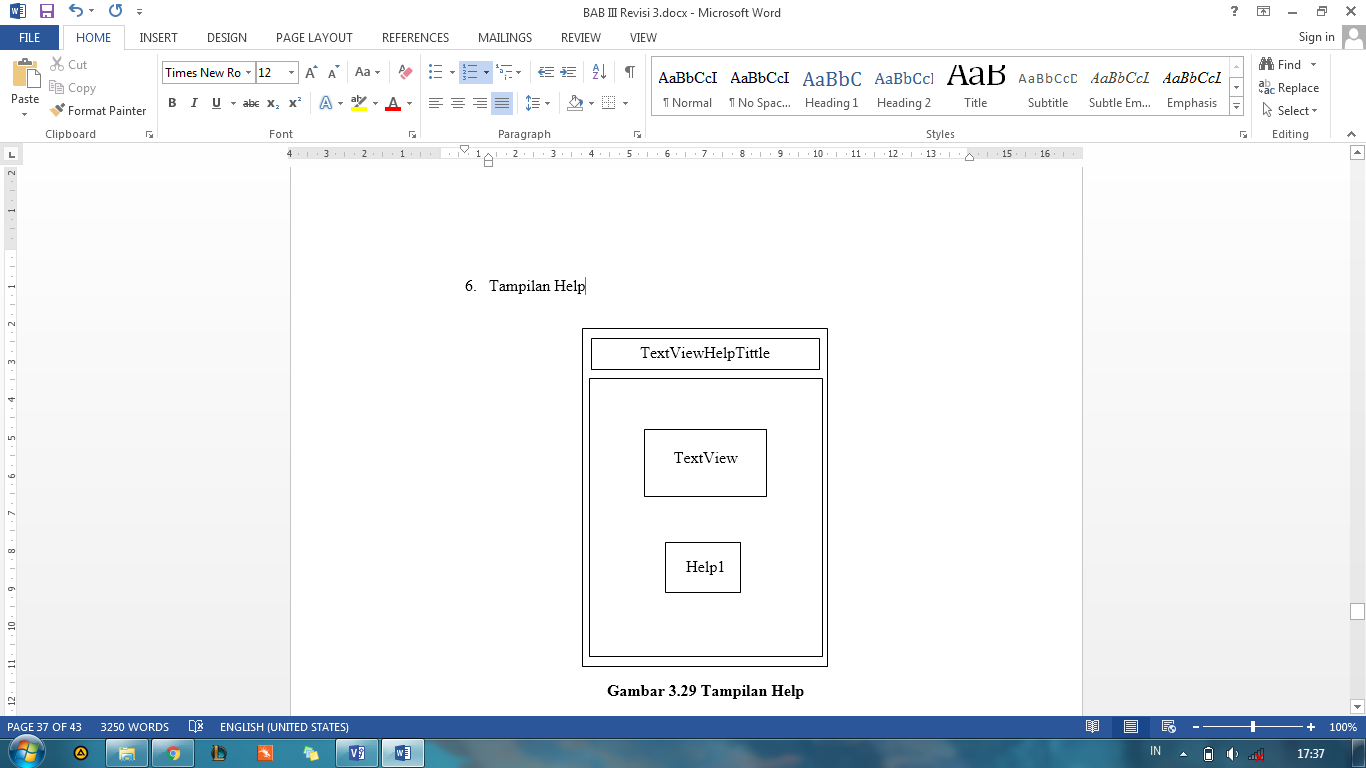


**Gambar 3.33 Tampilan Monitoring**

**Tabel 3.16 Keterangan Tampilan Monitoring**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewStatus | Caption | Status Sensor |
| TextView Sensor1 | Caption | Status Halaman |
| TextView Sensor2 | Caption | Status Ruang Tengah |
| TextView Sensor3 | Caption | Status *Magnetic Switch* |

1. Tampilan Help



**Gambar 3.34 Tampilan Help**

**Tabel 3.17 Keterangan Tampilan help**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewHelpTittle | Caption | Petunjuk |
| TextView Help1 | Caption | Petunjuk1 |

1. Tampilan Kelola Password



**Gambar 3.35 Tampilan Kelola Password**

**Tabel 3.18 Keterangan Tampilan Kelola Pasword**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewKelolaProfil | Caption | Kelola Profil |
| Edit Text 1 | Caption | Password Lama |
| Edit Text 2 | Caption | Password Baru |
| Edit Text 3 | Caption | Konfirmasi Password |
| Button 1 | Caption | Update |
| Button 2 | Caption | Kembali |

1. Tampilan Kelola Profil

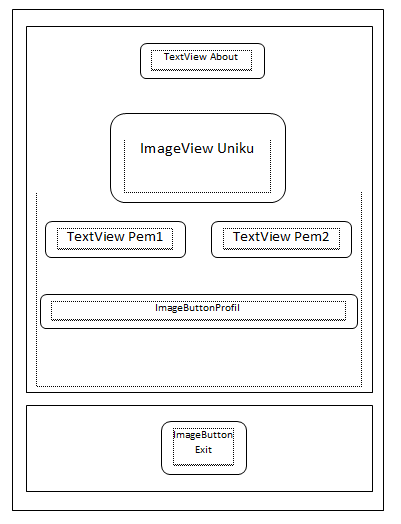


**Gambar 3.36 Tampilan Profil**

**Tabel 3.19 Keterangan Tampilan Profil**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewKelolaProfil | Caption | Kelola Profil |
| Edit Text 1 | Caption | Username |
| Edit Text 2 | Caption | Nama |
| Edit Text 3 | Caption | Alamat |
| Edit Text 4 | Caption | Tipe Account |
| Edit Text 5 | Caption | Tanggal Registrasi |
| Edit Text 6 | Caption | API Key |
| Edit Text 7 | Caption | Secure Key |
| Button 1 | Caption | Update |
| Button 2 | Caption | Kembali |

1. Tampilan About



**Gambar 3.37 Tampilan About**

**Tabel 3.20 Keterangan Tampilan About**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewAbout | Caption | Terimakasih kepada |
| ImageView Uniku | Caption | Image Logo Uniku |
| TextView Pem1 | Caption | Pembimbing 1 |
| TextView Nama1 | Caption | Sugeng Supriyadi, M.Kom |
| TextView Pem2 | Caption | Pembimbing 2 |
| TextView Nama2 | Caption | Nunu Nugraha, M.T |

**3.6.2 Perancangan Antarmuka Web Service**

1. Tampilan Login



**Gambar 3.38 Tampilan Login**

**Tabel 3.21 Keterangan Tampilan Login**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| SideViewBg | Caption | Background |
| EditText1 | Caption | Username |
| EditText2 | Caption | Password |
| Button1 | Caption | Login |
| Button2 | Caption | Daftar |

1. Tampilan Registrasi



**Gambar 3.39 Tampilan Registrasi**

**Tabel 3.22 Keterangan Tampilan Registrasi**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| SideViewBg | Caption | Background |
| EditText1 | Caption | Username |
| EditText2 | Caption | Password |
| EditText3 | Caption | Nama |
| EditText4 | Caption | Alamat |
| Button1 | Caption | Daftar |
| Button2 | Caption | Login |

1. Tampilan Dashboard



**Gambar 3.40 Tampilan Dashboard**

**Tabel 3.23 Keterangan Tampilan Dashboard**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextTotal | Caption | Total User |
| TextBlock | Caption | User Block |
| TextRecent | Caption | Data Hari Ini |
| TextTotalData | Caption | Total Data |

1. Tampilan Kelola Password



**Gambar 3.41 Tampilan Kelola Password**

**Tabel 3.24 Keterangan Tampilan Kelola Pasword**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextViewKelolaProfil | Caption | Kelola Profil |
| Edit Text 1 | Caption | Username |
| Edit Text 2 | Caption | Nama |
| Edit Text 3 | Caption | Alamat |
| Button 1 | Caption | Update |
| Button 2 | Caption | Kembali |

1. Tampilan Kelola Password



**Gambar 3.42 Tampilan Kelola Password**

**Tabel 3.25 Keterangan Tampilan Kelola Pasword**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| Edit Text 1 | Caption | Password Lama |
| Edit Text 2 | Caption | Password Baru |
| Edit Text 3 | Caption | Konfirmasi Password |
| Button 1 | Caption | Update |
| Button 2 | Caption | Kembali |

1. Tampilan Monitoring



**Gambar 3.43 Tampilan Monitoring**

**Tabel 3.26 Keterangan Tampilan Monitoring**

|  |  |  |
| --- | --- | --- |
| Komponen | Properti | Nilai |
| TextSensor1 | Caption | Status Halaman |
| TextSensor2 | Caption | Status Ruang Tengah |
| TextSensor3 | Caption | Status *Magnetic Switch* |