



**SISTEM OTOMATISASI LAMPU RUMAH DENGAN  
ALGORITME NAÏVE BAYES BERDASARKAN KEBIASAAN  
PENGHUNI RUMAH**

**SKRIPSI**

**KEMINATAN TEKNIK KOMPUTER**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Ahmad Yazid Bastomi

NIM: 135150307111022



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

**MALANG**

**2019**



## PERNYATAAN PENGESAHAN

SISTEM OTOMATISASI LAMPU RUMAH DENGAN ALGORITME NAIVE BAYES  
BERDASARKAN KEBIASAAN PENGHUNI RUMAH

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Ahmad Yazid Bastomi

NIM: 135150307111022

Skripsi ini telah diuji dan dinyatakan lulus pada  
Malang, 25 Juli 2019

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing 1

Dosen Pembimbing 2

Dahnial Syauqy, S.T., M.T., M.Sc.  
NIK: 2016078704231002

Issa Arwani, S.Kom, M.Sc.  
NIK: 19830922 201212 1 003

Mengetahui

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D  
NIP: 19710518 200312 1 001



## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 9 Juli 2019



Ahmad Yazid Bastomi

NIM: 135150307111022



## PRAKATA

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat serta hidayahnya sehingga penulis dapat mengerjakan dan menyelesaikan skripsi yang berjudul “Sistem Otomatisasi Lampu Rumah dengan Algoritme *Naive Bayes* Berdasarkan Kebiasaan Penghuni Rumah” dengan lancar serta halangan yang berarti. Shalawat serta salam tidak lupa diucapkan penulis pada junjungan kita Nabi besar Muhammad SAW yang telah membimbing kita kejalan yang benar.

Terdapat banyak halangan dalam penulisan skripsi ini, tetapi banyak pihak yang membantu penulis dengan dukungan dan bimbingan sehingga skripsi ini dapat terselesaikan. Oleh karena itu penulis menyampaikan terimakasih dan rasa hormat sebesar – besarnya kepada:

1. Bapak As’adur Rofiq S.E, Khusnul Khotimah dan Sofiatius Sa’adah selaku kedua orangtua dan saudara penulis yang telah memberikan do’a dan dorongannya sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D. selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya Malang.
3. Bapak Heru Nurwarsito, Ir., M. Kom. Selaku Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Brawijaya Malang.
4. Bapak Tri Astoto Kurniawan, S.T., M.T. Ph.D selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang yang telah memberikan banyak nasihat.
5. Bapak Dahnia Syauqy, S.T., M.T., M.Sc. selaku Ketua Program Studi Teknik Komputer Universitas Brawijaya Malang dan pembimbing I yang telah membimbing dan mengarahkan sehingga penulis dapat menyelesaikan skripsi ini.
6. Bapak Issa Arwani, S.Kom, M.Sc. selaku pembimbing II yang telah membimbing dan mengarahkan sehingga penulis dapat menyelesaikan skripsi ini.
7. Teman – teman Teknik Komputer angkatan 2013 khususnya Idang, Iswan, Dimas, Aldi, dan Latief yang telah memberikan saran dan dorongannya serta seluruh pihak yang tidak dapat disebutkan satu persatu oleh penulis yang telah berperan dalam penyelesaian skripsi.

Malang, 9 Juli 2019

Penulis

a.yazidbas@gmail.com



## ABSTRAK

**Ahmad Yazid Bastomi, Sistem Otomatisasi Lampu Rumah dengan Algoritme Naive Bayes Berdasarkan Kebiasaan Penghuni Rumah**

**Pembimbing: Dahnil Syauqy, S.T., M.T., M.Sc. dan Issa Arwani, S.Kom, M.Sc.**

Sebagai alat penerangan yang memudahkan aktifitas manusia dalam keadaan gelap, lampu mempunyai fungsi yang sangat penting dalam hidup keseharian. Di dalam rumah lampu mempunyai peran sebagai alat penerangan yang mendukung aktifitas di dalam rumah dan fungsinya hampir tidak tergantikan. Akan tetapi penggunaan yang harus secara manual menyebabkan penghuninya menemui masalah seperti lupa mematikan lampu pada siang hari sehingga menyebabkan pemborosan listrik dan berdampak pada pengeluaran biaya yang lebih besar. Penelitian ini ditunjukkan untuk membuat sistem otomatisasi pada lampu dengan menggunakan acuan kebiasaan penghuni rumah dalam menyalakan dan mematikan lampu. Sistem menggunakan kebiasaan penghuni rumah dan mengolahnya dengan algoritme Naive Bayes untuk menentukan nyala lampu secara otomatis. Salah satu kelebihan Algoritme ini adalah penggunaannya yang mudah, akurat dan tidak memerlukan terlalu banyak data. Sistem menggunakan mikrokontroler Arduino UNO sebagai pengontrol utama keseluruhan sistem. Dalam penggunaannya terdapat tiga saklar yang berfungsi sebagai input utama sistem. Saklar tersebut mengatur mode sistem dan menangkap data waktu secara *Real Time* sehingga penggunaannya lebih mudah. Data yang didapat juga disimpan di EEPROM Arduino UNO sehingga dapat digunakan secara berkelanjutan. Dari keseluruhan hasil sistem yang meliputi akurasi saklar 1, saklar 2 dan hasil data yang disimpan di EEPROM diperoleh akurasi sebesar 91,6% untuk saklar 1, 83,3% untuk saklar 2, dan 100% untuk hasil data di EEPROM.

**Kata kunci:** Smart Home, Lampu, Kebiasaan, *Naive Bayes*



## ABSTRACT

As one of the lighting tools that ease a human activities at dark state, lamp has very important function for daily lives. Inside the house lamp has a role as lighting tools that support house activities and nearly unreplaceable. However because the manual using with it, the resident in the house became very often encountering a problem. One of the example it's when the resident forgot to switch off the lamp, it make electricity wasted and impact on expense. These Research intended to make automatic lamp system using habit of occupants in the house to switch off and on the lamp inside it. System using habit of the resident and process it using Naïve Bayes algorithm to set the lamp condition. One of excess using Naïve Bayes algorithm is easy to use, have high accuracy and not need a lot of data. System using Arduino UNO microcontroller as main controller and three switch as main input. The switch control system mode and acquire data in real time. Acquired data also saved in Arduino UNO EEPROM so it can be used over and over again. All of the result of system that including switch 1 accuration, switch 2 accuration, and data result inside EEPROM, the switch 1 getting 91,6% accuracy, the switch 2 getting 83,3% accuracy, and 100% for the EEPROM.

Keyword: Smart Home, Lamp, Habit, *Naïve Bayes*



## DAFTAR ISI

SISTEM OTOMATISASI LAMPU RUMAH MENGGUNAKAN ALGORITME NAÏVE BAYES BERDASARKAN KEBIASAAN PENGHUNI RUMAH .....	i
PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
PRAKATA.....	iv
ABSTRAK.....	v
ABSTRACT .....	vi
DAFTAR ISI .....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR .....	xi
DAFTAR LAMPIRAN .....	xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan .....	2
1.4 Manfaat.....	2
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan .....	3
BAB 2 LANDASAN KEPUSTAKAAN .....	5
2.1 Tinjauan Pustaka.....	5
2.2 Dasar Teori .....	6
2.2.1 Smart Home .....	6
2.2.2 Pengertian Kebiasaan.....	6
2.2.3 Arduino IDE .....	6
2.2.4 Mikrokontroler Arduino UNO .....	7
2.2.5 Modul Relay .....	8
2.2.6 Real Time Clock (RTC) DS3231 .....	10
2.2.7 Naïve Bayes .....	10
BAB 3 METODOLOGI PENELITIAN .....	13
3.1 Alur Metodologi Penelitian.....	13



3.2 Studi dan Kajian Literatur .....	13
3.3 Analisis Kebutuhan .....	14
3.4 Pengumpulan Data .....	14
3.5 Perancangan Sistem.....	14
3.5.1 Perancangan Perangkat Lunak.....	14
3.5.2 Perancangan Perangkat Keras .....	15
3.6 Implementasi Sistem .....	15
3.7 Pengujian dan Analisis .....	15
3.8 Kesimpulan.....	15
<b>BAB 4 REKAYASA KEBUTUHAN.....</b>	<b>16</b>
4.1 Gambaran Umum Sistem.....	16
4.2 Analisis Kebutuhan .....	16
4.2.1 Kebutuhan Pengguna .....	16
4.2.2 Kebutuhan Fungsional.....	17
4.2.3 Kebutuhan Non Fungsional.....	18
4.3 Batasan Sistem.....	18
<b>BAB 5 PERANCANGAN DAN IMPLEMENTASI .....</b>	<b>20</b>
5.1 Perancangan Sistem.....	20
5.1.1 Perancangan Perangkat Keras .....	21
5.1.2 Perancangan Perangkat Lunak.....	23
5.1.2.1 Rancangan Program Arduino UNO .....	23
5.1.2.2 Rancangan Program Naïve Bayes.....	25
5.2 Implementasi Sistem .....	28
5.2.1 Implementasi Perangkat Keras .....	28
5.2.2 Implementasi Perangkat Lunak.....	29
5.2.2.1 Inisialisasi Awal Sistem.....	29
5.2.2.2 Saklar1.....	30
5.2.2.3 Saklar2.....	32
5.2.2.4 n_bayes .....	34
<b>BAB 6 PENGUJIAN DAN ANALISIS.....</b>	<b>38</b>
6.1 Pengujian Data EEPROM.....	38
6.1.1 Tujuan Pengujian.....	38





6.1.2	Prosedur Pengujian .....	38
6.1.3	Hasil dan Analisa Pengujian .....	39
6.2	Pengujian Sistem Utama .....	40
6.2.1	Tujuan Pengujian.....	40
6.2.2	Prosedur Pengujian .....	40
6.2.3	Hasil dan Analisa Pengujian .....	42
BAB 7	PENUTUP .....	45
7.1	Kesimpulan.....	45
7.2	Saran .....	45
DAFTAR	PUSTAKA.....	46
LAMPIRAN	.....	48

**DAFTAR TABEL**

Tabel 2.1 Spesifikasi Mikrokontroler Arduino UNO .....	8
Tabel 4.1 Kebutuhan Untuk Pengguna .....	16
Tabel 6.1 Hasil Pengujian dari Saklar 1 dari Data EEPROM .....	39
Tabel 6.2 Hasil Pengujian dari Saklar 2 dari Data EEPROM .....	39
Tabel 6.3 Hasil Pengujian dari Saklar 1 .....	42
Tabel 6.4 Hasil Pengujian dari Saklar 2 .....	43



## DAFTAR GAMBAR

Gambar 2.1 Mikokontroler Arduino UNO.....	7
Gambar 2.2 Modul Relay .....	9
Gambar 2.3 Struktur Relay.....	9
Gambar 2.4 RTC DS3231.....	10
Gambar 5.1 Alur Perancangan Sistem .....	20
Gambar 5.2 Skematika Pin RTC dan Arduino UNO .....	21
Gambar 5.3 Skematika Pin Modul Relay dan Arduino UNO .....	21
Gambar 5.4 Skematika Pin Switch dan Arduino UNO.....	22
Gambar 5.5 Diagram Alur Program Utama.....	23
Gambar 5.6 Isi Data EEPROM Arduino UNO Pada Program Utama .....	24
Gambar 5.7 Diagram Alur Program Naïve Bayes .....	25
Gambar 5.8 Rangkaian Perangkat Utama .....	28
Gambar 6.1 Hasil Hasil Data rekam EEPROM .....	38
Gambar 6.2 Hasil Pengaturan Waktu RTC .....	41
Gambar 6.3 Pengambilan Data Saat Saklar Training Menyala .....	42
Gambar 6.4 Hasil Output Saat Saklar Training Mati .....	42



## DAFTAR LAMPIRAN

LAMPIRAN A DATA HASIL PERCOBAAN .....	48
---------------------------------------	----





## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Salah satu teknologi terkini yang mendukung kehidupan sehari - hari manusia adalah *Smart Home*. Rumah pintar atau *Smart Home* merupakan sistem otomatisasi lanjut untuk mengontrol cahaya dan suhu, peralatan multimedia untuk memantau dan menyalakan sistem keamanan yang terkait dengan jendela, pintu dan fungsi fungsi lainnya. Rumah pintar dikatakan "*smart*" karena sistem komputernya yang bisa memantau banyak aspek dalam kehidupan sehari – hari (Bregman, 2010). Dalam *Smart Home* interaksi antara pengguna dengan sistem komputernya dibuat seminimal mungkin agar memenuhi aspek otomatisasi yang mempermudah kehidupan sehari – hari dalam suatu rumah. Rumah pintar menyediakan lingkungan hidup yang nyaman, aman, hemat energi, dan sehat di dalamnya (Lin, 2013).

Pada kehidupan sehari – hari lampu merupakan salah satu aspek penting di dalamnya, tidak terkecuali di dalam rumah. Akan tetapi saat menyalakan atau mematikan lampu penghuninya diharuskan untuk melakukannya secara manual. Hal tersebut akan menjadi hal yang kurang nyaman untuk dilakukan apabila terdapat banyak tombol lampu yang harus ditekan setiap kali penghuninya menyalakan atau mematikan. Terkadang juga terdapat lampu yang lupa dimatikan penghuninya akibat lalai ataupun terburu – buru melakukan aktifitas yang lain sehingga penggunaan listrik menjadi boros dan kurang efektif. Selain itu sistem peralatan yang dikendalikan oleh komputer memberikan efisiensi, keamanan, dan ketelitian yang lebih jika dibandingkan dengan pengerjaan secara manual oleh manusia (Turesna, 2015).

Dengan latar belakang yang telah disebutkan, penulis ingin membuat sistem yang dapat membantu kehidupan sehari - hari khususnya di dalam rumah yang berjudul "*Sistem Otomatisasi Lampu Rumah dengan Algoritme Naive Bayes Berdasarkan Kebiasaan Penghuni Rumah*". Pada sistem ini, perilaku penghuni rumah menjadi dasar keputusan apakah lampu akan menyala ataupun mati secara otomatis. Penghuni rumah diharuskan untuk mematikan dan menyalakan secara manual terlebih dahulu untuk mempelajari kebiasaan waktu penggunaan lampu tersebut, setelah itu data akan disimpan dan diproses sedemikian rupa dengan menggunakan Arduino UNO. Mikrokontroler ini digunakan karena mempunyai kelebihan diantaranya adalah penggunaannya yang mudah, harganya tidak terlalu mahal, dan sistemnya yang *open source* sehingga memudahkan dalam penggunaannya. Sedangkan dalam tahap pemrosesan data sistem yang dibuat menggunakan algoritme *Naive Bayes*. Algoritme ini merupakan metoda klasifikasi menggunakan probabilitas dan statistik yang memprediksi kemungkinan di masa yang akan datang berdasarkan pengalaman di masa sebelumnya. Salah satu keunggulan metode ini adalah akurasi yang dibanding metode klasifikasi yang lain.



Pada penelitian sebelumnya yang berjudul “Implementasi Algoritme *Naive Bayes* Menggunakan Aduino Uno Untuk Otomatisasi Lampu Ruangan-Berdasarkan Penghuni Rumah” dimana algoritme *Naive Bayes* digunakan untuk membuat sistem otomatisasi lampu berdasarkan waktu kebiasaan penghuni rumah menggunakan lampu tersebut (Pratama, 2018). Penelitian lainnya yang berjudul “Pengembangan Centralized Smarthome Simulator Sebagai Solusi Penghematan Biaya Listrik Menggunakan *K-Means* yang merancang sistem simulasi *smarthome* untuk efisiensi penggunaan alat listrik didalam rumah dengan menggunakan algoritme *K-Means*. Sistem bertumpu pada penggunaan waktu pada alat – alat listrik agar penggunaannya bisa disesuaikan secara efisien (Rinjani, 2017).

Diharapkan dengan adanya penelitian pembuatan sistem ini dapat mempermudah aktifitas penghuni rumah serta efisiensi penggunaan listrik didalamnya bisa ditingkatkan dengan lebih baik.

## 1.2 Rumusan Masalah

Dengan dasar latar belakang yang telah disebutkan, dapat dirumuskan permasalahan yaitu sebagai berikut:

1. Bagaimana cara mengimplementasikan algoritme *Naive Bayes* agar dapat mempelajari kebiasaan penghuni rumah saat menyalakan ataupun mematikan lampu rumah ?
2. Bagaimana cara menyimpan data masukan pengguna agar dapat digunakan secara berkelanjutan ?
3. Berapa tingkat akurasi data dengan penggunaan fungsi penyimpanan EEPROM Arduino Uno ?
4. Berapa tingkat akurasi Sistem dengan penggunaan algoritme *Naive Bayes* ?

## 1.3 Tujuan

Tujuan pembuatan sistem ini adalah sebagai berikut :

1. Mengetahui tata cara sistem dalam mengimplementasi penggunaan algoritme *Naive Bayes*.
2. Membuat sistem yang dapat digunakan secara berkelanjutan dengan menyimpan data masukan pengguna di EEPROM.
3. Mengetahui tingkat keakuratan data yang terekam di EEPROM Arduino UNO.
4. Mengetahui tingkat keakuratan sistem otomatisasi lampu rumah berdasarkan algoritme *Naive Bayes*.

## 1.4 Manfaat

Manfaat penelitian ini adalah membuat suatu sistem yang bisa memudahkan kehidupan sehari – hari penghuni suatu rumah dan juga meningkatkan efisiensi penggunaan listrik di dalamnya.



## 1.5 Batasan Masalah

1. Kebiasaan yang diamati hanya tertuju pada satu orang.
2. Pengamatan dilakukan terhadap kebiasaan menggunakan lampu baik dalam keadaan nyala atau mati
3. Pengamatan menggunakan satuan jam dan menit yang nantinya diolah sistem sesuai dengan rancangan yang dibuat.
4. Jumlah lampu yang digunakan dalam percobaan tidak lebih dari 2 buah.
5. Data yang didapatkan dalam satu hari maksimal 2 kali baik dalam keadaan lampu nyala ataupun mati.

## 1.6 Sistematika Pembahasan

### BAB 1 PENDAHULUAN

Berisikan tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan.

### BAB 2 LANDASAN KEPUSTAKAAN

Berisikan tentang penjelasan dasar teori yang digunakan dalam pembuatan sistem serta tinjauan pustaka dari penelitian – penelitian yang telah dilakukan sebelumnya.

### BAB 3 METODOLOGI PENELITIAN

Berisikan tentang penjelasan tentang langkah – langkah dalam melakukan penelitian, antara lain studi dan kajian literatur, analisis kebutuhan, perancangan sistem, implementasi, pengujian sistem, analisis hasil dan kesimpulan.

### BAB 4 REKAYASA KEBUTUHAN

Berisikan tentang penjelasan secara rinci terkait gambaran umum sistem, analisis kebutuhan fungsional dan kebutuhan non fungsional, dan batasan desain sistem.

### BAB 5 PERANCANGAN DAN IMPLEMENTASI

Berisikan tentang penjelasan mengenai proses perancangan pada sistem, proses implementasi perangkat keras, proses implementasi perangkat lunak dan juga algoritme *Naive Bayes* dalam melakukan klasifikasi.

### BAB 6 PENGUJIAN DAN ANALISIS

Berisikan tentang hasil pengujian terhadap sistem dan juga analisis terhadap hasil pengujian terhadap sistem.





## BAB 7 PENUTUP

Berisikan tentang uraian kesimpulan yang diperoleh dari hasil penelitian yang telah dilakukan, dan saran pengembangan lebih lanjut untuk sistem.



## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Tinjauan Pustaka

Pada penelitian yang dilakukan sebelumnya mengenai otomatisasi lampu ruangan oleh Septian Mukti Pratama dengan judul “Implementasi Algoritme *Naïve Bayes* Menggunakan Arduino Uno Untuk Otomatisasi Lampu Ruangan Berdasarkan Kebiasaan Dari Penghuni Rumah”. Penelitian ini membahas tentang sistem otomatisasi lampu ruangan dengan mengamati kebiasaan seorang penghuni rumah dalam menggunakan lampu ada di dalam rumah baik dalam keadaan mati ataupun nyala. Sistem menggunakan algoritme *Naïve Bayes* dalam mengolah data latih agar dapat memprediksi data yang dimasukan pengguna secara langsung. Untuk mengolah dan mengelola sistem, digunakan Arduino UNO dan RTC (Real Time Clock) sebagai sumber penentuan waktu secara *real time* (Pratama, 2018). Kelebihan dari penelitian ini terletak pada penggunaan waktu untuk mengukur kebiasaan dengan metode *Naïve Bayes*, karena metode tersebut hanya membutuhkan data dalam kuantitas yang sedikit. Sedangkan kekurangan penelitian ini terletak pada sistemnya yang tidak dapat menyimpan data sehingga penggunaanya terbatas.

Penelitian selanjutnya mempunyai judul “Pengembangan *Centralized Smarthome Simulator* Sebagai Solusi Penghematan Biaya Listrik Menggunakan Algoritme K-Means”. Penelitian ini membahas tentang *Smarthome Simulator* untuk solusi penghematan biaya listrik dengan menggunakan algoritme K-Means. Sistem dibuat secara otomatis mendeteksi penggunaan alat listrik melalui kebiasaan dan diolah dengan algoritme K-Means sehingga nyala dan mati alat – alat tersebut dapat diatur dan dikendalikan. Sistem mengoptimasi waktu penggunaan seluruh alat listrik sehingga tidak terjadi pemborosan sumber daya. Selain itu terdapat sistem lain yang dapat menghitung persentase biaya penggunaan alat – alat tersebut (Rinjani, 2017). Kelebihan utama sistem ini terdapat pada pengaturan sistem otomatisnya yang mempunyai cakupan luas untuk seluruh alat elektronik rumah sehingga mempermudah penggunaanya. Kekurangannya tereletak pada penerapan langsung sistem utamanya yang masih berupa simulator sehingga belum diimplementasikan secara langsung pada perangkat keras.

Berdasarkan penelitian-penelitian yang telah disebutkan diatas, penulis tertarik untuk melakukan penelitian yang sama dengan penelitian pertama dan kedua yaitu dengan melakukan penelitian untuk membuat sistem yang dapat melakukan kontrol secara otomatis berdasarkan kebiasaan di dalam rumah dengan pengembangan sistem yang lebih baik dengan judul “Sistem Otomatisasi Lampu Rumah dengan Algoritme *Naïve Bayes* Berdasarkan Kebiasaan Penghuni Rumah”. Sistem ini diharapkan dapat mengontrol nyala dan mati lampu secara otomatis dengan mempelajari kebiasaan penggunaan kesehariannya serta menyimpan data tersebut sehingga sistem dapat digunakan secara terus menerus.



## 2.2 Dasar Teori

Sub bab ini membahas dasar – dasar teori yang digunakan dalam pembuatan rancangan sistem utama.

### 2.2.1 Smart Home

Smart home merupakan ide dari pengguna rumah untuk dapat mengatur bagian rumahnya yang terintegrasikan ke suatu alat elektronik dengan tujuan meningkatkan kualitas hidup penggunaannya dan dapat mengelola rumah dengan baik (Masykur, 2016). Secara umum smart home memerlukan 3 syarat agar bisa disebut “*smart*”, yaitu :

1. Internal Network : berupa kabel,wireless.
2. Intelligent Control : berupa gateway untuk mengelola sistem.
3. Home Automation : mengatur dan mengelola.

Smart home dalam beroperasi dibantu oleh komputer untuk memberikan segala kenyamanan, keselamatan, keamanan dan penghemat energi yang berjalan secara otomatis dan terprogram melalui komputer pada gedung ataupun pada rumah tinggal kita. Sistem smart home dapat digunakan untuk mengendalikan hampir semua perlengkapan dan peralatan di rumah (Pratama, 2018).

### 2.2.2 Pengertian Kebiasaan

Kebiasaan adalah hasil belajar yang menunjukkan pola perilaku tertentu. Kebiasaan memiliki sifat atau corak seperti konsisten, otomatis, pasti, mudah, terintegrasi dengan pribadi individu. Kebiasaan juga bisa kuat atau lemah tergantung motivasi yang menggiringnya dari maksud dan tujuan kegiatan yang telah menjadi kebiasaan itu (Ardial, 2012). Kebiasaan memberi pengaruh besar terhadap kehidupan sehari-hari karena kebiasaan merupakan hal yang sering dilakukan berulang-ulang. Kebiasaan juga dapat membentuk watak suatu individu sehingga apabila ada orang yang mempunyai kebiasaan buruk maka orang tersebut akan terus melakukan hal tersebut sampai dia menghentikan sendiri kebiasaan tersebut.

### 2.2.3 Arduino IDE

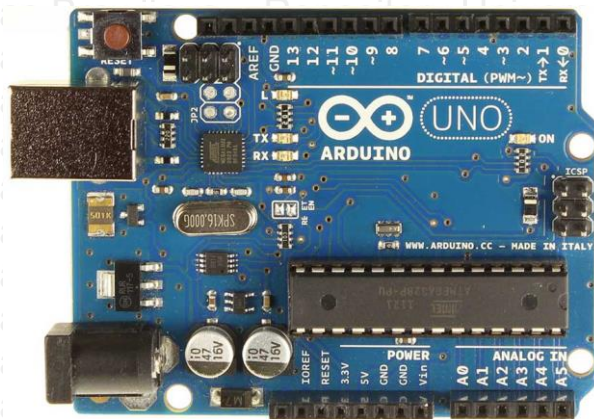
Arduino IDE merupakan perangkat lunak dari developer Arduino yang mempunyai fungsi menulis dan menyusun program ke dalam perangkat keras Arduino. Di dalamnya terdapat dua fungsi utama yang digunakan agar suatu program dapat berjalan. Fungsi pertama yaitu “Void Setup()” yang digunakan untuk menginisialisasi pengaturan awal dan fungsi kedua yaitu “Void Loop()” yang berfungsi menjalankan ulang program sampai terdapat perintah berhenti atau mikrokontroler kehilangan catu daya. Terdapat juga serial monitor yang berfungsi menampilkan hasil program yang telah di upload ke Arduino UNO. Saat program berhasil di upload LED TX dan RX bereaksi dengan kedipkan cahaya. Terdapat beberapa library juga yang dapat menjalankan beberapa fungsi



berbeda untuk mempermudah pembuatan suatu sistem. Library tersebut juga dapat di buat sendiri dan diupload agar dapat digunakan orang lain.

#### 2.2.4 Mikrokontroler Arduino Uno

Arduino Uno memiliki 14 pin digital input/output (6 diantaranya dapat digunakan sebagai output Pulse Width Modulation), 6 pin analog, sebuah port USB, sebuah power jack, sebuah ICSP header, dan sebuah tombol reset. Arduino Uno ini juga mudah dihubungkan ke sebuah komputer dengan sebuah kabel USB atau mensuplainya dengan sebuah adaptor AC ke DC atau menggunakan baterai (Arduino, 2017). Komponen – komponen tersebut akan dirakit sehingga membentuk sebuah mikrokontroler. Pada Gambar 2.1 merupakan gambar dari mikrokontroler Arduino Uno.



**Gambar 2.1 Mikrokontroler Arduino Uno**

Sumber : (Ecadio, 2016 )

Arduino Uno dilengkapi dengan static random-access memory (SRAM) berukuran 2 KB, Flash memory berukuran 32 KB, dan erasable programmable read-only memory (EEPROM). SRAM digunakan untuk menampung data atau hasil pemrosesan data selama Arduino menerima pasokan catu daya, flash memory untuk menaruh program yang dibuat serta EEPROM digunakan untuk menaruh program bawaan dari Arduino Uno dan sebagian lagi dapat dimanfaatkan untuk menaruh data secara permanen (Elektronika, 2016). Isi data yang tersimpan di EEPROM pada alamat tertentu dapat ditulis ulang dan dihapus sampai batas 300.000 kali, setelah itu EEPROM tidak bisa digunakan lagi untuk menyimpan data. Data maksimal yang dapat disimpan di EEPROM adalah sebesar 256 Byte.



Berikut ini adalah spesifikasi lengkap yang dimiliki Mikrokontroler Arduino Uno yang akan dijelaskan pada tabel 2.1.

**Tabel 2.1 Spesifikasi Mikrokontroler Arduino Uno**

Mikrokontroler	Atmega328P
<i>Operating Voltage</i>	5V
<i>Input Voltage (recommended)</i>	7 – 12V
<i>Input Voltage (limit)</i>	6 – 20 V
<i>Digital I/O pins</i>	14 ( <i>of which provide PWM output</i> )
<i>PWM Digital I/O Pins</i>	6
<i>Analog Input Pins</i>	6
<i>DC Current per I/O Pin</i>	20 mA
<i>DC Current for 3.3 V Pin</i>	50 mA
<i>Flash Memory</i>	32 KB (Atmega328P) <i>of which 0.5 KB used by bootloader</i>
<i>SRAM</i>	2 KB (Atmega328P)
<i>EEPROM</i>	1 KB (Atmega328P)
<i>Clock Speed</i>	16 MHZ
<i>LED_BULETIN</i>	13
<i>Length</i>	68.6 mm
<i>Width</i>	53.4 MM
<i>Weight</i>	25 g

**Sumber :** (Arduino, 2017)

Pada Chip Atmega328 telah mendukung untuk komunikasi I2C (TWI) dan SPI. Didalam Arduino Software (IDE), sudah termasuk Wire Library untuk memudahkan penggunaan bus I2C.

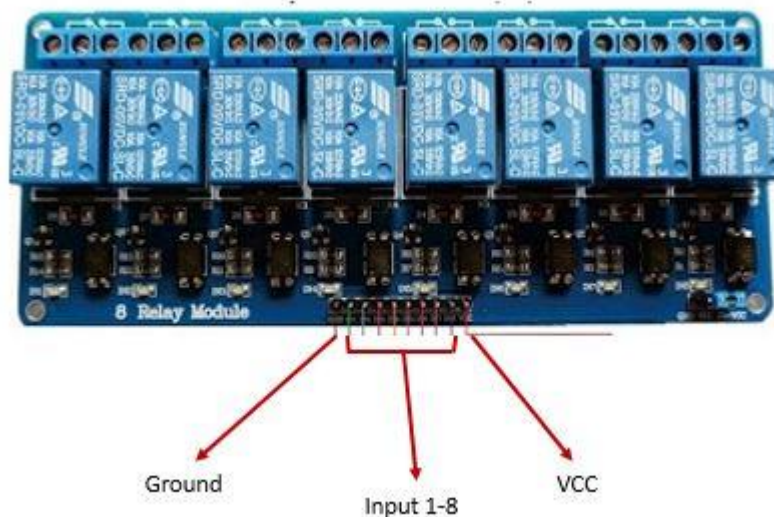
### 2.2.5 Modul Relay

Modul relay merupakan kumpulan beberapa komponen dan relay yang dapat langsung digunakan sebagai relay pada suatu rangkaian yang menggunakan mikrokontroler tanpa merangkai suatu komponen relay. Terdapat beberapa jack male yang berfungsi sebagai VCC, Ground, dan relay itu sendiri. Sedangkan pengertian relay sendiri yaitu saklar otomatis yang beroperasi dengan menggunakan prinsip elektromagnetik yang digunakan untuk memicu suatu kontaktor agar mempunyai kondisi “HIGH” atau “LOW”. Kontaktor akan tertutup



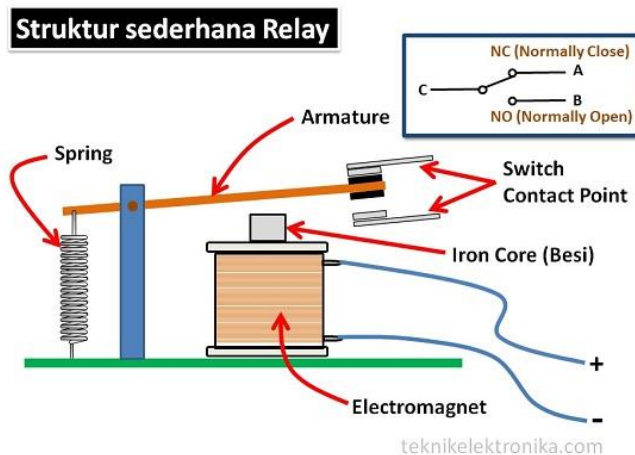
atau terbuka tergantung pada kondisi induksi magnet yang dihasilkan kumparan induktor yang dialiri arus listrik.

Berikut ini adalah tampilan bentuk modul relay 8 channel struktur dasar relay yang dapat dilihat pada Gambar 2.2 dan Gambar 2.3.



**Gambar 2.2 Modul Relay**

Sumber : (Belajarduino, 2016)



**Gambar 2.3 Struktur Relay**

Sumber : (Elektronika, 2016)

Terdapat dua macam jenis relay yaitu:

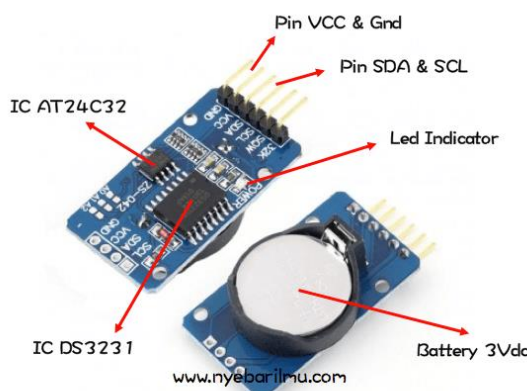
1. NC (Normally Close) yaitu jenis relay yang kondisi awal saklarnya dalam keadaan tertutup dan akan terbuka jika arus mengalir.
2. NO (Normally Open) yaitu jenis relay yang kondisi awal saklarnya dalam keadaan terbuka dan akan tertutup jika arus mengalir.



Perbedaannya dengan saklar terletak pada pemacu utamanya, untuk menggerakkan kontaktornya yang memicu kondisi “ON” atau “OFF”, saklar menggunakan cara manual untuk memicu kondisi tersebut. Sedangkan dalam penggunaan relay harus ada arus listrik yang mengalir ke kontaktor untuk memicu salah satu kondisi tersebut secara otomatis tergantung tipe relay yang digunakan.

### 2.2.6 Real Time Clock (RTC) DS3231

RTC DS3231 merupakan komponen IC penghitung yang berfungsi sebagai sumber data waktu yaitu data tahun, bulan, hari, jam, menit, sampai dengan detik. RTC DS3231 juga dilengkapi dengan sensor suhu yang dapat mendeteksi suhu sampai 85°C. Komponen dasar RTC DS3231 terdiri dari IC AT24C32 sebagai pengolah suhu dan IC DS3231 sebagai pengolah waktu. Terdapat juga pin VCC, Ground, SDA, dan SCL yang berfungsi sebagai penghubung dengan komponen elektronik yang lain serta baterai 3,6V sebagai catu daya cadangan. Berikut tampilan modul RTC jenis DS3231 pada Gambar 2.4.



**Gambar 2.4** RTC DS3231

**Sumber :** (Nyebarilmu, 2017)

SDA (Serial Data) dan SCL (Serial Clock) adalah dua pin unik yang digunakan untuk berkomunikasi secara I2C (Inter-Integrated Circuit). Inter Integrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didesain khusus untuk mengirim ataupun menerima data.

### 2.2.7 Naive Bayes

*Naive Bayes* adalah suatu metode klasifikasi secara probabilitas dan statistik yang memperkirakan peluang masa depan berdasarkan data sebelumnya serta mempunyai asumsi dan independensi dari masing masing data (Informatikalogi, 2017). Metode ini merupakan pengembangan teorema bayes dimana nilai *evidence* dihilangkan dari rumus utama. Dalam proses perhitungan, terdapat dua cara perhitungan tergantung dari tipe data yaitu data diskrit dan kontinyu.



Karena tipe data dalam sistem utama berupa data kontinu maka akan dijelaskan cara penghitungan dengan data kontinu yaitu:

1. Menentukan dan menghitung jumlah kelas dari masing-masing data.
2. Menghitung peluang prior dari jenis kelas masing – masing.

Untuk menghitung peluang dari masing masing kelas digunakan rumus sebagai berikut:

$$Px = \frac{nx}{nX} \quad (2.1)$$

Berdasarkan **Persamaan (2.1)**, dapat dilihat dimana  $Px$  merupakan prior suatu kelas,  $nx$  merupakan jumlah total suatu kelas, dan  $nX$  merupakan jumlah seluruh data.

3. Menghitung nilai rata – rata (*mean*) dari setiap kelas yang ada.

Untuk menghitung nilai rata – rata suatu kelas digunakan rumus sebagai berikut:

$$Mx = \frac{\sum_{i=1}^n kx}{nx} \quad (2.2)$$

Berdasarkan **Persamaan (2.2)**, dapat dilihat dimana  $Mx$  merupakan rata – rata suatu kelas,  $kx$  merupakan data latih suatu kelas, dan  $nx$  merupakan jumlah total suatu kelas.

4. Menghitung nilai standar deviasi dari tiap kelas yang ada.

Untuk menghitung nilai standar deviasi suatu kelas digunakan rumus sebagai berikut:

$$\sigma x = \sqrt{\frac{\sum_{i=1}^n (kx - Mx)^2}{nx - 1}} \quad (2.3)$$

Berdasarkan **Persamaan (2.3)**, dapat dilihat dimana  $\sigma x$  merupakan nilai standar deviasi suatu kelas,  $kx$  merupakan data latih suatu kelas,  $Mx$  merupakan rata – rata suatu kelas, dan  $nx$  merupakan jumlah total suatu kelas.

5. Menghitung nilai *Likelihood* dari data yang dibandingkan.

Untuk menghitung nilai likelihood dari suatu fitur yang diujikan dengan data kontinu digunakan rumus distribusi gaussian sebagai berikut:

$$P(Lx|Ly) = \frac{1}{\sqrt{2\pi\sigma x^2}} e^{-\frac{(Dx - Mx)^2}{2\sigma x^2}} \quad (2.4)$$





Berdasarkan **Persamaan (2.4)**, dapat dilihat dimana  $P(Lx|Ly)$  merupakan nilai likelihood suatu data  $x$  terhadap  $y$ ,  $\sigma_x$  adalah rata-rata data suatu kelas,  $e$  adalah nilai tetap yaitu 2,718,  $Dx$  adalah data suatu fitur yang akan diujikan, dan  $Mx$  merupakan rata-rata suatu kelas.

6. Menghitung peluang posterior tiap kelas yang ada.

Untuk menghitung peluang posterior suatu kelas digunakan rumus sebagai berikut:

$$Pk = (Px)(P(Lx|Ly)) \quad (2.5)$$

Berdasarkan **Persamaan (2.5)**, dapat dilihat dimana  $Pk$  merupakan peluang posterior suatu kelas,  $Px$  merupakan prior suatu kelas, dan  $P(Lx|Ly)$  merupakan nilai likelihood suatu data  $x$  terhadap  $y$ .



## BAB 3 METODOLOGI

### 3.1 Alur Metodologi Penelitian

Urutan dari metode penelitian adalah sebagai berikut:

1. Studi dan kajian literatur yang bertujuan untuk mencari tentang penelitian yang serupa dan juga dasar teori yang mendukung untuk melakukan penelitian tersebut.
2. Analisis kebutuhan untuk mencari tahu kebutuhan yang digunakan dalam proses penelitian.
3. Pengumpulan data, dimana data – data yang akan digunakan dalam penelitian akan dikumpulkan.
4. Perancangan sistem, dimana dasar rancangan sistem akan dibuat sedemikian rupa agar penelitian dapat berlangsung secara teratur.
5. Implementasi sistem, dimana penerapan perancangan sistem dilakukan.
6. Pengujian, dimana sistem yang dibuat akan diuji untuk mengetahui apakah hasilnya sudah sesuai dengan rancangan awal dan tujuan akhir pembuatan sistem.
7. Analisis hasil akhir, dimana hasil akhir sistem akan dimasukkan ke dalam data akhir.
8. Kesimpulan, dimana kesimpulan dari seluruh penelitian akan dibuat.

### 3.2 Studi dan Kajian Literatur

Meliputi pencarian teori – teori yang memperkuat asumsi dalam pembuatan rancangan sistem serta implementasinya sehingga kesalahan dapat dikurangi. Kumpulan teori tersebut didapatkan dari media seperti jurnal, paper, buku, dan internet yang mempunyai hubungan dengan penelitian ini. Adapun beberapa studi literatur yang digunakan adalah sebagai berikut:

1. Kumpulan teori pemrograman yang mendukung dan menggunakan perangkat lunak Arduino IDE sebagai acuan.
2. Datasheet RTC DS3231 serta media lain yang mengacu pada pembahasan komponen IC tersebut.
3. Datasheet mikrokontroler Arduino Uno serta media lain yang membahas tentangnya.
4. Media literatur tentang module relay.
5. Pengolahan data dengan algoritme *Naive Bayes* pada mikrokontroler Arduino UNO.



### 3.3 Analisis Kebutuhan

Merupakan analisa kebutuhan yang diperlukan untuk membentuk rancangan sistem implementasinya. Analisis kebutuhan sistem dibagi menjadi kebutuhan perangkat lunak dan kebutuhan perangkat keras.

Kebutuhan perangkat lunak yang digunakan sistem yaitu:

1. Arduino IDE 1.8.6 yang berfungsi mengolah program utama agar dapat dijalankan oleh Arduino UNO.
2. Arduino *library* yang berisi fungsi – fungsi dalam pemrograman Arduino IDE yang dapat digunakan untuk mendukung program utama.

Sedangkan kebutuhan perangkat keras meliputi piranti dan alat yang digunakan sebagai pengolah program dan output yang terdiri dari :

1. Mikrokontroler Arduino Uno sebagai perangkat pengatur dan penghubung keseluruhan sistem yang dibuat.
2. Modul Relay sebagai pengatur output sistem.
3. RTC DS3231 sebagai sumber data waktu.
4. Lampu 220VAC sebagai output sistem.
5. Saklar sebagai alat input data dan pengatur mode sistem.
6. Kabel Jumper sebagai penghubung antara pin.
7. Laptop sebagai perangkat yang menjalankan program Arduino IDE.
8. Resistor 10k ohm sebagai pengatur nilai pin saklar yang terhubung pada Arduino UNO

### 3.4 Pengumpulan Data

Merupakan proses pengumpulan data baik data latih maupun *real time*. Data yang dikumpulkan berupa waktu kebiasaan penggunaan lampu dalam keadaan nyala ataupun mati dalam waktu satu hari Pengambilan data sampel dilakukan selama enam hari secara beruntun dan dilakukan perseorangan pada suatu rumah. Setelah data diambil kemudian akan diolah dengan sistem yang telah dibuat.

### 3.5 Perancangan Sistem

Rancangan sistem dilakukan agar pembuatan sistem dapat dilakukan secara terstruktur agar kesalahan dapat diperkecil dan mempermudah dalam penerapan sistem.

#### 3.5.1 Perancangan Perangkat Lunak

Rancangan perangkat lunak sistem akan dijelaskan sebagai berikut:

1. Mendapatkan data sistem yaitu berupa kebiasaan menyalakan dan mematikan lampu dan menyimpannya.



2. Mengubah data yang didapatkan ke satuan menit.
3. Mengolah data yang didapatkan dengan algoritme *Naive Bayes*.
4. Mendapatkan hasil data dan menerapkannya pada hasil akhir yang berupa nyala dan mati lampu.

### 3.5.2 Perancangan Perangkat Keras

Rancangan perangkat keras ini akan dijelaskan sebagai berikut :

1. User akan memilih metode training atau non training melalui saklar training yang telah tersedia.
2. Apabila saklar training diaktifkan user memasukan data waktu dengan menggunakan saklar lampu yang telah disediakan.
3. Data yang dimasukan akan disimpan di EEPROM Arduino UNO.
4. Data yang direkam di EEPROM diubah menjadi menjadi satuan menit dan akan diolah menggunakan Algoritme *Naive Bayes* .
5. Hasil akan ditampilkan saat mode training off yang berupa mati dan nyala lampu.

### 3.6 Implementasi Sistem

Implementasi pada sistem merupakan pengaplikasian/pembuatan perangkat dari rancangan sistem awal yang telah dibuat sebelumnya. Terdapat gambar dan syntax program berikut penjelasannya.

### 3.7 Pengujian dan Analisis

Tahapan pengujian ini dilakukan agar dapat diketahui sistem yang telah dibuat dapat berfungsi dengan baik dan sesuai dengan perancangan awal. Pengujian yang akan dilakukan pada sistem yaitu antara lain:

1. Pengujian kesesuaian data yang telah disimpan.
2. Pengujian hasil data yang telah diolah.
3. Analisis persentase keberhasilan data.

### 3.8 Kesimpulan

Merupakan tahap penarikan kesimpulan dari seluruh sistem yang telah dibuat berdasarkan rumusan masalah serta tujuan awal sistem. Agar pengembangan sistem dapat dilanjutkan terdapat saran agar sistem yang telah dibuat semakin baik kedepannya.



## BAB 4 REKAYASA KEBUTUHAN

### 4.1 Gambaran Umum Sistem

Sistem otomatisasi ini dirancang untuk mempermudah penggunaan alat penerangan yang ada di dalam suatu rumah. Peralatan yang dikontrol oleh sistem ini berupa lampu. Sistem ini bekerja dengan input pengguna berupa waktu menyalakan dan mematikan lampu yang dimasukkan melalui saklar. Setelah input masuk, maka data akan disimpan dan diproses dengan algoritme *Naive Bayes*. Setelah itu data akan dibandingkan dengan waktu yang ada dalam RTC yang akan membuat keluaran sistem yang berupa hasil lampu yang menyala ataupun mati sesuai waktu yang dimasukkan penggunaannya.

### 4.2 Analisis Kebutuhan

Sub analisis ini akan menjelaskan keseluruhan kebutuhan pengguna dan sistemnya agar bekerja sesuai dengan tujuan penelitian ini yang terdiri dari kebutuhan pengguna, kebutuhan fungsional dan kebutuhan non fungsional.

#### 4.2.1 Kebutuhan Pengguna

Untuk memenuhi kebutuhan pengguna, sistem dibuat secara sederhana agar penggunaannya lebih mudah. Adapun hal-hal yang dibutuhkan oleh pengguna untuk sistem akan dijelaskan pada Tabel 4.1.

Tabel 4.1 Kebutuhan Untuk Pengguna

No	Fitur	Keterangan
1.	Menggunakan saklar training, saklar lampu 1 dan lampu 2.	<p>Pengguna diharuskan menyalakan saklar training untuk memulai merekam data.</p> <p>Pengguna diharuskan menyalakan dan mematikan saklar lampu 1 dan 2 untuk merekam data.</p> <p>Data waktu menyalakan akan direkam dan disimpan di EEPROM.</p> <p>Data maksimal adalah 1440 menit dan 2 kali penyimpanan dalam satu hari.</p>
2.	Sistem dapat mengetahui kebiasaan penghuni rumah dalam menyalakan dan mematikan lampu.	Jika sistem telah menerima masukan dari pengguna, sistem akan mengolah data waktu tersebut dan mengolah kebiasaan waktu pengguna tersebut



		dengan algoritme <i>Naive Bayes</i> . Waktu yang diambil untuk sample data adalah enam hari dan untuk satu pengguna.
3.	Menampilkan hasil pengolahan data algoritme <i>Naive Bayes</i> pada keluaran modul relay yang mengontrol nyala lampu.	Jika sistem telah memproses data dengan algoritme <i>Naive Bayes</i> , sistem akan menampilkan hasil pengolahan data yaitu berupa keluaran lampu nyala atau mati sesuai dengan masukan pengguna. Data bersifat tetap sampai ada masukan nilai baru dari pengguna.

#### 4.2.2 Kebutuhan Fungsional

Merupakan kebutuhan agar sistem dapat berjalan sesuai dengan rancangan awal pembuatannya. Kebutuhan fungsional adalah sebagai berikut:

1. Sistem dapat menerima masukan pengguna. Fungsi ini mengharuskan sistem dapat membaca inputan dari pengguna yang mengaktifkan lampu menggunakan saklar. Data waktu menit diambil dari waktu RTC DS3231 dan diteruskan ke Arduino IDE untuk diolah.
2. Sistem dapat mengolah data dan memberikan keputusan. Fungsi ini mengharuskan sistem dapat mengolah data masukan user menggunakan algoritme *Naive Bayes*. Proses ini akan mengolah data waktu lampu mati dan nyala agar dapat diproses oleh sistem selanjutnya.
3. Sistem dapat menyimpan data yang telah diambil. Fungsi ini mengharuskan sistem agar dapat menyimpan data yang telah diambil dan menyimpannya pada suatu media yang dapat diakses berulang kali tanpa kehilangan nilai awal pengambilan datanya.
4. Sistem dapat menampilkan hasil pengolahan data pada lampu pijar. Fungsi ini mengharuskan sistem dapat menampilkan hasil olahan data yang sesuai dengan algoritme *Naive Bayes*. Data yang didapatkan akan menentukan nyala atau mati lampu pijar.
5. Output modul relay sesuai masukan pengguna. Fungsi ini mengharuskan sistem menghasilkan keluaran berupa nyala atau mati sesuai dengan data yang telah diolah.



### 4.2.3 Kebutuhan Non Fungsional

Kebutuhan non fungsional ini terdiri dari kebutuhan perangkat lunak dan kebutuhan perangkat keras yang akan dijelaskan sebagai berikut:

1. Kebutuhan perangkat lunak pada sistem ini adalah sebagai berikut:
  - 1) Arduino IDE versi 1.8.7 yang berfungsi menjalankan mikrokontroler Arduino dan menjalankan program sesuai dengan syntax tertentu yang diberikan penggunaannya.
  - 2) Arduino library yaitu kumpulan perintah ekstra yang digunakan untuk mengendalikan suatu perangkat atau persamaan tertentu. Terdapat beberapa library yang digunakan yaitu "EEPROM" sebagai algoritme yang dapat menjalankan fungsi agar Arduino dapat menyimpan data di EEPROM sehingga data dapat digunakan lagi meskipun catu dayanya tidak ada dan library "DS3231" sebagai algoritme pengatur RTC DS3231.
2. Kebutuhan perangkat keras pada sistem ini berfungsi mendukung implementasi sistem. Alat yang diperlukan adalah sebagai berikut:
  - 1) Laptop yang berfungsi sebagai media pembuatan program utama dan sebagai penghubung antara perangkat lunak Arduino IDE dan perangkat keras Arduino Uno.
  - 2) Mikrokontroler Arduino Uno yang berfungsi sebagai pengolah data, pengatur, dan penghubung keseluruhan sistem.
  - 3) RTC (Real Time Clock) DS3231 yang berfungsi memberikan data waktu secara real time.
  - 4) Modul Relay empat channel yang berfungsi mengatur nyala dan mati lampu pijar.
  - 5) Resistor 10k ohm sebagai penyeimbang nilai tegangan.
  - 6) Jumper sebagai penghubung antar port.
  - 7) Breadboard sebagai penghubung antar jumper dan port.
  - 8) Lampu pijar sebagai keluaran sistem utama.

### 4.3 Batasan Sistem

Batasan sistem berfungsi agar pembuatan sistem dapat terfokus dengan tujuan awal pembuatan dan agar tujuan akhirnya dapat tercapai tanpa mengurangi atau menambahkan hal yang tidak diperlukan. Batasan-batasan sistem ini adalah sebagai berikut:

1. Kontrol sistem utama yaitu nyala dan mati lampu sebanyak 2 buah.
2. Fitur yang digunakan sebagai masukan data dalam 1 hari dimasukkan dengan penggunaan saklar oleh pengguna.



3. Data masukan langsung direkam ketika saklar taining menyala.
4. Dalam satu hari, data hanya direkam dua kali untuk nyala dan mati lampu
5. Data sampel yang digunakan sebagai pengolahan data menggunakan algoritme *Naive Bayes* diambil selama enam hari untuk satu pengguna.
6. Menit yang digunakan dalam data bernilai minimum menit ke 1 dan maksimal menit ke 1440.
7. Sistem dapat bekerja ketika saklar training, lampu 1, dan lampu 2 digunakan.

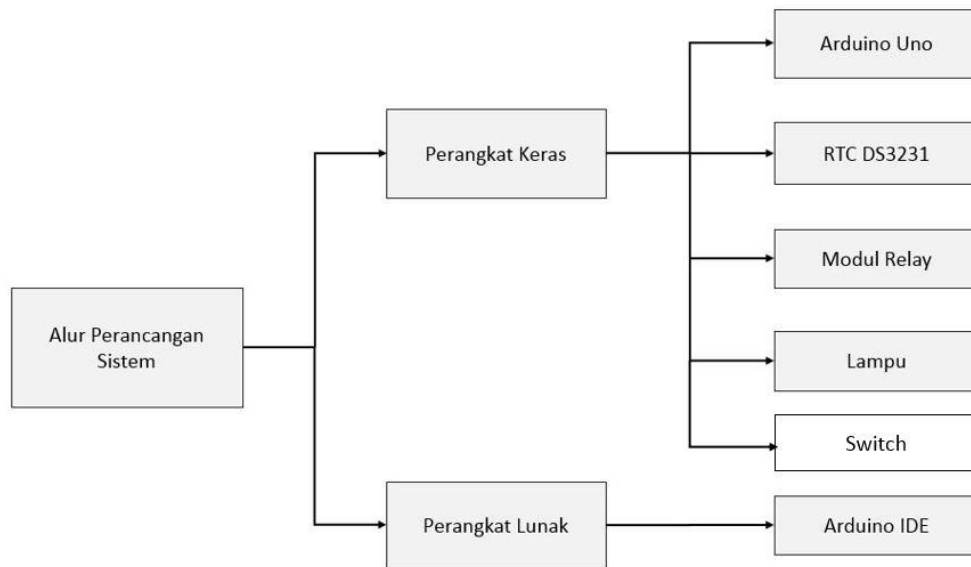




## BAB 5 PERANCANGAN DAN IMPLEMENTASI

### 5.1 Perancangan Sistem

Perancangan pada sistem meliputi rancangan perangkat keras yang memberikan input dan output sistem utama dan perangkat lunak yang berfungsi mengatur kerja perangkat keras agar sesuai dengan hasil yang diinginkan. Alur perancangan sistem dijelaskan pada gambar 5.1 berikut.



**Gambar 5.1 Alur Perancangan Sistem.**

Perancangan sistem terbagi menjadi dua bagian yaitu perangkat keras dan perangkat lunak. Pada bagian perangkat keras terdiri dari *Arduino Uno* sebagai penghubung dan pengolah seluruh sistem. Setelah itu terdapat RTC DS3231 sebagai sumber acuan waktu sistem, Modul Relay sebagai pengatur utama nyala lampu sistem, Switch (Saklar) sebagai pengatur mode, pengambil data waktu dan pengatur modul relay, dan lampu yang berfungsi sebagai output sistem.

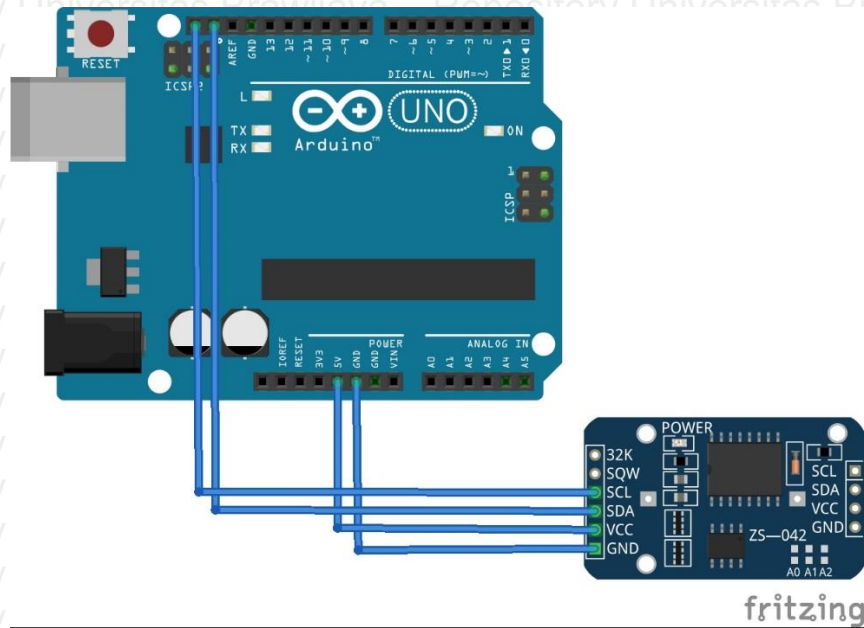
Pada bagian perangkat lunak terdapat *Arduino IDE* yang berfungsi sebagai program pengolah seluruh sistem. Didalamnya terdapat beberapa library yang digunakan untuk menjalankan perintah tertentu terhadap suatu perangkat atau algoritme tertentu.



### 5.1.1 Perancangan Perangkat Keras

Pada bagian ini dijabarkan hubungan skematik antara pin-pin yang digunakan untuk membangun keseluruhan sistem. Terdapat beberapa gambar skematik sistem dan penjelasan tentang hubungan antar komponennya.

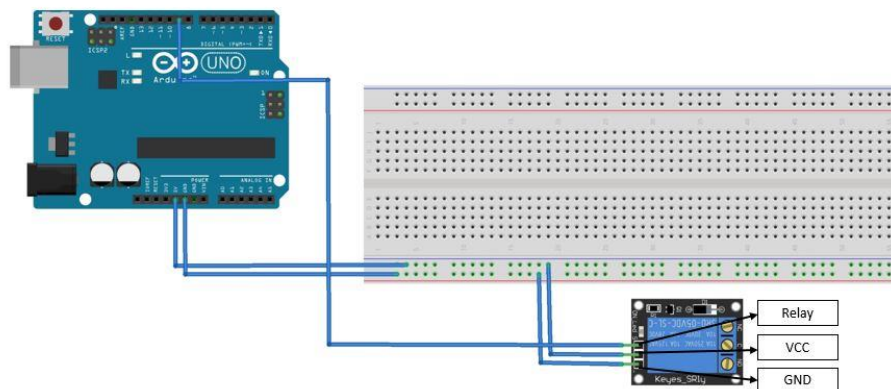
Bagian pertama adalah antara *Arduino Uno* dan RTC DS3231 yang akan ditunjukkan pada gambar 5.2 berikut.



**Gambar 5.2 Skematika Pin RTC dan Aduino Uno.**

Pada gambar 5.2 bisa dilihat pin GND (Ground) pada RTC terhubung dengan pin GND (Ground) *Arduino Uno*, pin VCC dengan pin 5V sebagai sumber tegangan, pin SDA RTC dengan SDA, dan pin SCL RTC dengan SCL.

Bagian kedua adalah antara *Arduino Uno* dan Modul Relay yang akan ditunjukkan pada gambar 5.3 berikut.

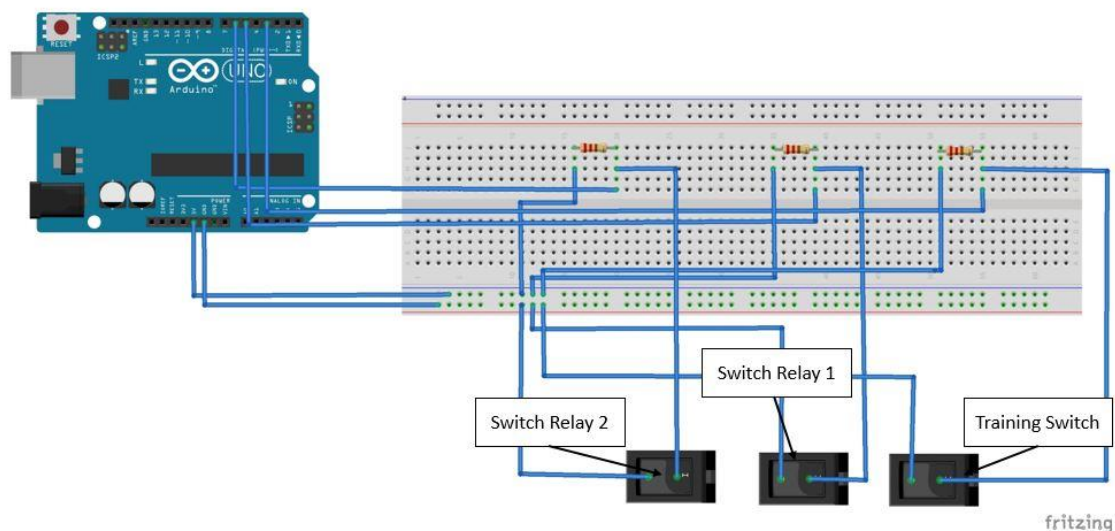


**Gambar 5.3 Skematika Pin Modul Relay dan Aduino Uno.**



Pada gambar 5.3 bisa dilihat pin GND Modul Relay tersambung melalui *Breadboard* dan berakhir terhubung pada pin GND *Arduino Uno*, pin VCC Modul-Relay tersambung melalui *Breadboard* dan berakhir terhubung pada pin 5V *Arduino Uno*, dan pin Relay pada Port 9. Dalam sistem terdapat Dua relay yang terhubung, relay kedua terhubung pada port 10.

Bagian ketiga adalah skematik switch yang berfungsi mengambil data waktu dan mengontrol Modul Relay secara langsung yang ditunjukkan pada gambar 5.4 berikut.



**Gambar 5.4 Skematika Pin Switch dan Aduino Uno.**

Terdapat tiga switch yang ada di sistem. Training Switch berfungsi sebagai pengatur apakah training mode akan nyala atau mati, jika training mode menyala maka data waktu dari switch relay 1 dan 2 saat di *trigger* baik nyala ataupun mati akan direkam di sistem, sebaliknya jika training mode mati maka output data dari switch relay 1 dan 2 tidak akan direkam. Akan tetapi data diolah dengan Algoritme *Naïve Bayes* untuk memicu nyala relay 1 dan 2.

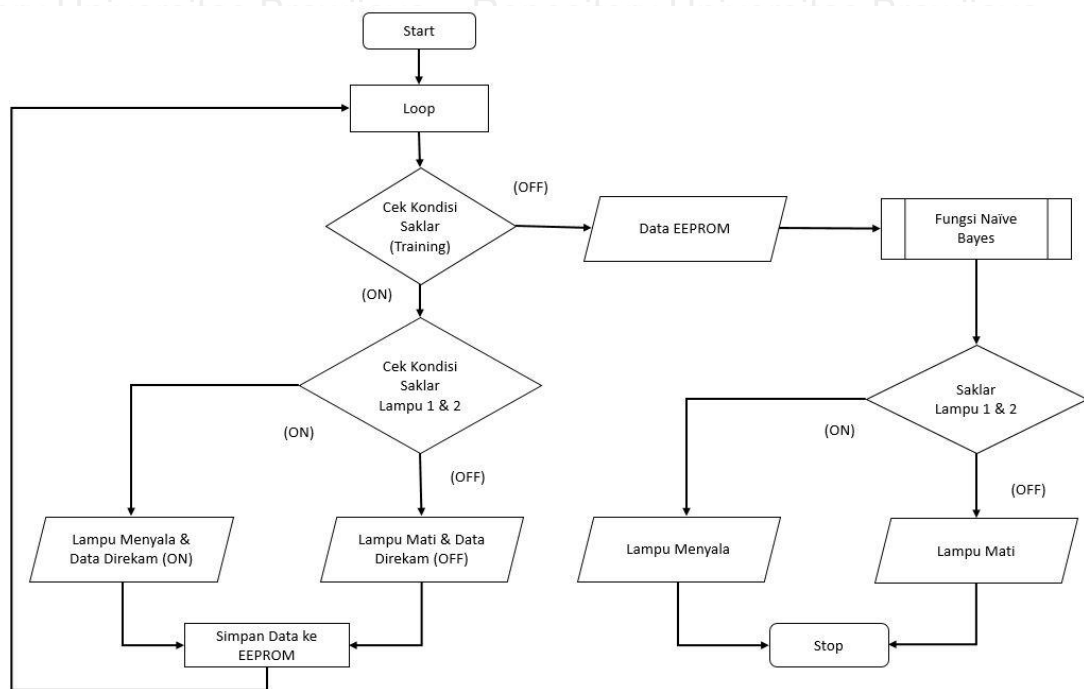


### 5.1.2 Perancangan Perangkat Lunak

Pada sub-bab ini menjelaskan rancangan program dari sistem, metode perhitungan, dan Algoritma yang digunakan untuk mengolah data yaitu *Naïve Bayes*.

#### 5.1.2.1 Rancangan Program Arduino Uno

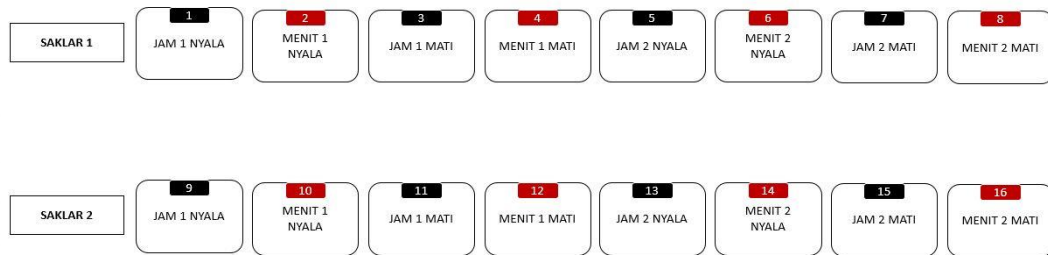
Rancangan program Arduino Uno ini meliputi program keseluruhan utama sistem. Detail program bisa dilihat pada gambar diagram 5.5.



**Gambar 5.5 Diagram Alur Program Utama.**

Program Utama dimulai dari pengecekan kondisi saklar(Training), jika kondisi ON maka saklar(Lampu 1 dan 2) akan dicek kondisinya dalam keadaan nyala atau mati dan kemudian merekam dan menyimpan datanya ke EEPROM dengan alamat tertentu. Setelah data disimpan ke EEPROM program kembali ke awal untuk mengecek kembali kondisi saklar. Apabila kondisi saklar(Training) OFF maka data dari EEPROM akan dicek, Setelah dicek data diolah menggunakan Algoritme *Naïve Bayes*. Apabila waktu data olahan dan waktu sekarang terpenuhi maka lampu akan menyala atau mati tergantung waktu dan kondisi yang terpenuhi.

Terdapat alamat tertentu yang digunakan sebagai penyimpanan data di EEPROM Arduino UNO. Untuk saklar 1 menggunakan alamat 1, 2, 3, 4, 5, 6, 7, dan 8 sedangkan saklar 2 menggunakan alamat 9, 10, 11, 12, 13, 14, 15, dan 16. Penjelasan isi data EEPROM bisa dilihat pada gambar 5.6.



**Gambar 5.6 Isi Data EEPROM Arduino Uno Pada Program Utama.**

Terdapat total 16 alamat yang digunakan untuk penyimpanan data. Setiap alamat menyimpan data dengan ukuran kurang dari 256 byte dikarenakan data maksimum yang dapat ditampung pada satu alamat EEPROM Arduino Uno sebesar 256 byte. Jika nilai yang ada di dalam satu alamat melebihi 256 byte maka nilainya akan berubah dan tidak sesuai dengan nilai perhitungan waktu yang dimasukkan oleh pengguna.

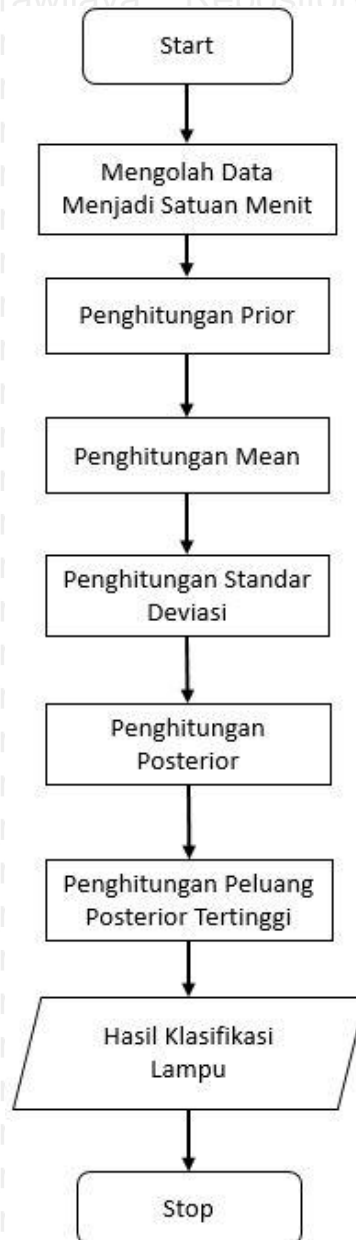
Pada Gambar 5.6 bisa dilihat tiap alamat menyimpan kondisi yang berbeda dari saklar 1 dan 2. Dimulai dari alamat 1 yang menyimpan data jam nyala saklar 1 dengan rentang waktu antara pukul 01.00 – 12.00 dan alamat 2 menyimpan data menit antara menit 1 – 60. Sedangkan alamat 3 menyimpan data jam mati saklar 1 dengan rentang waktu antara pukul 01.00 – 12.00 dan alamat 5 menyimpan data menit antara menit 1 – 60.

Untuk alamat 5 data yang disimpan berupa jam nyala saklar 1 dengan rentang waktu antara pukul 13.00 – 24.00 dan alamat 6 menyimpan data menit antara menit 1 – 60. Sedangkan alamat 7 menyimpan data jam mati saklar 1 dengan rentang waktu antara pukul 13.00 – 24.00 dan alamat 8 menyimpan data menit antara menit 1 – 60. Kondisi yang sama juga berlaku pada saklar 2 dengan alamat 9, 10, 11, 12, 13, 14, 15, dan 16.



### 5.1.2.2 Rancangan Program *Naïve Bayes*

Rancangan program *Naïve Bayes* menjelaskan tentang urutan perhitungan data menggunakan metode tersebut. Detail alur bisa dilihat pada gambar 5.7.



Gambar 5.7 Diagram Alur Program *Naïve Bayes*.

Program dimulai dengan membaca data dari EEPROM dan mengubahnya menjadi satuan menit. Setelah itu dilanjutkan dengan perhitungan nilai prior. Nilai prior merupakan nilai terjadinya suatu kelas dengan membagi banyaknya data dalam suatu kelas dengan keseluruhan jumlah data yang ada. Untuk menghitung nilai prior digunakan **Persamaan 2.1**.



Contoh perhitungannya adalah sebagai berikut:

$$P l_{nyala} = \frac{nx}{nX} = \frac{12}{24} = 0,5$$

$$P l_{mati} = \frac{nx}{nX} = \frac{12}{24} = 0,5$$

Setelah nilai prior diketahui maka berlanjut ke nilai rata – rata (*mean*) dari dua kelas sistem yang ada yaitu nyala dan mati terhadap saklar 1. Untuk menghitung nilai mean dari suatu kelas digunakan **Persamaan 2.2**.

Contoh perhitungannya adalah sebagai berikut:

$$M_{nyala} = \frac{\sum_{i=1}^n kx}{nx} = \frac{1000+245+1002+255+\dots+1000+235}{12} = 630,66$$

$$M_{mati} = \frac{\sum_{i=1}^n kx}{nx} = \frac{347+1335+337+1325+\dots+360+1315}{12} = 838,75$$

Setelah nilai *mean* diketahui selanjutnya dilanjutkan dengan perhitungan standar deviasi dari dua kelas yaitu nyala dan mati terhadap saklar 1. Untuk menghitung nilai standar deviasi dari suatu kelas digunakan **Persamaan 2.3**.

Contoh perhitungannya adalah sebagai berikut:

$$\begin{aligned} \sigma_{nyala} &= \sqrt{\frac{\sum_{i=1}^n (kx - Mx)^2}{nx - 1}} \\ &= \sqrt{\frac{(1000 - 630,66)^2 + (245 - 630,66)^2 + (1002 - 630,66)^2 + \dots + (1000 - 630,66)^2 + (235 - 630,66)^2}{12 - 1}} \\ &= 396,89 \end{aligned}$$

$$\begin{aligned} \sigma_{mati} &= \sqrt{\frac{\sum_{i=1}^n (kx - Mx)^2}{nx - 1}} \\ &= \sqrt{\frac{(347 - 838,75)^2 + (1335 - 838,75)^2 + (363 - 838,75)^2 + \dots + (360 - 838,75)^2 + (1315 - 838,75)^2}{12 - 1}} \\ &= 505,02 \end{aligned}$$

Setelah nilai standar deviasi ditemukan selanjutnya adalah menghitung nilai *Likelihood* dari kelas data nyala dan mati. Data yang digunakan untuk perbandingan adalah 1020 dalam menit yang jika di rubah dalam satuan jam yaitu pukul 17.00. Untuk menghitung nilai yang bentuknya kontinu digunakan persamaan distribusi Gaussian. Rumus distribusi Gaussian terdapat pada **Persamaan 2.4**.



Contoh perhitungannya adalah sebagai berikut:

$$\begin{aligned} P(L = 1020 | L \text{ nyala}) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(Dx-Mx)^2}{2\sigma^2}} \\ &= \frac{1}{\sqrt{2 \times 3,14 \times 396,89^2}} e^{-\frac{(1020-630,66)^2}{2 \times 396,89^2}} \\ &= 0,60 \end{aligned}$$

$$\begin{aligned} P(L = 1020 | L \text{ mati}) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(Dx-Mx)^2}{2\sigma^2}} \\ &= \frac{1}{\sqrt{2 \times 3,14 \times 505,02^2}} e^{-\frac{(1020-838,75)^2}{2 \times 505,02^2}} \\ &= 0,74 \end{aligned}$$

Setelah nilai likelihood dari kelas nyala dan mati diketahui selanjutnya adalah menghitung nilai peluang posterior dari keduanya. Untuk perhitungannya digunakan **Persamaan 2.5**.

Contoh perhitungannya adalah sebagai berikut:

$$\begin{aligned} P | 1020 \text{ nyala} &= (Px)(P(Lx|Ly)) \\ &= (0,5)(0,60) \\ &= 0,30 \end{aligned}$$

$$\begin{aligned} P | 1020 \text{ mati} &= (Px)(P(Lx|Ly)) \\ &= (0,5)(0,74) \\ &= 0,37 \end{aligned}$$

Dari hasil dari perhitungan peluang posterior tersebut, didapatkan nilai 0,30 untuk kondisi nyala dan 0,37 pada kondisi mati saat menit 1020 atau pukul 17.00. Didapatkan kesimpulan yaitu kebiasaan penggunaan pada rentang waktu tersebut adalah kondisi **mati**. Hasil tersebut dikarenakan nilai dari peluang posterior kondisi mati lebih besar dari kondisi saat nyala.



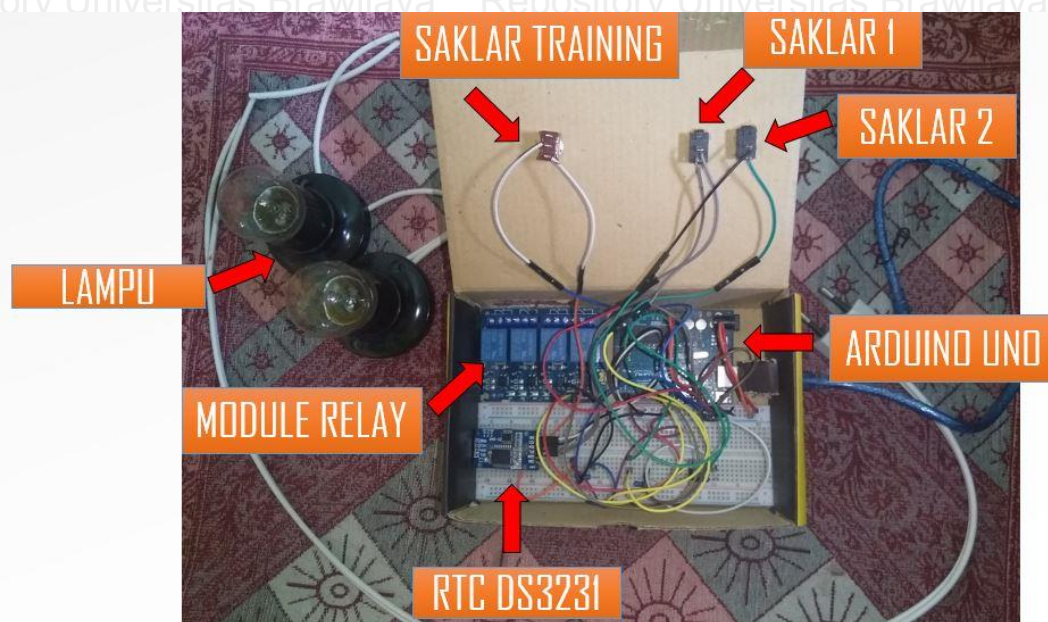


## 5.2 Implementasi Sistem

Implementasi Sistem merupakan tahap penerapan seluruh rancangan yang telah dibuat untuk membangun keseluruhan sistem. Terdapat dua implementasi yaitu perangkat keras dan lunak yang menjadi struktur pembangunan sistem yang utuh.

### 5.2.1 Implementasi Perangkat Keras

Sesuai dengan rancangan pada bab sebelumnya terdapat beberapa komponen yang akan dirancang guna mendapatkan sistem yang utuh. Rangkaian keseluruhan sistem dapat dilihat pada gambar 5.8.



**Gambar 5.8 Rangkaian Perangkat Utama.**

Pada gambar 5.8 dasar-dasar komponen utama sistem terdiri dari :

1. Arduino Uno sebagai pengatur keseluruhan sistem dan penghubung antara perangkat keras dan lunak.
2. Saklar sebagai input utama sistem.
3. RTC DS3231 sebagai sumber pengambilan data waktu secara real time.
4. Modul Relay sebagai pengatur output utama sistem yang menghubungkan dengan lampu.
5. Lampu sebagai output utama sistem.

Seluruh komponen menopang antara satu dengan lainnya sehingga dapat membuat satu sistem utuh yang dapat menjalankan program yang dijalanannya.



## 5.2.2 Implementasi Perangkat Lunak

Merupakan penerapan program keseluruhan sistem yang menggerakkan perangkat sistem. Terdapat beberapa bagian dari program yaitu inialisasi awal sistem, saklar1, saklar2 dan n\_bayes.

### 5.2.2.1 Inialisasi Awal Sistem

Program inialisasi awal sistem berisikan library sistem serta variable – variable yang digunakan. Detail dari syntax program adalah sebagai berikut:

Baris	PROGRAM
1	#include <DS3231.h>
2	#include <EEPROM.h>
3	#include "Arduino.h"
4	const int training = 3;
5	const int saklar1 = 5;
6	const int saklar2 = 6;
7	const int relay1 = 9;
8	const int relay2 = 10;
9	DS3231 rtc(SDA, SCL);
10	Time t;
11	int training_state = 0;
12	int s1_state = 0;
13	int s2_state = 0;
14	bool t_v = 0;
15	bool z_v = 0;
16	bool t_v2 = 0;
17	bool z_v2 = 0;

Library DS3231 berfungsi sebagai library bagi RTC agar dapat digunakan, sedangkan EEPROM digunakan agar sistem dapat menyimpan data dalam EEPROM meskipun dalam keadaan tanpa daya dan menggunakannya. Selanjutnya inialisasi variable port – port yang digunakan untuk menghubungkan antar sistem perangkat keras. Terakhir yaitu inialisasi nilai awal dari beberapa variable yang digunakan di bagian fungsi yang lain dalam program.



### 5.2.2.2 Saklar1

Program saklar1 merupakan program yang berisi perintah sistem dalam menjalankan saklar training, saklar 1, dan menyimpan datanya ke EEPROM dengan alamat tertentu. Susunan syntax programnya adalah sebagai berikut :

Baris	PROGRAM
1	<code>void saklar_1(){</code>
2	<code>  training_state = digitalRead(3);</code>
3	<code>  s1_state = digitalRead(5);</code>
4	<code>  t = rtc.getTime();</code>
5	<code>  int w_1 = t.hour;</code>
6	<code>  int m_1 = t.min;</code>
7	<code>  if (training_state == HIGH){</code>
8	<code>    Serial.println("Training Lampu Aktif");</code>
9	<code>    if(s1_state == LOW){</code>
10	<code>      digitalWrite(relay1,HIGH);</code>
11	<code>      if(t_v == 0 &amp;&amp; w_1 &lt;= 12){</code>
12	<code>        EEPROM.write(1,w_1);</code>
13	<code>        EEPROM.write(2,m_1);</code>
14	<code>        Serial.println("Data1 REC(Ny) Waktu ");</code>
15	<code>        Serial.println(w_1);</code>
16	<code>        Serial.println(m_1);</code>
17	<code>        t_v = 1;</code>
18	<code>      }else if(t_v == 0 &amp;&amp; w_1 &gt;= 13){</code>
19	<code>        EEPROM.write(3,w_1);</code>
20	<code>        EEPROM.write(4,m_1);</code>
21	<code>        Serial.println("Data2 REC(Ny) Waktu");</code>
22	<code>        Serial.println(w_1);</code>
23	<code>        Serial.println(m_1);</code>
24	<code>        t_v = 1;</code>
25	<code>      }else if(t_v == 1 &amp;&amp; s1_state == LOW){</code>
26	<code>        Serial.println("Stop");</code>
27	<code>      }</code>
28	<code>      }else if(t_v == 1 &amp;&amp; s1_state == HIGH){</code>
29	<code>        Serial.println("Data Ny UNREC");</code>
30	<code>        t_v = 0;</code>
31	<code>    }</code>
32	<code>  else if (s1_state == HIGH){</code>
33	<code>    digitalWrite(relay1,LOW);</code>



```

34     if(z_v == 0 && w_1 <= 12){
35         EEPROM.write(5,w_1);
36         EEPROM.write(6,m_1);
37         Serial.println("Data1 REC(Mt) Waktu ");
38         Serial.println(w_1);
39         Serial.println(m_1);
40         z_v = 1;
41     }else if(z_v == 0 && w_1 >= 13){
42         EEPROM.write(7,w_1);
43         EEPROM.write(8,m_1);
44         Serial.println("Data2 REC(Mt) Waktu");
45         Serial.println(w_1);
46         Serial.println(m_1);
47         z_v = 1;
48     }else if(z_v == 1 && s1_state == HIGH){
49         Serial.println("Stop2");
50     }
51     }else if(z_v == 1 && s1_state == LOW){
52         Serial.println("Data Mt UNREC");
53         z_v = 0;
54     }
55 }else if(training_state == LOW){
56     Serial.println(" Training Lampu Mati");
57     n_bayes();
58 }
59     delay(1800);
60 }

```

Program dimulai dengan pembuatan fungsi saklar1. Di dalam fungsi saklar1 program akan mengecek kondisi port 3 dan 5, jika kondisinya terpenuhi maka program akan mengecek kondisi saklar training dan saklar 1. Jika kondisi saklar training "HIGH" maka program dalam kondisi training "ON" dimana akan mengecek kondisi saklar 1 apakah dalam keadaan "LOW" atau "HIGH" dan merekam waktu saat kondisi tersebut terpenuhi. Data yang direkam disimpan ke EEPROM dengan alamat tertentu. Bila kondisi saklar training "LOW" maka program dalam kondisi training "OFF" dimana hasil program training akan diolah dan digunakan sebagai trigger lampu apakah dalam kondisi menyala atau mati.



### 5.2.2.3 Saklar2

Program saklar2 merupakan program yang berisi perintah sistem dalam menjalankan saklar training, saklar 2, dan menyimpan datanya ke EEPROM dengan alamat tertentu. Susunan syntax programnya adalah sebagai berikut:

Baris	PROGRAM
1	<code>void saklar_2(){</code>
2	<code>  training_state = digitalRead(3);</code>
3	<code>  s2_state = digitalRead(6);</code>
4	<code>  t = rtc.getTime();</code>
5	<code>  int w_2 = t.hour;</code>
6	<code>  int m_2 = t.min;</code>
7	<code>  if (training_state == HIGH){</code>
8	<code>    Serial.println("Training Lampu Aktif");</code>
9	<code>    if(s2_state == LOW){</code>
10	<code>      digitalWrite(relay2,HIGH);</code>
11	<code>      if(t_v2 == 0 &amp;&amp; w_2 &lt;= 12){</code>
12	<code>        EEPROM.write(9,w_2);</code>
13	<code>        EEPROM.write(10,m_2);</code>
14	<code>        Serial.println("Data3 REC(Ny) Waktu ");</code>
15	<code>        Serial.println(w_2);</code>
16	<code>        Serial.println(m_2);</code>
17	<code>        t_v2 = 1;</code>
18	<code>      }else if(t_v2 == 0 &amp;&amp; w_2 &gt;= 13){</code>
19	<code>        EEPROM.write(11,w_2);</code>
20	<code>        EEPROM.write(12,m_2);</code>
21	<code>        Serial.println("Data4 REC(Ny) Waktu");</code>
22	<code>        Serial.println(w_2);</code>
23	<code>        Serial.println(m_2);</code>
24	<code>        t_v2 = 1;</code>
25	<code>      }else if(t_v2 == 1 &amp;&amp; s2_state == LOW){</code>
26	<code>        Serial.println("Stop");</code>
27	<code>      }</code>
28	<code>      }else if(t_v2 == 1 &amp;&amp; s2_state == HIGH){</code>
29	<code>        Serial.println("Data Ny UNREC");</code>
30	<code>        t_v2 = 0;</code>
31	<code>    }</code>
32	<code>  else if (s2_state == HIGH){</code>
33	<code>    digitalWrite(relay2,LOW);</code>



```

34     if(z_v2 == 0 && w_2 <= 12) {
35         EEPROM.write(13,w_2);
36         EEPROM.write(14,m_2);
37         Serial.println("Data3 REC(Mt) Waktu ");
38         Serial.println(w_2);
39         Serial.println(m_2);
40         z_v2 = 1;
41     }else if(z_v2 == 0 && w_2 >= 13){
42         EEPROM.write(15,w_2);
43         EEPROM.write(16,m_2);
44         Serial.println("Data4 REC(Mt) Waktu");
45         Serial.println(w_2);
46         Serial.println(m_2);
47         z_v2 = 1;
48     }else if(z_v2 == 1 && s2_state == HIGH){
49         Serial.println("Stop2");
50     }
51     }else if(z_v2 == 1 && s2_state == LOW){
52         Serial.println("Data Mt UNREC");
53         z_v2 = 0;
54     }
55 }else if(training_state == LOW){
56     Serial.println(" Training Lampu Mati");
57     n_bayes();
58 }
59     delay(1800);
60 }

```

Program dimulai dengan pembuatan fungsi saklar2. Di dalam fungsi saklar2 program akan mengecek kondisi port 3 dan 6, jika kondisinya terpenuhi maka program akan mengecek kondisi saklar training dan saklar 2. Jika kondisi saklar training "HIGH" maka program dalam kondisi training "ON" dimana akan mengecek kondisi saklar 2 apakah dalam keadaan "LOW" atau "HIGH" dan merekam waktu saat kondisi tersebut terpenuhi. Data yang direkam disimpan ke EEPROM dengan alamat tertentu. Bila kondisi saklar training "LOW" maka program dalam kondisi training "OFF" dimana hasil program training akan diolah dan digunakan sebagai trigger lampu apakah dalam kondisi menyala atau mati.



### 5.2.2.4 n\_bayes

Program n\_bayes berisikan algoritme *Naive Bayes* yang digunakan untuk mengolah keseluruhan data yang sudah diambil. Ada beberapa bagian sistem di satu fungsi n\_bayes yaitu pembacaan nilai EEPROM, pencarian nilai prior, nilai mean, nilai standar deviasi, dan distribusi Gaussian dari saklar 1 dan 2. Syntax programnya adalah sebagai berikut:

#### 1. Syntax pembacaan nilai EEPROM:

Baris	PROGRAM
1	//saklar 1
2	int j_1 = (EEPROM.read(1))*60+(EEPROM.read(2)); //Nyala
3	int j_2 = (EEPROM.read(3))*60+(EEPROM.read(4)); //Nyala
4	int j_3 = (EEPROM.read(5))*60+(EEPROM.read(6)); //Mati
5	int j_4 = (EEPROM.read(7))*60+(EEPROM.read(8)); //Mati
6	//saklar 2
7	int j_5 = (EEPROM.read(9))*60+(EEPROM.read(10)); //Nyala
8	int j_6 = (EEPROM.read(11))*60+(EEPROM.read(12)); //Nyala
9	int j_7 = (EEPROM.read(13))*60+(EEPROM.read(14)); //Mati
10	int j_8 = (EEPROM.read(15))*60+(EEPROM.read(16)); //Mati

Nilai EEPROM berupa nilai jam yang disimpan pada alamat 1,3,5 dan 7 untuk saklar 1 serta 9,11,13 dan 15 untuk saklar 2. Nilai tersebut diolah dengan nilai dari alamat 2,4,6,8,10,12,14 dan 16 yang berupa nilai menit sehingga menghasilkan nilai total dalam satuan menit.

#### 2. Syntax pencarian nilai prior:

Baris	PROGRAM
1	float j_kelas = 56;
2	float l_nyala = 28;
3	float l_mati = 28;
4	//prior
5	float p_nyala = l_nyala/j_kelas;
6	float p_mati = l_mati/j_kelas;

Nilai prior didapatkan dari pembagian jumlah nilai l\_nyala dan j\_kelas untuk kondisi nyala. Sedangkan untuk kondisi mati pembagian nilai dilakukan dari jumlah l\_mati dengan j\_kelas.



### 3. Syntax pencarian nilai mean:

Baris	PROGRAM
1	//mean 1
2	float m_nyala1 = 7568+j_1+j_2/l_nyala;
3	float m_mati1 = 1065+j_3+j_4/l_mati;

Nilai mean nyala didapatkan dari jumlah seluruh data sampel dibagi dengan jumlah kondisi lampu nyala. Sedangkan nilai mean mati didapatkan dari jumlah seluruh data sampel dibagi dengan jumlah kondisi lampu mati.

### 4. Syntax pencarian nilai standar deviasi:

Baris	PROGRAM
1	//st_dev 1
2	Float sdev_nyala1 = sqrt((sq(1000-m_nyala1)+sq(245-
3	m_nyala1)+sq(1002-m_nyala1)+sq(255-m_nyala1)+sq(1025-
4	m_nyala1)+sq(270-m_nyala1)+sq(1021-m_nyala1)+sq(240-
5	m_nyala1)+sq(1015-m_nyala1)+sq(260-m_nyala1)+sq(1000-
6	m_nyala1)+sq(235-m_nyala1)+sq(j_1-m_nyala1)+sq(j_2-
7	m_nyala1))/l_nyala-1);
8	float sdev_mati1 = sqrt((sq(347-m_mati1)+sq(1335-
9	m_mati1)+sq(337-m_mati1)+sq(1325-m_mati1)+sq(363-
10	m_mati1)+sq(1315-m_mati1)+sq(350-m_mati1)+sq(1323-
11	m_mati1)+sq(375-m_mati1)+sq(1320-m_mati1)+sq(360-
12	m_mati1)+sq(1315-m_mati1)+sq(j_1-m_mati1)+sq(j_2-
13	m_mati1))/l_mati-1);

Terdapat dua nilai standar deviasi yang dihitung yaitu saat kondisi nyala dan saat kondisi mati.

### 5. Syntax pencarian distribusi Gaussian:

Baris	PROGRAM
1	float p_n1,p_m1;
2	float s1_gaus1,s1_gaus2;
3	float nyala1,mati1;
4	double q1,w1,e1,r1;
5	//Saat Nyala
6	if(waktu_s == 1000){
7	p_n1 = 1 / l_nyala;
8	p_m1 = 0 / l_mati;
9	nyala1 = (p_nyala * p_n1);
10	mati1 = (p_mati * p_m1);
11	}else if(waktu_s == 245){





```

12     p_n1 = 1 / l_nyala;
13     p_m1 = 0 / l_mati;
14     nyala1 = (p_nyala * p_n1);
15     mati1 = (p_mati * p_m1);
16 }else if(waktu_s == 1002){
17     p_n1 = 1 / l_nyala;
18     p_m1 = 0 / l_mati;
19     nyala1 = (p_nyala * p_n1);
20     mati1 = (p_mati * p_m1);
21 }else if(waktu_s == 255){
22     p_n1 = 1 / l_nyala;
23     p_m1 = 0 / l_mati;
25     nyala1 = (p_nyala * p_n1);
26     mati1 = (p_mati * p_m1);
27 .....
28 //Saat Mati
29     else if(waktu_s == 347){
30     p_n1 = 0 / l_nyala;
31     p_m1 = 1 / l_mati;
32     nyala1 = (p_nyala * p_n1);
33     mati1 = (p_mati * p_m1);
34 }else if(waktu_s == 1335){
35     p_n1 = 0 / l_nyala;
36     p_m1 = 1 / l_mati;
37     nyala1 = (p_nyala * p_n1);
38     mati1 = (p_mati * p_m1);
39 }else if(waktu_s == 337){
40     p_n1 = 0 / l_nyala;
41     p_m1 = 1 / l_mati;
42     nyala1 = (p_nyala * p_n1);
43     mati1 = (p_mati * p_m1);
44 }else if(waktu_s == 1325){
45     p_n1 = 0 / l_nyala;
46     p_m1 = 1 / l_mati;
47     nyala1 = (p_nyala * p_n1);
48     mati1 = (p_mati * p_m1);
49 .....
50 }else {

```



```

51 //P-Nyala
52 { q1 = sqrt (2 * 3.14 * (pow(sdev_nyala1, 2)));
53 w1 = -((pow((waktu_s - m_nyala1), 2)) / (2 *pow(sdev_nyala1,
54 2)));
55 e1 = pow(2.718282, w1) * 1000; // dikali 1000 pembulatan
56 r1 = 1 / q1 * 1000; // dikali 1000 pembulatan
57 s1_gaus1 = e1 * r1;
58 nyala1 = (p_nyala * s1_gaus1); // perhitungan peluang
posterior
59
60
61
62 }
63 //P-Mati
64 { q1 = sqrt (2 * 3.14 * (pow(sdev_matil, 2)));
65 w1 = -((pow((waktu_s - m_matil), 2)) / (2 *pow(sdev_matil,
66 2)));
67 e1 = pow(2.718282, w1) * 1000; // dikali 1000 pembulatan
68 r1 = 1 / q1 * 1000; // dikali 1000 pembulatan
69 s1_gaus2 = e1 * r1;
70 matil = (p_mati * s1_gaus2); // perhitungan peluang posterior
71 }

```

Pencarian distribusi Gaussian dimulai dengan perbandingan nilai sample dan kondisi waktu sekarang dalam menit. Setelah dibandingkan apabila kondisi tersebut tidak terpenuhi maka data dioalah dengan rumus Gaussian dengan variable yang telah dideklarasikan di dalam fungsi `n_bayes`.

Urutan Fungsi `n_bayes` dimulai dengan pembacaan nilai dari beberapa alamat tertentu di EEPROM dan mengubanhnya menjadi satuan menit. Setelah diubah menjadi satuan menit data diolah dengan metode *Naive Bayes* yang dimulai dengan pencarian nilai prior data, nilai mean dan standar deviasi dari saklar 1 dan 2. Setelah itu data tersebut dibandingkan dengan data latih selama enam hari yang sudah diambil secara manual. Apabila datanya tidak ada yang sama maka data akan dihitung dengan densitas Gaussian untuk mencari peluang posteriornya.

Setelah peluang posterior nya ditemukan maka akan dibandingkan nilai yang paling tinggi dan hasil waktu sekarang untuk menghasilkan output yang berupa kondisi lampu nyala atau mati.



## BAB 6 PENGUJIAN DAN ANALISIS

Terdapat dua pengujian utama pada bab ini yaitu pengujian hasil data EEPROM terhadap *trigger* saklar 1 dan 2 serta pengujian hasil data olahan dari program utama sistem.

### 6.1 Pengujian Data EEPROM

Sub bab ini berisikan pengujian data pada EEPROM di Arduino Uno yang di *trigger* oleh saklar 1 dan 2.

#### 6.1.1 Tujuan Pengujian

Tujuan dari pengujian ini adalah untuk melihat apakah data yang disimpan di EEPROM sama dengan hasil masukan dari saklar 1 dan 2. Input dari saklar 1 dan 2 akan di *trigger* dengan penggunaan saklar saat keadaan nyala dan mati secara bergantian. Output dari pengujian akan dilihat pada hasil di serial monitor Arduino IDE.

#### 6.1.2 Prosedur Pengujian

1. Menghubungkan jack USB Arduino UNO ke USB laptop.
2. Mengupload Program ke Arduino UNO.
3. Melihat hasil di serial monitor

Hasil Pengujian berupa waktu dalam jam dan menit yang akan dibandingkan dengan waktu data tersebut dimasukan. Pengujian dilakukan sebanyak 10 kali dengan saklar 1 dan 2 masing – masing sebanyak 5 kali idengan menggunakan program utama untuk menampilkan hasilnya. Contoh hasil pengujian bisa dilihat pada Gambar 6.1.

```

COM16 (Arduino/Genuino Uno)
=> Lampu Otomatis <==
Training Lampu Aktif
=====
Data1 REC (Ny) Waktu
7.35
Training Lampu Aktif
=====
Data3 REC (Ny) Waktu
7.35
Training Lampu Aktif
=====
Saklar 1 Stop
Training Lampu Aktif
=====
Saklar 2 Stop
Training Lampu Aktif
=====
Saklar 1 Stop
Training Lampu Aktif
=====
Saklar 2 Stop
Training Lampu Aktif
=====
Saklar 1 Stop
Training Lampu Aktif
=====
Saklar 2 Stop
Autoscroll Show timestamp
Nevline 9600 baud Clear output
Gemu EN 7:35 26/07/2019
  
```

Gambar 6.1 Hasil Data Rekam EEPROM.



### 6.1.3 Hasil dan Analisis Pengujian

**Tabel 6.1 Hasil Pengujian Saklar 1 dari Data EEPROM.**

NO	Hasil Masukan Saklar 1	Hasil Output	Sesuai
1	09.23	09.23	Ya
2	09.24	09.24	Ya
3	09.25	09.25	Ya
4	09.26	09.26	Ya
5	09.27	09.27	Ya

Berdasarkan hasil pengujian pada **Tabel 6.1** dengan menggunakan saklar 1 untuk memasukan data ke EEPROM dan melihat hasil outputnya melalui serial monitor maka didapatkan akurasi data yaitu:

$$\begin{aligned}
 A. Sistem &= \frac{D. Keseluruhan - D. Tidak Sesuai}{D. Keseluruhan} \times 100\% \\
 &= \frac{5 - 0}{5} \times 100\% \\
 &= \frac{5}{5} \times 100\% = 100\%
 \end{aligned}$$

**Tabel 6.2 Hasil Pengujian Saklar 2 dari Data EEPROM.**

NO	Hasil Masukan Saklar 2	Hasil Output	Sesuai
1	09.23	09.23	Ya
2	09.24	09.24	Ya
3	09.25	09.25	Ya
4	09.26	09.26	Ya
5	09.27	09.27	Ya

Berdasarkan hasil pengujian pada **Tabel 6.2** dengan menggunakan saklar 2 untuk memasukan data ke EEPROM dan melihat hasil outputnya melalui serial monitor maka didapatkan akurasi data yaitu:

$$\begin{aligned}
 A. Sistem &= \frac{D. Keseluruhan - D. Tidak Sesuai}{D. Keseluruhan} \times 100\% \\
 &= \frac{5 - 0}{5} \times 100\% \\
 &= \frac{5}{5} \times 100\% = 100\%
 \end{aligned}$$



## 6.2 Pengujian Sistem Utama

Sub bab ini berisikan pengujian data pada program utama yang akan menentukan kondisi nyala lampu dengan algoritme Naïve Bayes.

### 6.2.1 Tujuan Pengujian

Tujuan pengujian ini adalah untuk melihat seberapa akurat sistem dalam menerapkan Algoritme *Naïve Bayes* dalam program utama dan hasil akhir dari pengolahannya.

### 6.2.2 Prosedur Pengujian

1. Menghubungkan jack USB Arduino UNO ke USB laptop.
2. Penyesuaian hari dan waktu RTC untuk pengujian.
3. Upload program Utama ke Arduino UNO.
4. Menggunakan saklar training untuk menyalakan mode training.
5. Menggunakan saklar 1 dan 2 untuk memasukan data waktu nyala dan mati.
6. Menggunakan saklar training untuk mematikan mode training.
7. Kembali ke proses 2 sampai didapatkan data selama 6 hari.

Pengujian dilakukan 1 orang sebanyak 12 data nyala dan 12 data mati dan dilakukan selama 6 hari secara bertahap. Tingkat akurasi sistem ditentukan dengan hasil akhir sistem apakah sesuai dengan data masukan dari saklar 1 dan 2.

Susunan syntax program untuk mengatur waktu RTC DS 3231 adalah sebagai berikut:

Baris	PROGRAM
1	#include <DS3231.h>
2	
3	DS3231 rtc(SDA, SCL);
4	void setup()
5	{
6	Serial.begin(9600);
7	rtc.begin();
8	rtc.setDOW(THURSDAY);
9	rtc.setTime(4,14,0)
10	rtc.setDate(4,7,2019);
11	}
12	
13	void loop()
14	{

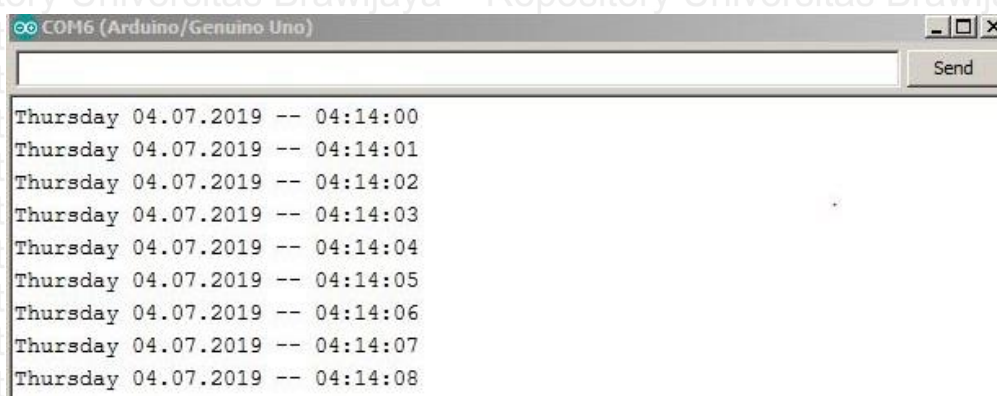


```

15 // Send Day-of-Week
16 Serial.print(rtc.getDOWStr());
17 Serial.print(" ");
18
19 // Send date
20 Serial.print(rtc.getDateStr());
21 Serial.print(" --");
22
23 // Send time
24 Serial.println(rtc.getTimeStr());
25
26 // Wait one second before repeating :)
27 delay (1000);
28 }

```

Pengaturan waktu diatur dengan penggalan hari, jam, waktu, menit dan tanggal secara berurutan sampai 6 hari. Contoh saat program dijalankan menghasilkan output seperti pada Gambar 6.2.



**Gambar 6.2 Hasil Pengaturan Waktu RTC.**

Setelah waktu diatur selanjutnya mengupload program utama dan menggunakan saklar training untuk menyalakan training mode. Pada saat training mode aktif, pengujian menggunakan saklar 1 dan 2 untuk mengatur nyala dan mati sesuai dengan kebiasaannya. Contoh hasil output program saat saklar training menyala bisa dilihat pada Gambar 6.3.



```

COM6 (Arduino/Genuino Uno)
=====> Lampu Otomatis <==
Training Lampu Aktif
=====
Data1 REC(Ny) Waktu
7.47
Training Lampu Aktif
=====
Data3 REC(Ny) Waktu
7.47
Training Lampu Aktif
=====
Saklar 1 Stop
Training Lampu Aktif
=====
Saklar 2 Stop
Training Lampu Aktif
=====
 Autoscroll  Show timestamp
Newline 9600 baud Clear output
  
```

**Gambar 6.3 Pengambilan Data Saat Saklar Training Menyala.**

Setelah data waktu nyala dan mati saklar direkam di EEPROM, program akan mengecek lagi kondisi saklar training. Jika kondisi saklar training mati maka program akan menjalankan fungsi Naïve Bayes yang akan mengolah data waktu di EEPROM untuk menentukan hasil output module relay yang akan membuat kondisi lampu menyala atau mati. Contoh hasil output saat saklar training mati bisa dilihat pada Gambar 6.4.

```

COM6 (Arduino/Genuino Uno)
Training Lampu Mati
<=====>
7.59 = Menit 479
<=====>
Lampu 1 Nyala
Lampu 2 Nyala
  
```

**Gambar 6.4 Hasil Output Saat Saklar Training Mati.**

### 6.2.3 Hasil dan Analisis Pengujian

Pengujian dimulai dari saklar 1 dan dilanjutkan dengan saklar 2.

**Tabel 6.3 Hasil Pengujian dari Saklar 1.**

No	Hari	Jam	Menit	Kelas	Hasil	Sesuai
1	1	16.40	1000	Nyala	Nyala	Ya
2		05.47	347	Mati	Mati	Ya
3	2	16.42	1002	Nyala	Nyala	Ya
4		05.37	337	Mati	Mati	Ya
5	3	17.05	1025	Nyala	Nyala	Ya
6		06.03	363	Mati	Mati	Ya
7	4	17.01	1021	Nyala	Nyala	Ya



8		05.50	350	Mati	Mati	Ya
9	5	16.55	1015	Nyala	Nyala	Ya
10		06.15	375	Mati	Nyala	Tidak
11	6	16.40	1000	Nyala	Nyala	Ya
12		05.37	337	Mati	Mati	Ya

Berdasarkan hasil pengujian pada **Tabel 6.3** dengan penggunaan saklar 1 terdapat 1 hasil yang tidak sesuai yaitu pada hari ke 5 saat lampu mati tetapi hasilnya nyala sehingga didapatkan akurasi sistem yaitu:

$$\begin{aligned}
 A. \text{ Sistem} &= \frac{D. \text{ Keseluruhan} - D. \text{ Tidak Sesuai}}{D. \text{ Keseluruhan}} \times 100 \\
 &= \frac{12 - 1}{12} \times 100\% \\
 &= \frac{11}{12} \times 100\% = 91.6\%
 \end{aligned}$$

**Tabel 6.4 Hasil Pengujian dari Saklar 2.**

No	Hari	Jam	Menit	Kelas	Hasil	Sesuai
1	1	16.40	1000	Nyala	Nyala	Ya
2		05.47	347	Mati	Mati	Ya
3	2	16.42	1002	Nyala	Nyala	Ya
4		05.37	337	Mati	Mati	Ya
5	3	17.05	1025	Nyala	Mati	Tidak
6		06.03	363	Mati	Mati	Ya
7	4	17.01	1021	Nyala	Nyala	Ya
8		05.50	350	Mati	Mati	Ya
9	5	16.55	1015	Nyala	Nyala	Ya
10		06.15	375	Mati	Nyala	Tidak
11	6	16.40	1000	Nyala	Nyala	Ya
12		05.37	337	Mati	Mati	Ya





Berdasarkan hasil pengujian pada **Tabel 6.4** dengan penggunaan saklar 2 terdapat 2 hasil yang tidak sesuai yaitu pada hari ke 3 saat lampu nyala tetapi hasilnya mati dan hari ke 5 saat lampu mati tetapi hasilnya nyala sehingga didapatkan akurasi sistem yaitu:

$$\begin{aligned}
 A. Sistem &= \frac{D. Keseluruhan - D. Tidak Sesuai}{D. Keseluruhan} \times 100\% \\
 &= \frac{12 - 2}{12} \times 100\% \\
 &= \frac{10}{12} \times 100\% = 83,3\%
 \end{aligned}$$



## BAB 7 PENUTUP

### 7.1 Kesimpulan

Kesimpulan yang didapat setelah keseluruhan analisis dan pengujian sistem yang telah dilakukan adalah sebagai berikut:

1. Implementasi sistem otomatisasi lampu rumah dengan menggunakan Algoritme *Naïve Bayes* didapatkan dengan mempelajari kebiasaan penghuni rumah secara berkala sehingga sistem dapat memprediksi kondisi lampu di kemudian hari menggunakan data yang sudah di simpan di EEPROM Arduino UNO.
2. Data sistem dapat disimpan dan dapat diketahui nilainya jika alamat data tersebut dipanggil dengan menggunakan fungsi EEPROM yang ada di Arduino. Alamat yang digunakan yaitu 1, 2, 3, 4, 5, 6, 7, dan 8 untuk saklar 1 sedangkan untuk saklar 2 yaitu 9, 10, 11, 12, 13, 14, 15, dan 16. Data maksimum yang dapat disimpan pada 1 slot alamat EEPROM sebesar 256 byte.
3. Jumlah akurasi data yang saat pengujian isi data dari EEPROM untuk saklar 1 dan 2 sebesar 100% dengan jumlah data uji 10, 5 untuk saklar 1 dan 5 untuk saklar 2.
4. Jumlah akurasi data saat pengujian sistem utama untuk saklar 1 sebesar 91.6% dan saklar 2 sebesar 83,3% dengan banyak total data uji sebanyak 12, 6 untuk nyala dan 6 untuk mati.

### 7.2 Saran

Saran yang diharapkan dapat membantu pengembangan lebih lanjut tentang sistem ini adalah sebagai berikut:

1. Memperbanyak data yang bisa diambil selama 1 hari sehingga keseluruhan sistem lebih fleksibel.
2. Menambahkan acuan Hari dalam 1 minggu sehingga akurasi sistem lebih baik.
3. Menggunakan penyimpanan yang lebih baik dari EEPROM agar data yang disimpan bisa bervariasi dan lebih banyak.
4. Menambahkan Algoritme lain yang digabungkan agar akurasi sistem bisa lebih baik.



## DAFTAR PUSTAKA

- Arduino, 2017. *Arduino Uno Rev3*. [online] Tersedia di: <https://store.arduino.cc/usa/arduino-uno-rev3> [Diakses 28 Agustus 2018]
- Ardial. M., 2012. Pengertian Kebiasaan Belajar. [online] Tersedia di: <https://www.psychologymania.com/2012/06/pengertian-kebiasaan-belajar.html> [Diakses 28 Agustus 2018]
- Bregman, D., 2010. *Smart Home Intelligence - The eHome that Learns*. [pdf]. Tersedia di: <https://pdfs.semanticscholar.org/9b96/20d9c1a69c1e3ae27ec0abc70a4d16a2823f.pdf> [Diakses 28 Agustus 2018]
- Belajarduino, 2016. *Belajar Duino*. [online] Tersedia di: <http://www.belajarduino.com/2016/10/programmable-timer-8channel-dengan.html> [Diakses 28 Agustus 2018]
- Ecadio, 2016. *Ecadio*. [online] Tersedia di: <http://ecadio.com/mengenal-dan-belajar-arduino-uno-r3> [Diakses 28 Agustus 2018]
- Elektronika, 2016. *Edukasi Elektronika*. [online] Tersedia di: <https://www.edukasielektronika.com/2016/05/arduino-uno.html> [Diakses 28 Agustus 2018]
- GudangLinux, 2013. *GudangLinux*. [online] Tersedia di: <http://gudanglinux.com/glossary/i%C2%B2c-inter-integrated-circuit/> [Diakses 05 April 2018]
- Informatikalogi, 2018. *Informatikalogi*. [online] Tersedia di: <https://informatikalogi.com/Algoritme-naive-bayes/> [Diakses 28 Agustus 2018]
- Masykur .F., 2016. Aplikasi Rumah Pintar (Smart Home) Pengendali Peralatan Elektronik Rumah Tangga Berbasis Web. [online] Tersedia di: <http://ejournal.uin-suska.ac.id/index.php/sitekin/article/view/2185> [Diakses 28 Agustus 2018]
- Faudin .A., 2017. Tutorial Arduino Mengakses Modul RTC DS3231. [online] Tersedia di: <https://www.nyebarilmu.com/tutorial-arduino-mengakses-module-rtc-ds3231/> [Diakses 24 Maret 2019]
- Lin, H T., 2013. Implementing Smart Homes with Open Source Solutions. [pdf]. Tersedia di: [http://www.sersc.org/journals/IJSH/vol7\\_no4\\_2013/29.pdf](http://www.sersc.org/journals/IJSH/vol7_no4_2013/29.pdf) [Diakses 28 Agustus 2018]
- Proyekarduino, 2015. *Proyek Arduino*. [online] Tersedia di: <https://proyekarduino.wordpress.com/2015/04/01/pengetahuan-dasar-rtc-ds3231/> [Diakses 28 Agustus 2018]
- Rinjani, P.R., 2017. Pengembangan Centralized Smarthome Simulator Sebagai Solusi Penghematan Biaya Listrik Menggunakan Algoritme *Naive Bayes*. S1.



Universitas Brawijaya. Tersedia di <http://jurnal-ptiik.ub.ac.id> [Diakses 28 Agustus 2018]

Turena, G., dkk., 2015. Pengendali Intensitas Lampu Ruang Berbasis Arduino UNO Menggunakan Metode Fuzzy Logic. S1. Universitas Langlang Buana. Tersedia di <http://jurnalunla.web.id/tiarsie/index.php/tiarsie> [Diakses 23 Maret 2019]

Pratama, S.M., 2018. Implementasi Algoritme *Naive Bayes* Menggunakan Arduino Uno untuk Otomatisasi Lampu Ruang Berdasarkan Kebiasaan dari Penghuni Rumah. S1. Universitas Brawijaya. Tersedia di <http://jurnal-ptiik.ub.ac.id> [Diakses 28 Agustus 2018]

Yudi Agusta., 2007. *Naive Bayes – Penerapan, Permasalahan dan Metode Terkait. Jurnal Sistem dan Informatika, 47-60*



## LAMPIRAN A DATA HASIL PERCOBAAN

No	Hari	Jam	Menit	Kelas	Hasil	Sesuai
1	1	16.40	1000	Nyala	Nyala	Ya
2		05.47	347	Mati	Mati	Ya
3	2	16.42	1002	Nyala	Nyala	Ya
4		05.37	337	Mati	Mati	Ya
5	3	17.05	1025	Nyala	Nyala	Ya
6		06.03	363	Mati	Mati	Ya
7	4	17.01	1021	Nyala	Nyala	Ya
8		05.50	350	Mati	Mati	Ya
9	5	16.55	1015	Nyala	Nyala	Ya
10		06.15	375	Mati	Nyala	Tidak
11	6	16.40	1000	Nyala	Nyala	Ya
12		05.37	337	Mati	Mati	Ya

No	Hari	Jam	Menit	Kelas	Hasil	Sesuai
1	1	16.40	1000	Nyala	Nyala	Ya
2		05.47	347	Mati	Mati	Ya
3	2	16.42	1002	Nyala	Nyala	Ya
4		05.37	337	Mati	Mati	Ya
5	3	17.05	1025	Nyala	Mati	Tidak
6		06.03	363	Mati	Mati	Ya
7	4	17.01	1021	Nyala	Nyala	Ya
8		05.50	350	Mati	Mati	Ya
9	5	16.55	1015	Nyala	Nyala	Ya
10		06.15	375	Mati	Nyala	Tidak
11	6	16.40	1000	Nyala	Nyala	Ya
12		05.37	337	Mati	Mati	Ya



NO	Hasil Masukan Saklar 1	Hasil Output	Sesuai
1	09.23	09.23	Ya
2	09.24	09.24	Ya
3	09.25	09.25	Ya
4	09.26	09.26	Ya
5	09.27	09.27	Ya

NO	Hasil Masukan Saklar 1	Hasil Output	Sesuai
1	09.23	09.23	Ya
2	09.24	09.24	Ya
3	09.25	09.25	Ya
4	09.26	09.26	Ya
5	09.27	09.27	Ya