

PERANCANGAN PROTOTYPE KENDALI LAMPU BERBASIS INTERNET OF THINGS (IOT) DENGAN NODEMCU ESP8266 DAN VOICE RECOGNITION PADA SMARTPHONE

Yus Jayusman¹, Rahmat Zaenal², Irfan Faisal³

^{1 2 3} STMIK BANDUNG

Sekolah Tinggi Manajemen Informatika dan Komputer Bandung

JL. Cikutra No.113 Bandung – 40192, Telp/Fax. 022-7207777

Contact address :

irvanfaisal.viel@gmail.com¹, yusjayusman@gmail.com²

ABSTRAK

Pembuatan prototype kontrol lampu ini bertujuan untuk memudahkan dalam menghidupkan maupun mematikan lampu karena pada dasarnya manusia menyukai sesuatu yang praktis dan mudah, dengan terlealisasinya prototype ini diharapkan dapat memudahkan manusia dalam menghidupkan atau mematikan lampu tanpa harus berjalan mendekati saklar lampu.

Aplikasi dari sistem kontrol lampu berbasis (IoT) Internet of Things ini menggunakan modul NodeMCU ESP8266 sebagai mikrocontroller dan aplikasi android sebagai alat pengendali. Alat ini memanfaatkan voice recognition dari google untuk menerima perintah suara dan diterjemahkan menjadi sebuah perintah untuk menyalakan dan mematikan lampu secara wireless.

Metodologi pembangunan perangkat menggunakan metode Prototype dengan perancangan sistem menggunakan UML. Pembuatan prototype menggunakan NodeMCU ESP8266 sebagai kontroler utama.

Kata kunci: NodeMCU, ESP8266, Kontrol Lampu Digital, Voice Recognition, IoT, Android

ABSTRACT

Manufacture of prototype of light control was used for simplicity in turning on or turning off the lights, because basically people like something more practical and easy. In the making of this prototype, hope that can make easy of turning on and turning off the lights without switching directly from the switch.

This Light Control system is based on Internet of Things (IoT) and using NodeMCU ESP8266 as the microcontroller and an Android App as a control gate of the system. This tool utilize voice recognition of Google to receive voice commands and translated to a form of command line for turning on or turning off the lights wirelessly.

Software development methodology uses the prototype method with system design using UML. NodeMCU ESP8266 is used as the core control system.

Keywords: NodeMCU, ESP8266, Digital Light Control, Voice Recognition, Prototype, UML.

1. Pendahuluan

Seiring berkembangnya teknologi dan mobilitas penduduk Indonesia yang tinggi, perangkat-perangkat mikrokomputer telah banyak digunakan dalam berbagai aspek kehidupan sehari-hari. Era Internet Of Things (IOT) saat ini dapat dimanfaatkan untuk membentuk sebuah

sistem smart home sebagai media kontrol dan monitoring terhadap berbagai macam perangkat elektrik di rumah sehingga penggunaannya dapat lebih efektif dan efisien sesuai dengan kebutuhan [1]. Dengan sistem ini, user dapat mengontrol peralatan elektrik hanya dengan mengakses websitedan mengendalikannya dengan komputer atau smartphoneyang terhubung ke jaringan internet

Perangkat mikrokomputer tersebut dapat berupa *smartphone* ataupun mikrokontroler. Penggunaan *smartphone* juga semakin meningkat dikarenakan dengan mudahnya akses pertukaran data dan dapat diakses dimana saja. Selain itu, dengan hadirnya sebuah teknologi bernama *Voice Recognition* oleh *Google* yang mampu menterjemahkan bahasa manusia menjadi sebuah teks dapat membuka peluang bagi para pengembang aplikasi untuk mengeksplorasi kemampuan tersebut untuk dimanfaatkan dalam sebuah sistem. Oleh karena itu hadirilah sebuah *Prototype Kontrol Lampu dengan Voice Recognition* yang nantinya mempermudah pengguna dalam mengontrol lampu ruangan. Perangkat harus dapat diakses dengan mudah, mudah diperluas sehingga dapat dengan mudah menambahkan perangkat baru, dan harus dapat dengan mudah dikendalikan. Tujuan dari implementasi *Prototype Kontrol Lampu dengan Voice Recognition* ini nantinya yaitu dapat mengendalikan sebuah sistem berupa lampu melalui bantuan *smartphone* dan *Google Voice Recognition* sehingga pengguna dapat melihat dan mengaturnya dengan mudah sehingga sesuai dengan kondisi yang diinginkan. Pengguna yang ditujukan untuk pengaplikasian *prototype* ini adalah sebuah keluarga, seorang individu yang sibuk, orang tua dan penyandang disabilitas.

Tidak bisa dipungkiri juga bahwa manusia pada umumnya selalu menginginkan sesuatu yang serba praktis, pada kasus ini contohnya adalah kontrol lampu. Orang-orang yang sedang mengalami keterbatasan untuk menyalakan atau menghidupkan lampu seperti orang yang sedang sakit atau penyandang disabilitas atau bahkan orang sibuk yang sering lupa waktu akan sangat terbantu karena tidak harus berjalan menghampiri *switch* untuk mengontrol lampu. Hal ini juga secara tidak langsung akan membantu dalam menghemat energi listrik.

1.1 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka dapat dirumuskan masalah yaitu:

1. Pengendalian lampu belum bisa dilakukan secara otomatis.
2. Pengendalian lampu masih dilakukan secara analog.
3. Orang penyandang disabilitas atau yang sedang sakit yang sering ditinggal wali nya.

1.2 Tujuan Penelitian

Berdasarkan permasalahan yang diteliti, maka tujuan dari skripsi ini adalah:

1. Mewujudkan sistem kendali secara *wireless* melalui *smartphone* untuk memudahkan pengguna.
2. Memungkinkan pengguna untuk mengontrol lampu secara digital dengan *voice recognition* melalui *smartphone*.
3. Membantu orang penyandang disabilitas atau orang yang sedang sakit untuk mengontrol lampu.

1.3 Internet of Things (IoT)

Internet of Things adalah konsep yang muncul dimana semua alat dan layanan terhubung satu dengan yang lain dengan mengumpulkan, bertukar dan memproses data untuk beradaptasi secara dinamis. Di dalam bahasan "*Smart Home Environments*" antara IoT dan alat ataupun layanan tradisional berintegrasi di dalam rumah untuk meningkatkan kualitas hidup. Ini memungkinkan peningkatan di berbagai bidang seperti penghematan energi, pengamatan kesehatan, dan yang lainnya [1].

1.4 Rangkaian Listrik

Rangkaian listrik (atau rangkaian elektrik) merupakan interkoneksi berbagai piranti (*device – device*) yang secara bersama melaksanakan suatu tugas tertentu. Tugas itu dapat berupa pemrosesan energi ataupun pemrosesan informasi. Melalui rangkaian listrik, energi maupun informasi dikonversikan menjadi energi listrik dan sinyal listrik, dan dalam bentuk sinyal inilah energi maupun informasi dapat disalurkan dengan lebih mudah ke tempat ia diperlukan [2].

1.5 Voice Recognition

Pengenalan ucapan atau pengenalan suara dalam istilah bahasa Inggrisnya, *voice recognition* adalah suatu pengembangan teknik dan sistem yang memungkinkan komputer untuk menerima masukan berupa kata yang diucapkan. Teknologi ini memungkinkan suatu perangkat untuk mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat. Kata-kata yang diucapkan diubah bentuknya menjadi sinyal digital dengan cara mengubah gelombang suara menjadi sekumpulan angka yang kemudian disesuaikan dengan kodekode tertentu untuk mengidentifikasikan kata-kata tersebut. Hasil dari identifikasi kata yang diucapkan dapat ditampilkan dalam bentuk tulisan atau dapat dibaca oleh perangkat teknologi sebagai sebuah komando untuk melakukan suatu pekerjaan, misalnya penekanan tombol pada telepon genggam yang dilakukan secara otomatis

dengan komando suara. Alat pengenalan ucapan/suara, yang sering disebut dengan voice recognizer, membutuhkan sampel kata sebenarnya yang diucapkan dari pengguna. Sampel kata akan didigitalisasi, disimpan dalam komputer, dan kemudian digunakan sebagai basis data dalam mencocokkan kata yang diucapkan selanjutnya. Sebagian besar alat pengenalan ucapan sifatnya masih tergantung kepada pembicara. Alat ini hanya dapat mengenal kata yang diucapkan dari satu atau dua orang saja dan hanya bisa mengenal kata-kata terpisah, yaitu kata-kata yang dalam penyampaian terdapat jeda antar kata [3].

1.6 NodeMCU ESP8266

NodeMCU adalah *chip* yang bersifat *opensource* yang terdiri dari ESP8266 WiFi SoC yang dirancang oleh Espressif System yang berbasis CPU RISC 32-bit dengan prosesor Tensilica Xtensa LX106. Fitur yang dimiliki seperti Wi-Fi bawaan (802.11 b/g/n), GPIO (*General Purpose Input/Output*), I2C (*Inter-Integrated Circuit*), analogto-digital conversion, *Serial Peripheral Interface* (SPI), UART (*Universal asynchronous receiver/transmitter*), and *pulse-width modulation* (PWM). ESP8266 memiliki sejumlah versi dengan kapasitas yang bervariasi [4].

1.7 Relay

Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen Elektromekanikal yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat Kontak Saklar/*Switch*). Relay menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan Relay yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan *Armature Relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A [4].

1.8 App Inventor

App Inventor adalah *Integrated Development Environment* (IDE) yang ditujukan bagi semua kalangan untuk mengembangkan aplikasi Android tanpa harus berpengalaman dalam dunia pemrograman. Aplikasi ini merupakan aplikasi *open source* (sumber terbuka) berbasis *web* yang semulanya disediakan oleh Google dan sekarang dikelola oleh *Massachusetts Institute of*

Technology (MIT). App Inventor didesain dengan simple dan mudah dimengerti. App Inventor juga memiliki *tool* berbasis *drag/drop* dan *visual blocks programming* untuk mengembangkan aplikasi yang berjalan di *Android Operating System*. Sehingga kita tidak perlu menuliskan kodekode pemrograman untuk membuat aplikasi. *Future Visual Blocks Programming* mampu mentransformasikan pengkodean bahasa pemrograman berbasis teks kedalam bahasa *visual* dalam bentuk kode-kode program [4].

2. Pembahasan

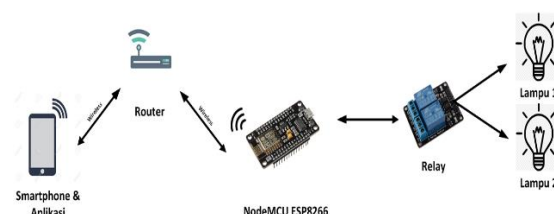
Prototype kendali lampu ini dirancang dengan memanfaatkan *Internet of Things* untuk memungkinkan mengendalikan lampu secara wireless dengan smartphone sebagai alat kendali dan NodeMCU ESP8266 sebagai mikrokontrolernya.

Secara singkat, cara kerja prototype ini adalah dengan smartphone mengirimkan perintah melalui internet yang akan diterima oleh NodeMCU yang juga terkoneksi melalui internet, kemudian NodeMCU akan mengolah perintah tersebut untuk dijadikan sebuah aksi berupa menyalakan lampu dengan mengirim value ke .

Game edukasi memilah sampah ini merupakan *game* yang berjenis *action-adventure* yang berisi konten pembelajaran tentang memilah sampah, yang dimana tujuannya dengan memainkan *game* ini diharapkan pemain dapat mengerti dan memahami tentang sampah organik dan anorganik dan dapat diterapkan dalam memilah sampah. *Game* yang dibangun *single player* atau dengan satu pemain yang menjalankan seorang karakter utama berupa seorang yang melawan monster sampah yang bernama uzwa.

Game dibangun dengan menerapkan konsep *action-adventure* agar mengenalkan nuansa petualangan yang diharapkan dapat melatih keterampilan, melatih motorik halus, melatih cara pemecahan masalah.

2.1 Arsitektur Sistem



Gambar 2.1 Arsitektur Sistem

Smartphone berfungsi sebagai pemegang kendali terhadap NodeMCU melalui sebuah aplikasi

dengan komunikasi yang dilakukan secara *wireless* melalui *router*. Kendali yang dimaksudkan adalah berupa perintah untuk mematikan dan menyalakan lampu dengan memberi perintah ke NodeMCU yang kemudian diterjemahkan dan diteruskan ke *Relay* untuk melakukan aksi menyalakan atau mematikan lampu yang terhubung ke *relay*.

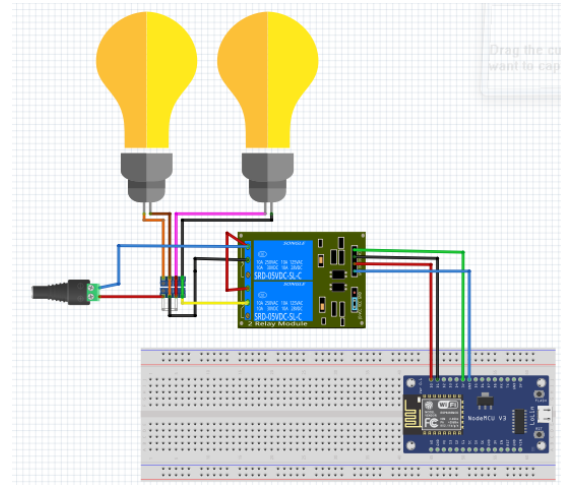
2.2 Arsitektur Software



Gambar 2.2 Arsitektur Software

Dari gambar diatas dapat di deskripsikan bahwa, saat pertama kali pengguna membuka aplikasi maka pengguna harus menetapkan IP Lokal dari NodeMCU. IP Lokal ini bisa diketahui dengan melakukan program pada NodeMCU. Jika IP *valid* maka aplikasi akan mengirimkan perintah ke NodeMCU untuk melakukan aksi, dan jika IP tidak sesuai maka saat pengguna menekan tombol, NodeMCU tidak akan merespon dan aplikasi akan terindikasi *error* koneksi karena aplikasi pada pengguna tidak terhubung ke NodeMCU meskipun terkoneksi pada *router* yang sama

2.3 Arsitektur Hardware



Gambar 2.3 Arsitektur Hardware

Dari gambar diatas dapat dideskripsikan bahwa NodeMCU adalah sebagai dapur utama dimana NodeMCU ini akan menerjemahkan perintah yang dikirim dari smartphone menjadi sebuah perintah untuk menyalakan atau mematikan lampu secara *wireless*. Rangkaian utama dari susunan perangkat keras ini adalah terdapat *Relay* yang terhubung ke Pin pada NodeMCU dan ke *Power Supply* sebagai tenaga utama untuk menghidupkan Lampu. *Relay* disini berperan sebagai *switch digital*, cara kerjanya sama dengan *switch* yang digunakan di rumah pada umumnya yaitu memutus arus listrik, hanya saja bisa dikontrol secara digital. *Power Supply* digunakan untuk sumber listrik untuk menyalakan Lampu, jadi lampu tidak menerima daya secara langsung dari *Relay* atau NodeMCU, tetapi langsung dari sumbernya. *Relay* dan NodeMCU hanya berperan sebagai *switch digital* saja.

2.4 Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh aplikasi yang berisi informasi-informasi apa saja yang harus ada dan dihasilkan oleh aplikasi. Adapun analisa kebutuhan fungsional meliputi:

- Sistem harus bisa memberikan akses kepada smartphone pengguna untuk dapat melakukan kontrol lampu.
- Setiap modul dan perangkat yang digunakan berjalan dengan semestinya.
- Sistem bisa terkoneksi dengan smartphone pengguna dengan baik.

2.5 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak terbagi menjadi dua bagian, yaitu analisis kebutuhan

perangkat lunak untuk pengembang dan analisis kebutuhan perangkat lunak untuk pengguna.

1. Analisis Kebutuhan Perangkat Lunak Untuk Pengembang

Perangkat lunak yang digunakan dalam pengembangan *prototype* smart home ini adalah sebagai berikut :

- a. Sistem Operasi Windows 7 64 bit.
 - b. Pemodelan dan perancangan sistem menggunakan pemodelan berbasis objek dengan *tools* UML (*Unified Modeling Language*).
 - c. Aplikasi dikembangkan dengan *tool* berbasis *web* yaitu MIT App Inventor.
 - d. Mikrokontroler diprogram dengan menggunakan *Arduino IDE*.
- ## 2. Analisis Kebutuhan Perangkat Lunak Untuk Pengguna

Perangkat lunak yang dibutuhkan pengguna dalam menjalankan *prototype* ini adalah Aplikasi Kontrol Lampu dan *Arduino IDE*.

2.6 Analisis Kebutuhan Perangkat Keras

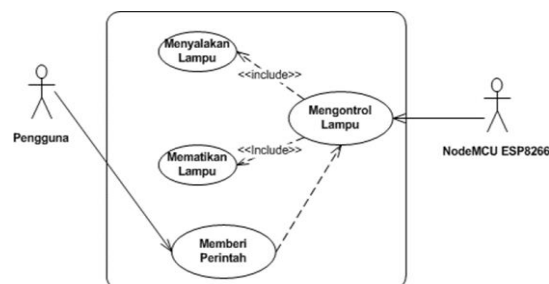
1. Pada *prototype* ini, analisis perangkat perangkat keras yang digunakan pengembang adalah sebagai berikut :
 - a) Laptop
 - b) NodeMCU ESP8266
 - c) Relay 2 Channel
 - d) Lampu 2.5watt
 - e) Breadboard
 - f) Adaptor 7-12V
 - g) Shield NodeMCU
 - h) Kabel Jumper
 - i) Stock Kontak

1. Analisis Kebutuhan Perangkat Keras Untuk Pengguna

Perangkat keras yang dibutuhkan pengguna dalam menjalankan *prototype* ini adalah sebuah *smartphone* yang dapat terkoneksi dengan internet dan *wifi*.

2.7 Use Case Diagram

Use case diagram merupakan pemodelan untuk mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Berikut adalah model *use case diagram* :



Gambar 2.4 Use Case Diagram

2.8 Use Case Skenario

- a. Use Case Skenario : Memberi Perintah

Tabel 2.1 Use Case Scenario Memberi Perintah

Scenario Utama	
Pre Condition : Valid IP	
Aktor	Sistem
1. Memasukan IP Lokal NodeMCU yang digunakan.	
	2. Set IP Lokal untuk disusun menjadi sebuah perintah tergantung input yang dilakukan pengguna.
3. Menekan tombol / memasukan perintah suara.	
	4. Menggabungkan IP Lokal NodeMCU dengan perintah yang diatur pada tombol. Format : IP_Lokal/command Contoh : 192.168.43.101/r1on

- b. Use Case Skenario : Memberi Perintah

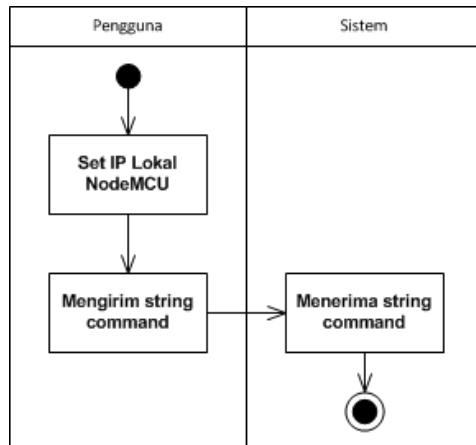
Tabel 2.2 Use Case Scenario mengontrol lampu

Scenario Utama	
Pre Condition : Valid IP	
Aktor	Sistem
1. Menerima perintah dari pengguna.	
	2. Mengidentifikasi perintah yang diterima kemudian di terjemahkan dan diteruskan ke relay. Contoh : command:r1on = relay 1 high.

2.9 Activity Diagram

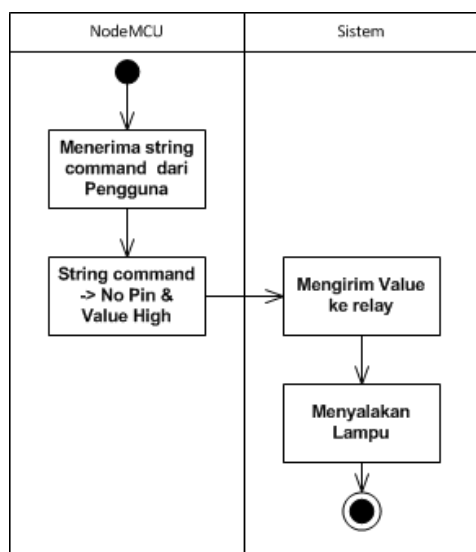
Activity diagram merupakan diagram yang memodelkan aliran kerja atau workflow dari urutan aktifitas dalam suatu proses yang mengacu pada use case diagram yang ada.

a. Activity Diagram Memberi Perintah



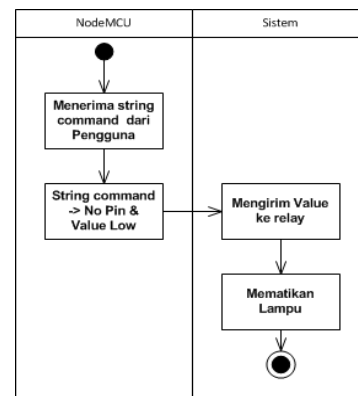
Gambar 2.5 Activity Diagram Memberi Perintah

b. Activity Diagram Menyalakan Lampu



Gambar 2.6 Activity Diagram Menyalakan Lampu

c. Activity Diagram Mematikan Lampu

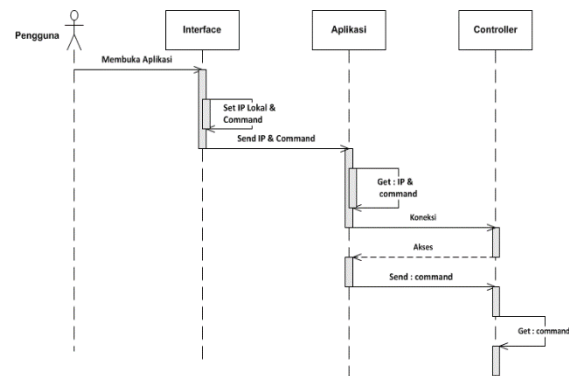


Gambar 2.7 Activity Diagram Mematikan Lampu

2.10 Sequence Diagram

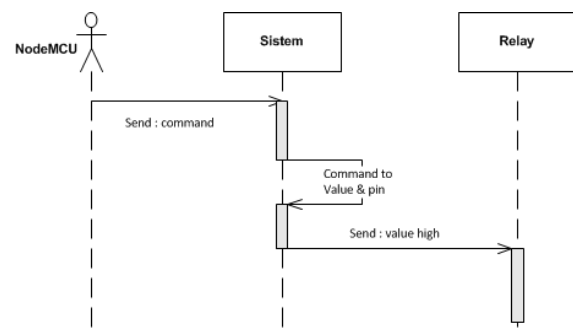
Sequence diagram menggambarkan interaksi antar objek di dalam dan disekitar sistem berupa *message* yang digambarkan terhadap waktu. Berikut ini merupakan *sequence diagram* yang terdapat pada *prototype* ini.

a. Sequence Diagram Memberi Perintah



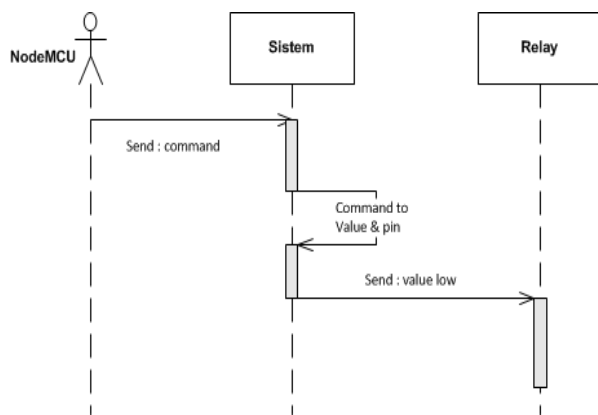
Gambar 2.8 Sequence Diagram Memberi Perintah

b. Sequence Diagram Menyalakan Lampu



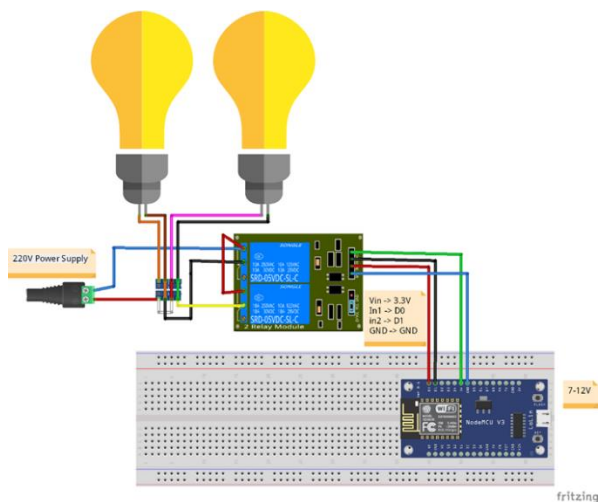
Gambar 2.9 Sequence Diagram Menyalakan Lampu

c. Sequence Diagram Mematikan Lampu



Gambar 2.10 Sequence Diagram Mematikan Lampu

2.11 Rancangan Skematik



Gambar 2.11 Rancangan Skematik

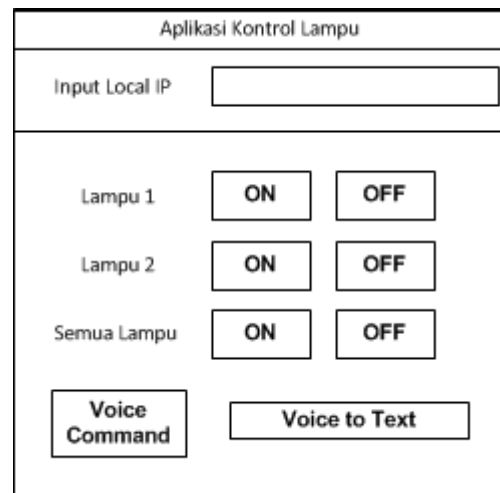
Susunan Pin yang digunakan pada NodeMCU yang dihubungkan ke Relay

- Vin -> 3.3V
- In1 -> D0
- In2 -> D1
- GND -> GND

Untuk daya yang mengalir NodeMCU menggunakan Adaptor 7-12v dan daya utama yang digunakan adalah 220v untuk mensupply daya ke Lampu. Relay dapat dinyalakan atau dimatikan dengan mengatur ada atau tidaknya arus yang mengalir ke relay untuk penggerakkan tuas saklar dengan lilitan kawat pada batang besi (solenoid) di dekatnya. Ketika solenoid dialiri arus listrik tuas akan tertarik karena adanya gaya magnet yang terjadi pada solenoid sehingga kontak saklar akan menutup. Pada saat arus dihentikan, gaya magnet akan hilang, tuas akan kembali ke posisi semula dan kontak saklar kembali terbuka. Relay di sini

bekerja untuk mengontrol hidup dan matinya lampu.

2.12 Rancangan Antarmuka Aplikasi



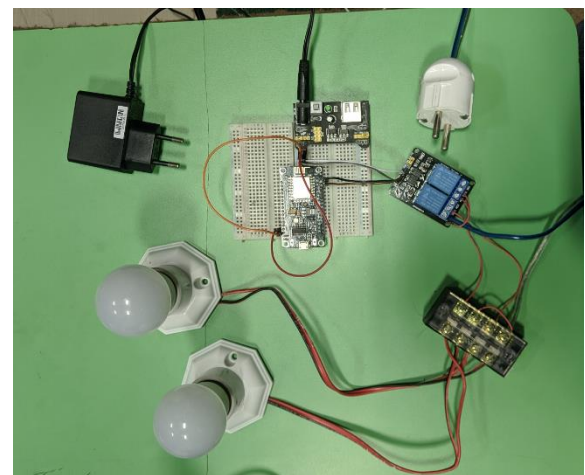
Gambar 2.12 Rancangan Antarmuka

Terdapat *field input* untuk *Input Local IP* dimana Lokal IP disini adalah Lokal IP NodeMCU. Terdapat 3 buah tombol ON dan 3 tombol OFF masing-masing untuk menghidupkan dan mematikan lampu 1, lampu 2 dan kedua lampu sekaligus. Terdapat pula tombol *Voice Command* untuk menginputkan perintah suara yang kemudian hasilnya akan ditampilkan dalam bentuk teks pada *field Voice to Text* tepat disampingnya.

3. Implementasi Sistem dan Pengujian

3.1 Mempersiapkan Perangkat Keras

Persiapan perangkat keras berikut merupakan bentuk implementasi dari rangkaian skematik yang telah dibangun. Berikut ini adalah rangkaian perangkat keras yang telah dibangun :



Gambar 3.1 Rangkaian Perangkat Keras

3.2 Mempersiapkan Perangkat Lunak

Perangkat lunak yang dipersiapkan dan digunakan haruslah sesuai dengan perangkat keras yang telah dipersiapkan. Adapun perangkat lunak (*software*) yang digunakan selama membangun *Prototype* ini adalah sebagai berikut:

1. *Arduino IDE*.

Seperti yang telah dibahas sebelumnya, terdapat satu syarat utama yang menjadi hal penting dalam proses kontrol lampu ini, yaitu IP Lokal NodeMCU. Berikut adalah cara untuk mengetahui IP Lokal NodeMCU. Sebelum itu, pada dasarnya *Arduino IDE* tidak mensupport NodeMCU, oleh karena itu *Arduino IDE* harus dapat mensupport NodeMCU terlebih dahulu agar NodeMCU dapat diprogram. Berikut adalah langkah langkahnya :

1. Instal *Arduino IDE* versi terbaru.
2. Buka *Arduino IDE*, kemudian klik *File > Preferences > Additional Boards Manager URLs*.
3. Salin pada *field* tersebut *link* berikut ini: https://github.com/esp8266/Arduino/releases/download/2.5.0/package_esp8266_com.tar.gz.

Kemudian klik OK.

4. Kemudian, klik *Tools > Board [...] > Board Managers*.
5. Pada *field filter*, ketik *esp8266* maka selanjutnya akan muncul nama librarynya *esp8266 by ESP8266 Community* kemudian instal.
6. Setelah selesai mengunduh, *close*, dan buka ulang *Arduino IDE*.
7. Pilih boardnya menjadi *ESP8266*, klik *Tools > (Geser kebawah) Pilih NodeMCU 1.0 (ESP-12E Module)*.
8. Pada tahap ini, NodeMCU sudah bisa di program dengan *Arduino IDE*.
9. Selebihnya bisa dilakukan testing dengan menyalakan *LED_BUILTIN* NodeMCU untuk memastikan.

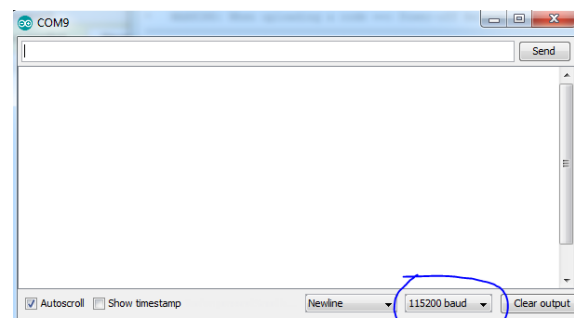
Setelah melewati langkah langkah diatas, ikuti langkah berikut ini untuk menyiapkan program untuk NodeMCU.

1. Buka Program Kontrol Lampu (ekstensi .ino).
2. Setelah *Arduino IDE* muncul, pada bagian *coding* terdapat '*ssid*' dan '*password*'.

```
#include <ESP8266WiFi.h>
WiFiClient client;
WiFiServer server(80);
const char* ssid = "yourssid";
const char* password = "yourssidpass";
String command = ""; // Command received from Android device
```

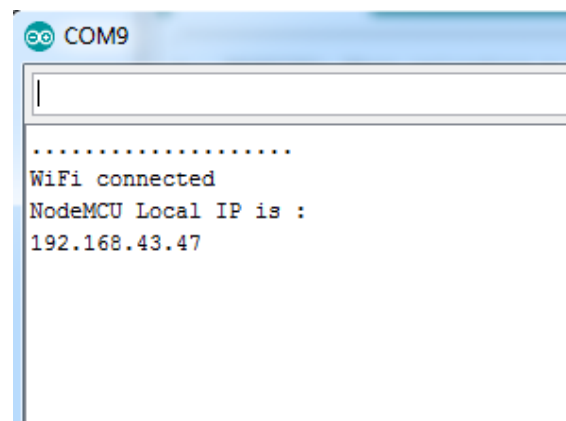
Gambar 3.2 Setting SSID Password

1. Isi '*ssid*' dan *password* ubah menjadi *ssid* dan *password* *wifi* yang akan digunakan. Yang dimaksudkan disini adalah *wifi* mana yang akan digunakan, NodeMCU akan mengkoneksikan dirinya dengan *wifi* tersebut.
2. Setelah itu, koneksikan NodeMCU dengan laptop menggunakan kabel USB yang cocok dengan slot NodeMCU.
3. Setelah konek, buka *Arduino IDE* kemudian klik *Upload* atau *ctrl+u* dan tunggu sampai *progress bar* menunjukkan status '*Done Uploading*'.
4. Jika sudah di *upload*, klik *serial monitor* di pojok kanan atas dan ganti *baud rate* menjadi 115200.



Gambar 3.3 Baud Rate

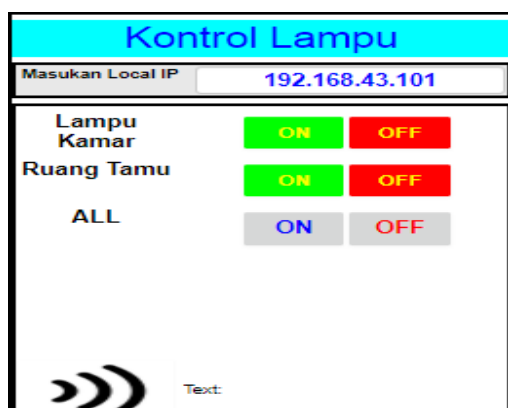
5. Jika *setting* *wifi* benar maka akan muncul tampilan ini pada *serial monitor*.



Gambar 3.4 Lokal IP NodeMCU

6. 192.168.43.47 adalah contoh lokal IP NodeMCU yang digunakan.
7. Pada tahap ini, Lokal IP NodeMC sudah didapatkan dan dapat digunakan pada aplikasi android Kontrol Lampu.

2. Aplikasi Kontrol Lampu



Gambar 3.5 Aplikasi Kontrol Lampu

Berikut beberapa syarat untuk dapat menggunakan aplikasi Kontrol Lampu ini :

- Terkoneksi dengan *ssid* yang sama dengan *ssid* yang dipakai NodeMCU.
- Memasukan Lokal IP NodeMCU pada *field* Lokal IP.

Jika kedua persyaratan diatas terpenuhi maka dengan menunggu beberapa detik, aplikasi ini akan terhubung dengan NodeMCU dan dapat digunakan untuk mengontrol lampu pada rangkaian *Prototype* dan siap untuk dilakukan pengujian.

3.3 Pengujian

Pengujian merupakan bagian penting dalam siklus pengembangan. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari sistem. Pengujian dilakukan untuk mengetahui apakah fungsi-fungsi yang telah direncanakan bekerja dengan baik atau tidak. Pengujian juga berguna untuk mengetahui tingkat kinerja dari fungsi [2]. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat sistem yang dibangun memiliki kualitas dan dapat bekerja dengan baik. Pengujian ini menggunakan metode pengujian *Black Box*. Pengujian *Black Box* digunakan untuk menguji fungsi-fungsi khusus dari perangkat yang dirancang.

3.4 Rencana Pengujian

Adapun rancangan pengujian sistem yang diuji dengan teknik pengujian *Prototype* dikelompokkan dalam tabel 3.1.

3.2 Kasus dan Hasil Pengujian

Setelah merencanakan pengujian, maka dilakukanlah pengujian. Berikut ini hasil pengujian yang telah dilakukan berdasarkan rencana pengujian

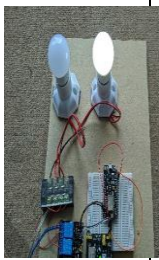
1. Pengujian Lampu 1

- Menyalakan Lampu 1 dengan Tombol pada Aplikasi

Tabel 3.1 Tabel Rencana Pengujian

No	Kelas Uji	Butir Uji	Jenis Pengujian
1	Lampu 1	Menyalakan dan mematikan lampu 1 melalui Tombol pada Aplikasi.	Black Box
		Menyalakan dan mematikan lampu 2 melalui Voice Recognition pada Aplikasi.	Black Box
2	Lampu 2	Menyalakan dan mematikan lampu 2 melalui Tombol pada Aplikasi.	Black Box
		Menyalakan dan mematikan lampu 2 melalui Voice Recognition pada Aplikasi.	Black Box

Tabel 3.2 Tabel Pengujian Menyalakan Lampu 1 Dengan Tombol

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Menekan tombol ON Lampu 1	Lampu 1 menyala		OK

- Mematikan Lampu 1 dengan Tombol pada Aplikasi

Tabel 3.3 Tabel Pengujian Mematikan Lampu 1 Dengan Tombol

Kasus dan Hasil Uji

Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Menekan tombol OFF lampu 1	Lampu yang awalnya menyala menjadi mati		OK

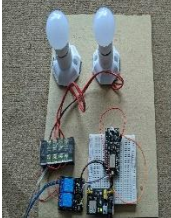
- c. Menyalakan Lampu 1 dengan Voice Recognition

Tabel 3.4 Tabel Pengujian Menyalakan Lampu 1 Dengan Voice Recognition

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Memasukan perintah suara "bedroom"	Lampu 1 menyala		OK

- d. Mematikan Lampu 1 dengan Voice Recognition

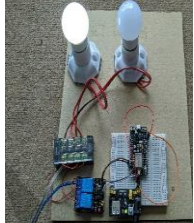
Tabel 3.5 Tabel Pengujian Menyalakan Lampu 1 Dengan Voice Recognition

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Memasukan perintah suara "night"	Lampu 1 yang awalnya menyala menjadi mati		OK

2. Pengujian Lampu 2

- a. Menyalakan Lampu 2 dengan Tombol pada Aplikasi

Tabel 3.6 Tabel Pengujian Menyalakan Lampu 2 Dengan Tombol

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Menekan tombol ON Lampu 2	Lampu 2 menyala		OK

- b. Mematikan Lampu 2 dengan Tombol pada Aplikasi

Tabel 3.7 Tabel Pengujian Menyalakan Lampu 2 Dengan Tombol

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Menekan tombol OFF lampu 2	Lampu 2 yang awalnya menyala menjadi mati		OK

- c. Menyalakan Lampu 2 dengan Voice Recognition

Tabel 3.8 Tabel Pengujian Menyalakan Lampu 2 Dengan Voice Recognition

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Memasukan perintah suara "kitchen"	Lampu 2 menyala		OK

- d. Mematikan Lampu 2 dengan Voice Recognition

Tabel 3.9 Tabel Pengujian Menyalakan Lampu 2 Dengan Voice Recognition

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Hasil Uji	Kesimpulan
Memasukkan perintah suara "apple"	Lampu 2 yang asalnya menyala menjadi mati		OK

3. Kesimpulan

Dari berbagai penjelasan yang telah diuraikan dalam laporan ini, maka dapat disimpulkan beberapa hal sebagai berikut:

1. Terwujudnya sebuah *prototype* sistem kendali lampu secara *wireless* melalui *smartphone*.
2. Terlepas dari kemampuan pengguna dalam pelafalan kalimat, sistem perintah suara atau *Voice Recognition* maupun tombol sebagai kontrol lampu berjalan dengan baik.
3. Pengguna merasa terbantu jika *prototype* ini di implementasikan ke skala rumah sesungguhnya.

5. Saran

Agar sebuah aplikasi / *game* dapat berjalan dengan baik dan dapat mencapai tujuan secara maksimal, maka dibutuhkannya hal-hal pendukung. Menyangkut hal ini maka rekomendasi yang diusulkan diantaranya:

1. Perlunya pengembangan dalam keragaman alat elektronik yang digunakan.
2. Mengembangkan kemampuan sensitifitas dan *wording* dari *Voice Recognition* agar

lebih mudah dioperasikan oleh semua kalangan.

3. Menindaklanjuti *prototype* ini dengan mengimplementasikannya pada skala rumah sesungguhnya.

DAFTAR PUSTAKA

- [1] Abdurrahman, Y. Jayusman and M. A. Rifqi, "SISTEM SMART HOME PENGONTROL PERALATAN ELEKTRIK RUMAH MELALUI INTERNET BERBASIS RASPBERRY PI," *Jurnal Teknologi Informasi dan Komunikasi*, pp. 23-40, 2016.
- [2] M. I. Rahayu, A. Sonjaya and A. Soepriadi, "PEMBUATAN PROTOTIPE KONTROL REL SLIDERKAMERA OTOMATIS MENGGUNAKAN MIKROKONTROLER," *Jurnal Teknologi Informasi dan Komunikasi*, vol. 4, no. 2, pp. 01 - 04, 2015.
- [3.] R. J. Robles dan T.-h. Kim, "Applications, Systems and Methods in Smart Home Technology: A Review," *International Journal of Advanced Science and Technology*, vol. 15, February 2010.
- [4.] Sudaryatno S., Analisis Rangkaian Listrik Jilid-1 (Rangkaian Arus Searah dan Arus BolakBalik Keadaan Mantap), (Bandung: Darpublic, 2012), hlm. 1.
- [5.] Rusdi M., Yani A., Sistem Kendali Peralatan Elektronik Melalui Media Bluetooth Menggunakan Voice Recognition.
- [6.] Erawan A., Karna N., Sanjoyo D.D, Desain Dan Implementasi Smarthome Pada Indekos. Universitas Telkom.
- [7.] Wicaksono F.M, Implementasi Modul Wifi NodeMCU ESP8266 Untuk Smart Home, 2007.