



**RANCANG BANGUN ALAT DISPLAY HARGA SECARA OTOMATIS
MENGUNAKAN LCD GRAFIS**

TUGAS AKHIR

**Program Studi
S1 Sistem Komputer**

**INSTITUT BISNIS
DAN INFORMATIKA**

stikom
SURABAYA

Oleh:

ADI BAROTO SARDADI

14410200024

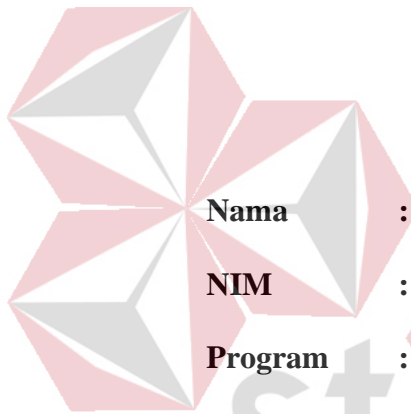
**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA
2018**

**RANCANG BANGUN ALAT DISPLAY HARGA SECARA OTOMATIS
MENGUNAKAN LCD GRAFIS**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Komputer



Oleh :

Nama : ADI BAROTO SARDADI

NIM : 14.41020.0024

Program : S1 (Strata Satu)

Jurusan : Sistem Komputer

stikom
SURABAYA

FAKULTAS TEKNOLOGI DAN INFORMATIKA

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

2018



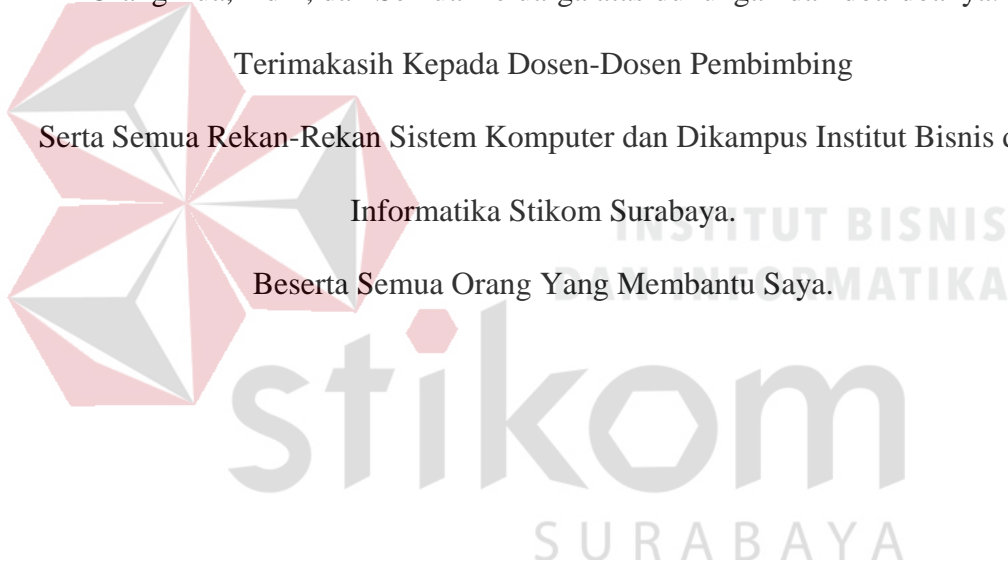
“Kehidupan ini dipenuhi dengan seribu macam kemanisan tetapi untuk mencapainya perlu seribu macam pengorbanan.”

- Donny Dhirgantoro-

Alhamdu'lilahi robbilalamin,
Segala Puji Bagi ALLAH SWT,
Shalawat dan salam tidak lupa selalu tercurah kepada baginda
Rasulullah Muhammad SAW.

Tugas Akhir ini Saya Persembahkan Kepada
Orang Tua, Adik, dan Semua Keluarga atas dukungan dan doa-doanya.

Terimakasih Kepada Dosen-Dosen Pembimbing
Serta Semua Rekan-Rekan Sistem Komputer dan Dikampus Institut Bisnis dan
Informatika Stikom Surabaya.
Beserta Semua Orang Yang Membantu Saya.



TUGAS AKHIR
RANCANG BANGUN ALAT DISPLAY HARGA SECARA OTOMATIS
MENGGUNAKAN LCD GRAFIS

Dipersiapkan dan disusun oleh

ADI BAROTO SARDADI

NIM : 14.41020.0024

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji

Pada : Agustus 2018

Susunan Dewan Penguji

Pembimbing

I. Dr. Susijanto Tri Rasmana, S.Kom., M.T.

NIDN. 0727097302

II. Heri Pratikno, M.T., MTCNA., MTCRE.

NIDN. 0716117302

Pembahas

I. Weny Indah Kusumawati, S.Kom., M.MT.

NIDN. 0721047201

Tugas Akhir ini telah diterima sebagai salah satu persyaratan

untuk memperoleh gelar Sarjana



FAKULTAS TEKNOLOGI
DAN INFORMATIKA

Dr. Jusak

Dekan Fakultas Teknologi dan Informatika

25/8 '18

21/8 '18

**SURAT PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH**

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, penulis :

Nama : Adi Baroto Sardadi
NIM : 14410200024
Program Studi : S1 Sistem Komputer
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Tugas Akhir
Judul Karya : **RANCANG BANGUN ALAT DISPLAY HARGA
SECARA OTOMATIS MENGGUNAKAN LCD GRAFIS**


Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, penulis menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah penulis tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama penulis sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut di atas adalah karya asli penulis, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka penulis.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka penulis bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada penulis.

Demikian surat pernyataan ini penulis buat dengan sebenarnya.

Surabaya, Agustus 2018




Adi Baroto Sardadi
NIM. 14.41020.0024

ABSTRAK

Dalam zaman millennial saat ini bisnis waralaba semakin berkembang dan selalu berinovasi untuk memenuhi kebutuhan sehari-hari. Munculnya bisnis retail yang semakin banyak saat ini memaksa para pelaku retail untuk membuat sebuah promosi harga untuk menarik perhatian para pembeli. Pada sebuah bisnis retail diwajibkan untuk memberi label harga, label harga pada bisnis retail saat ini masih menggunakan kertas, label harga tersebut di tempel oleh karyawan setiap kali harga mengalami perubahan. Masalah yang sering ditemui saat ini yaitu perbedaan harga antara di rak dan yang tertera pada mesin kasir.

Dari masalah yang terjadi tersebut akan di buat sebuah sistem yang akan digunakan untuk menggantikan fungsi kertas menjadi layar LCD *display* grafik yang dapat memberikan sebuah informasi nama barang, harga barang dan kode barang pada pembeli. Sistem tersebut akan menghubungkan antara aplikasi yang berada pada komputer dengan LCD yang berada di rak barang menggunakan jaringan *Wireless*.

Pada Tugas Akhir ini merupakan pengembangan dari Tugas Akhir yang telah di buat dengan memanfaatkan LCD TFT yang hanya dapat menampilkan data berupa text mengubah LCD TFT tersebut dengan LCD grafis yang dapat menampilkan data menjadi gambar dan menggunakan arduino sebagai pengolah data menuju LCD grafis .

Kata Kunci : Wemos D1 Mini, LCD *display*, *Wireless*, Arduino

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa Allah SWT yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir tepat waktu dan sebaik-baiknya tanpa adanya halangan yang berarti. Penulis mengambil judul “RANCANG BANGUN ALAT DISPLAY HARGA SECARA OTOMATIS MENGGUNAKAN LCD GRAFIS” ini sebagai salah satu syarat dalam menyelesaikan Tugas Akhir di Institut Bisnis dan Informatika Stikom Surabaya.

Pada kesempatan kali ini penulis juga ingin mengucapkan terima kasih kepada :

1. Orang tua dan adik penulis tecinta, karena dengan segala dukungan serta motivasinya mulai dari awal penulis mengerjakan Tugas Akhir ini sampai selesai.
2. Pimpinan Institut Bisnis dan Informatika Stikom Surabaya yang telah banyak memberi motivasi serta teladan yang dapat membantu penulis selama menempuh pembelajaran hingga saat ini.
3. Bapak Dr. Jusak selaku Dekan Fakultas Teknologi dan Informatika (FTI) Institut Bisnis dan Informatika Stikom Surabaya telah membantu proses penyelesaian Tugas Akhir yang dibuat oleh penulis dengan baik.
4. Bapak Pauladie Susanto, S.Kom., M.T., selaku Ketua Program Studi S1 – Sietem Komputer telah membantu dalam proses penyelesaian Tugas Akhir ini dengan sebaik-baiknya
5. Bapak Dr. Susjianto Tri Rasmana, S.Kom., M.T., selaku Dosen Pembimbing pertama dan Heri Pratikno, M.T., selaku Dosen Pembimbing kedua yang telah

meluangkan waktu dan pikiran serta dengan sabar membimbing penulis dari awal hingga Tugas Akhir ini dapat terselesaikan.

6. Ibu Weny Indah Kusumawati, S.Kom., M.MT., selaku Pembahas yang telah membimbing penulis serta memberi masukan dalam menyusun buku Tugas Akhir dan membimbing selama menempuh perkuliahan di Institut Bisnis dan Informatika Stikom Surabaya.
7. Seluruh Dosen pengajar Program Studi S1 – Sistem Komputer yang telah mendidik, memberi motivasi kepada penulis selama masa kuliah di Institut Bisnis dan Informatika Stikom Surabaya

Banyak hal dalam laporan Tugas Akhir ini yang masih perlu diperbaiki. Oleh karena itu penulis mengharapkan kritik dan saran yang dapat membangun dari semua pihak agar dapat menyempurnakan penulisan ini kedepannya. Penulis juga memohon maaf yang sebesar - besarnya bila terdapat kata – kata yang salah serta menyinggung perasaan pembaca. Akhir kata penulis mengucapkan banyak terima kasih yang sebesar – besarnya kepada para pembaca, semoga tulisan ini dapat bermanfaat dan memberi inspirasi lebih banyak untuk pembaca dan pembaca dapat berkarya melebihi karya penulis.

Surabaya, Agustus 2018

Penulis

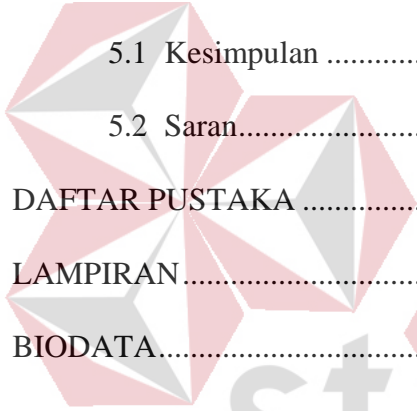
DAFTAR ISI

	Halaman
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	4
1.4 Tujuan	4
1.5 Manfaat	4
1.6 Sistematika Penulisan.....	5
BAB II LANDASAN TEORI	6
2.1 Sistem Komputer.....	6
2.2 Wemos	6
2.2.1 Wemos D1 Mini.....	7
2.2.2 <i>Microcontroller Chipset Pada Microcontroller Wemos</i> ...	9
2.2.3 Fitur – Fitur Wemos	10
2.3 Arduino UNO.....	12
2.4 Arduino IDE.....	14
2.5 <i>Liquid Crystal Display (LCD)</i>	15
2.6 <i>Router</i>	17

2.7	<i>Access Point</i>	17
2.8	Protokol TCP/IP	18
BAB III METODE PENELITIAN DAN PERANCANGAN SISTEM.....		20
3.1	Metode Penelitian.....	20
3.2	Prosedur Penelitian Sistem Keseluruhan.....	22
3.3	Perancangan Perangkat Keras	23
3.3.1	Rangkaian Arduino UNO Ke LCD TFT	23
3.3.2	Rangkaian Koneksi I2C Arduino UNO Ke Wemos.....	24
3.3.3	Algoritma Pengiriman Data Dari Server Ke Wemos	26
3.3.4	Algoritma Pengiriman Data Dari Wemos Ke Arduino... ..	28
3.4	Perancangan Perangkat Lunak	30
3.4.1	Algoritma Pengiriman Data	30
3.5	Perancangan Topologi Jaringan	35
3.5.1	Konfigurasi IP <i>Server</i>	36
3.5.2	Konfigurasi <i>Routing</i> Dan IP <i>Router</i>	38
3.5.3	Konfigurasi IP <i>Static</i> Pada Wemos D1 Mini	42
BAB IV HASIL PENGUJIAN DAN PENGAMATAN		44
4.1	Pengujian Wemos.....	44
4.1.1	Tujuan Pengujian Wemos	45
4.1.2	Alat Yang Digunakan Pengujian Wemos.....	45
4.1.3	Prosedur Pengujian wemos	45
4.1.4	Hasil Pengujian Wemos	50
4.2	Pengujian <i>Scaning Access Point</i> Oleh Wemos	51
4.2.1	Tujuan Pengujian <i>Scaning Access Point</i> Oleh Wemos... ..	51
4.2.2	Alat Yang Digunakan Pengujian <i>Scaning Access Point</i> Oleh Wemos.....	51

4.2.3	Prosedur Pengujian <i>Scaning Access Point</i> Oleh Wemos	52
4.2.4	Hasil Pengujian <i>Scaning Access Point</i> Oleh Wemos	53
4.3	Pengujian Menghubungkan WIFI Wemos Pada <i>Access Point</i>	54
4.3.1	Tujuan Pengujian Menghubungkan WIFI Wemos Pada <i>Access Point</i>	54
4.3.2	Alat Yang Digunakan Pengujian Menghubungkan WIFI Wemos Pada <i>Access Point</i>	54
4.3.3	Prosedur Pengujian Menghubungkan WIFI Wemos Pada <i>Access Point</i>	55
4.3.4	Hasil Pengujian Menghubungkan WIFI Wemos Pada <i>Access Point</i>	56
4.4	Pengujian <i>Scaning</i> Dan Pengiriman I2C Wemos Menuju LCD	57
4.4.1	Tujuan Pengujian <i>Scaning</i> Dan Pengiriman I2C Wemos Menuju LCD	57
4.4.2	Alat Yang Digunakan Pengujian <i>Scaning</i> Dan Pengiriman I2C Wemos Menuju LCD	58
4.4.3	Prosedur Pengujian <i>Scaning</i> Dan Pengiriman I2C Wemos Menuju LCD	58
4.4.4	Hasil Pengujian <i>Scaning</i> Dan Pengiriman I2C Wemos Menuju LCD	59
4.5	Pengujian Jarak Jangkauan WIFI Wemos.....	60
4.5.1	Tujuan Pengujian Jarak Jangkauan WIFI Wemos	61
4.5.2	Alat Yang Digunakan Pengujian Jarak Jangkauan WIFI Wemos	61
4.5.3	Prosedur Pengujian Jarak Jangkauan WIFI Wemos	61
4.5.4	Hasil Pengujian Pengujian Jarak Jangkauan WIFI Wemos	64
4.6	Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet.....	65
4.6.1	Tujuan Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet	65
4.6.2	Alat Yang Digunakan Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet	65

4.6.3	Prosedur Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet	66
4.6.4	Hasil Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet	69
4.7	Pengujian Sistem.....	70
4.7.1	Tujuan Pengujian Sistem.....	70
4.7.2	Alat Yang Digunakan Pengujian Sistem.....	70
4.7.3	Prosedur Pengujian Sistem.....	71
4.7.4	Hasil Pengujian Sistem	75
BAB V PENUTUP.....		77
5.1	Kesimpulan	77
5.2	Saran.....	78
DAFTAR PUSTAKA		79
LAMPIRAN.....		80
BIODATA.....		95



INSTITUT BISNIS
 DAN INFORMATIKA
stikom
 SURABAYA

DAFTAR GAMBAR

Halaman

Gambar 2.1 <i>Board</i> Wemos D1 Mini Sebelah Kiri Dan Wemos D1 Sebelah Kanan	7
Gambar 2.2 Wemos D1 Mini	7
Gambar 2.3 <i>Board</i> Arduino UNO	13
Gambar 2.4 Tampilan Arduino IDE Dan Skecth	15
Gambar 2.5 LCD TFT 2.4 Arduino Shield	16
Gambar 2.6 <i>Access Point</i>	17
Gambar 3.1 Blok Diagram Rancangan Tugas Akhir	20
Gambar 3.2 <i>Flowchart</i> Proses Penelitian Sistem Keseluruhan	22
Gambar 3.3 Rangkaian Arduino UNO Ke LCD TFT	24
Gambar 3.4 Rangkaian Koneksi I2C Arduino UNO Ke Wemos	25
Gambar 3.5 Rangkaian Pararel I2C / BUS koneksi	26
Gambar 3.6 Algoritma Penerimaan Data Dari <i>Server</i> Ke Wemos	26
Gambar 3.7 Algoritma Penerimaan Data Ke Arduino	28
Gambar 3.8 <i>Flowchart</i> Algoritma Pengiriman Data	30
Gambar 3.9 Tampilan Perangkat Lunak	31
Gambar 3.10 Tampilan <i>Source Code</i> Pada Visual Studio	35
Gambar 3.11 Topologi Jaringan Sistem	35
Gambar 3.12 Tampilan <i>Network Connection</i>	37
Gambar 3.13 Tampilan <i>Ethernet Properties</i>	37
Gambar 3.14 Tampilan <i>Ipv4 Properties</i>	38
Gambar 3.15 Setting IP LAN dan WLAN Mikrotik	39

Gambar 3.16 Konfigurasi <i>Interface Router</i> Mikrotik	40
Gambar 3.17 Konfigurasi WLAN dan Keamanan WLAN	41
Gambar 3.18 Setting <i>Routing</i> Mikrotik.....	42
Gambar 4.1 Tampilan Peralatan Yang Digunakan Untuk Pengujian.....	44
Gambar 4.2 <i>File Preferences</i>	46
Gambar 4.3 <i>Board Manager</i>	46
Gambar 4.4 Pencari Dan <i>Install Board Wemos</i>	47
Gambar 4.5 Pengecekan <i>Board Wemos D1 Mini</i> Sudah Terinstall.....	47
Gambar 4.6 Tampilan <i>Device Manager</i>	48
Gambar 4.7 Setting <i>Port Wemos</i>	49
Gambar 4.8 <i>Icon Upload</i>	49
Gambar 4.9 Proses <i>Upload Program</i> Ke Wemos.....	50
Gambar 4.10 <i>Upload Program</i> Ke Wemos Sukses	50
Gambar 4.11 Hasil Dari <i>Upload Program</i>	50
Gambar 4.12 Tersedia Jaringan WLAN.....	52
Gambar 4.13 <i>Icon Serial Monitor</i>	53
Gambar 4.14 Hasil <i>Scaning Access Point</i> Oleh Wemos	53
Gambar 4.15 Laptop Yang Didapat Dari <i>Access Point</i>	55
Gambar 4.16 Wemos Terhubung Ke <i>Access Point</i>	56
Gambar 4.17 Diagram Blok Pengujian <i>Scaning</i> Dan Pengiriman Data Dari Wemos Menuju LCD	57
Gambar 4.18 Hasil <i>Serial Monitor Scaning I2C</i>	59
Gambar 4.19 Hasil Pengiriman Data I2C Wemos Menuju LCD TFT	60
Gambar 4.20 Diagram Blok Pengujian Jangkauan WIFI Wemos	60
Gambar 4.21 Wemos Berhasil Menjadi <i>Access Point</i>	62

Gambar 4.22	<i>Access Point</i> Wemos Terdeteksi Oleh Laptop	62
Gambar 4.23	AP Yang Dibuat Wemos Terlihat Oleh Wemos <i>Client</i>	63
Gambar 4.24	Hasil <i>Scaning</i> WIFI (Kanan AP Terlihat Dan Kiri AP Tidak Terlihat	64
Gambar 4.25	Diagram Blok Pengujian Penerimaan Data Dari PC Menuju Wemos	65
Gambar 4.26	Wemos Siap Untuk Telnet	66
Gambar 4.27	IP Yang Didapatkan Laptop Dari <i>Access Point</i>	67
Gambar 4.28	<i>Software</i> Pengujian Penerimaan Wemos.....	68
Gambar 4.29	Data Yang Diterima Wemos (Kiri Awal Dan Kanan Akhir Data).....	68
Gambar 4.30	Hasil <i>Scaning</i> Wemos ke 1.....	71
Gambar 4.31	IP Address Wemos Ke 1.....	72
Gambar 4.32	Tampilan LCD Menunjukkan <i>Ready</i>	72
Gambar 4.33	Tampilan <i>Software</i> Aplikasi Display Harga.....	73
Gambar 4.34	Data Berhasil Di Tampilkan LCD 1	73
Gambar 4.35	Data Berhasil Di Tampilkan LCD 2.....	74

DAFTAR TABEL

	Halaman
Tabel 2.1 Spesifikasi Wemos D1 Mini	8
Tabel 2.2 Konfigurasi Pin Wemos	12
Tabel 2.3 Karakteristik Arduino UNO.....	13
Tabel 2.4 Arduino <i>Pin Connections</i>	16
Tabel 4.1 Hasil Pengujian Jarak Jangkauan Wemos.....	64
Tabel 4.2 Hasil 10 Kali Percobaan Pengiriman Data.....	69
Tabel 4.3 Hasil Percobaan Skenario	75



BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam kehidupan millennial saat ini bisnis waralaba semakin berkembang dan selalu berinovasi untuk memenuhi kebutuhan sehari-hari. Perubahan cara pandang masyarakat terhadap bisnis waralaba menyebabkan adanya perubahan pola berbelanja masyarakat yang semakin selektif. Menurut Suhartanto dan Nuralia, (2001,27). “Bisnis waralaba yang secara tradisional dipandang sebatas penyedia barang dan jasa telah berkembang menjadi tidak sekedar tempat berbelanja tetapi juga tempat rekreasi dan bersosialisasi, sebagai konsekuensinya bisnis waralaba yang semula dikelola secara tradisional berubah menjadi bisnis yang semakin inovatif, dinamis, dan kompetitif”. Perubahan gaya hidup masyarakat inilah yang melatarbelakangi berkembangnya bisnis pasar swalayan.

Menurut Swastha (2003, 241). “ Harga merupakan komponen yang berpengaruh langsung terhadap proses penentuan dimana seorang akan berbelanja ”. Keputusan berbelanja biasanya memerlukan pertimbangan yang benar-benar mendukung dan dapat menguntungkan pembeli seperti faktor harga. Untuk itu swalayan harus selalu membuat inovasi dengan cara mengadakan promosi dan pembaruan harga secara rutin agar tidak kalah dengan para pesaingnya. Swalayan harus melakukan *update* harga pada setiap rak-rak barang yang ada karena perubahan harga.

Sistem untuk perubahan harga yang digunakan saat ini hanya sebatas *update* di bagian kasir. Karena komputer pada kasir mempunyai *database* untuk harga

barang, dari *database* tersebut pegawai baru bisa mencetak label barang yang berada di rak melalui sistem yang ada. Tetapi sistem ini belum sampai menjamah ke label harga di rak barang.

Perkembangan teknologi yang semakin maju saat ini mencakup semua bidang dalam kehidupan manusia. Hal ini dapat dimanfaatkan untuk membangun sebuah sistem yang berbasis *controller* untuk menampilkan label harga dengan otomatis. Sehingga bila *database* harga barang berubah harga di rak barang juga ikut mengalami perubahan.

Sebelumnya Tugas Akhir ini pernah di terapkan oleh mahasiswa Sistem Komputer Deny Setyo Utomo dengan judul “PRODUCT PRIZE DISPLAY USING WEMOS” dengan *Microcontroller* sebagai sarana yang digunakan untuk membantu mewujudkan sistem label harga otomatis. *Microcontroller* tersebut akan mengendalikan LCD sebagai pengganti label harga yang selama ini menggunakan kertas. Hal ini dapat dilakukan dengan memberi pengalamatan disetiap LCD agar 1 *microcontroller* dapat memberi perintah kepada lebih dari 1 LCD.

Kelemahan sistem tersebut yaitu masih menggunakan LCD standart yang hanya dapat menampilkan harga saja. Sesuai peraturan Menteri Perdagangan “Pada dasarnya setiap pelaku usaha yang memperdagangkan barang secara eceran atau jasa kepada konsumen wajib mencantumkan harga barang atau tarif jasa secara jelas, mudah dibaca dan mudah dilihat (Pasal 2 ayat (1) Permendag 35/2013). Dan pada Pasal 1 angka 3 dari PP No. 69 Tahun 1999 menentukan yang dimaksud dengan “Label pangan adalah setiap keterangan mengenai pangan yang berbentuk gambar, tulisan, kombinasi keduanya, atau bentuk lain yang disertakan pada

pangan, dimasukkan ke dalam, ditempelkan pada, atau merupakan bagian kemasan pangan, yang selanjutnya dalam Peraturan Pemerintah ini disebut Label”.

Sehubung dengan hal tersebut maka penulis akan mengembangkan topik yang sudah ada dengan mengganti LCD text yang hanya dapat menampilkan harga saja menjadi LCD grafis, sehingga dapat mencantumkan harga barang, *barcode*, dan keterangan lainnya. Dalam hal tersebut penulis mengambil topik untuk menjadikan bahan penulisan Tugas Akhir **“RANCANG BANGUN ALAT DISPLAY HARGA SECARA OTOMATIS MENGGUNAKAN LCD GRAFIS”**

1.2 Rumusan Masalah

Tugas Akhir ini merupakan pengembangan Tugas Akhir yang sudah dibuat. Berdasarkan Tugas Akhir sebelumnya yang di kerjakan Deny Setyo Utomo tentang “Product Price Display Using Wemos”, masih perlu pengembangan dalam sisi tampilan dimana tampilan yang dimunculkan belum dapat memenuhi kriteria label harga yang diatur oleh kementerian perdagangan, dimana “ Setiap Pelaku Usaha yang memperdagangkan Barang secara Eceran dan/ atau Jasa kepada Konsumen wajib mencantumkan Harga Barang atau Tarif Jasa secara jelas, mudah dibaca dan mudah dilihat” (Pasal 2 ayat (1) Permendag 35/2013). Secara umum harus memiliki kode barang/ *barcode* yang sama antara barang yang akan dijual dengan kode barang yang ditampilkan pada LCD Grafis, serta keterangan di layar LCD harus lengkap mengenai Nama Barang, Kode barang, Harga, dan keterangan lainnya yang diperlukan .

1.3 Batasan Masalah

Pada pengimplementasian alat ini terdapat beberapa batasan masalah antara lain :

- a. Dalam 1 Wemos akan terkoneksi Arduino dan diolah lalu di tampilkan dalam LCD Grafis.
- b. Dapat mengirimkan informasi nama, harga barang, label barang, serta keterangan lainnya dari komputer / *server* ke Wemos untuk di tampilkan di LCD grafis .
- c. *Controller* menggunakan Wemos D1 mini dan Arduino Uno.
- d. Dalam 1x pengiriman data hanya dapat mengirim 32 karakter.

1.4 Tujuan

Tujuan dilaksanakan Tugas Akhir tentang **RANCANG BANGUN ALAT DISPLAY HARGA SECARA OTOMATIS MENGGUNAKAN LCD GRAFIS** adalah untuk meminimalisir perbedaan harga antara di kasir dan di rak barang, secara mudah, cepat, dan *portable*. Label harga yang akan di tampilkan meliputi Nama Barang, *Barcode* atau kode barang, Harga Barang, dan keterangan lainnya apabila di perlukan.

1.5 Manfaat

Dengan menggunakan LCD Grafis ini, dapat memperbarui harga secara mudah tidak perlu lagi menggunakan kertas yang di print lalu di letakkan pada rak, karena sering terjadi kesalahan pada rak barang ketika karyawan supermarket lupa untuk mengganti harga yang sudah berubah.

1.6 Sistematika Penulisan

BAB I PENDAHULUAN

Dalam bab ini penulis menguraikan Latar Belakang Pemilihan Judul, Perumusan Masalah, Batasan Masalah, Tujuan, dan sistematika penulisan, sehingga permasalahan tersebut memiliki titik fokus dan tidak mengambang dari judul yang telah dibuat.

BAB II LANDASAN TEORI

Dalam bab ini penulis menguraikan yang terdiri dari Landasan teori sistem, program yang digunakan dalam pembuatan Tugas Akhir ini, serta konsep-konsep baru dalam menyelesaikan masalah yang berkenaan dengan topik dan fokus.

BAB III METODE PENELITIAN DAN PERANCANGAN SISTEM

Dalam bab ini penulis akan menguraikan rancangan sistem yang akan dibuat berupa rangkaian elektronik, hardware, *software*, dan pemrograman.

BAB IV HASIL PENGUJIAN DAN PENGAMATAN

Dalam Bab ini penulis akan membahas mengenai hasil dari penelitian dan pengamatan sistem yang dibuat.

BAB V PENUTUP

Dalam bab ini penulis akan menyimpulkan hasil analisis dan rancangan sistem dalam rangka menjawab tujuan penelitian yang dilakukan, serta saran-saran yang penulis berikan untuk lebih memaksimalkan alat ini.

BAB II

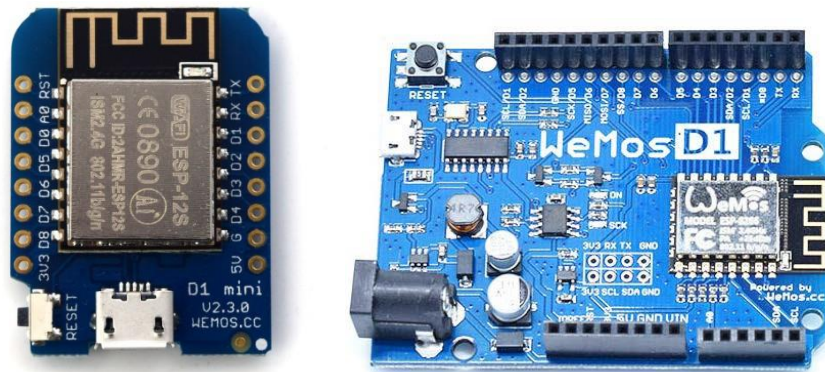
LANDASAN TEORI

2.1 Sistem Komputer

Sistem komputer adalah kumpulan perangkat-perangkat komputer yang saling berhubungan dan berinteraksi satu sama lain untuk melakukan proses pengolahan data, sehingga dapat menghasilkan informasi yang diharapkan oleh penggunanya. Perangkat tersebut meliputi, perangkat lunak (*software*), perangkat keras (*hardware*) dan data, mesin pengolah data yang biasanya meliputi komputer, program aplikasi, jaringan, dan manusia untuk menghasilkan informasi. Sistem komputer ada pada hampir setiap perusahaan atau instansi untuk mendukung kegiatan bisnis mereka sehari-hari.

2.2 Wemos

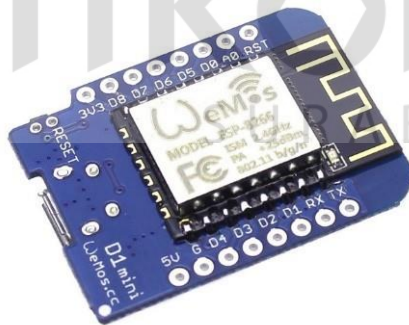
Wemos merupakan salah satu modul *board* yang dapat berfungsi dengan Arduino. Wemos juga merupakan sebuah *microcontroller* yang dikembangkan berbasis ESP8266. *Microcontroller* Wemos dibuat sebagai solusi dari mahalnya sebuah sistem *wireless* berbasis *microcontroller* lainnya. Dengan menggunakan *microcontroller* Wemos biaya yang dikeluarkan untuk membangun sistem *microcontroller* berbasis *wifi* sangat mudah dan murah, biaya yang dikeluarkan apabila membangun sistem *wifi* dengan menggunakan *microcontroller* Arduino Uno, *wifi shield* dan *microcontroller* Wemos adalah kurang lebih sepersepuluhnya. Yang berbeda pada *Microcontroller* Wemos ini adalah kemampuannya untuk menyediakan fasilitas konektivitas Wifi dengan mudah.



Gambar 2. 1 Board Wemos D1 Mini Sebelah Kiri Dan Wemos D1 Sebelah Kanan

2.2.1 Wemos D1 Mini

Wemos D1 Mini merupakan *board microcontroller open source* yang menggunakan ESP8266 sebagai *microcontrollernya*. Wemos memiliki 12 *pin input / output*, 1 *pin* digunakan untuk analog *input* maksimal 3.2V, 11 *pin* sisanya digunakan untuk digital *input / output*. Lalu juga terdapat tombol reset, koneksi USB dengan kabel data micro USB.



Gambar 2. 2 Wemos D1 Mini

Pada dasarnya pemrograman Wemos ini dapat menggunakan Arduino IDE yang sudah di *setting* pada setingan *board* Arduino IDE menjadi setingan Wemos.

Berikut adalah spesifikasi dari Wemos D1 Mini:

Tabel 2. 1 Spesifikasi Wemos D1 Mini

Microcontroller	ESP8266EX
Operting Voltage	3.3V
Digital I/O Pins	11
Analog Input Pins	1 (Max input 3.2V)
Clock Speed	80 MHz/ 160 MHz
Flash	4M bytes
Length	34,2 mm
Width	25.6 mm
Weignt	10g

A. Daya (*Power*)

Wemos Dapat diaktifkan melalui koneksi USB atau dengan catu daya *external*. *External* (non – USB) daya dapat berasal baik dari adaptor DC atau Baterai. Adaptor ini ditancapkan pada *pin* 5V pada Wemos D1 Mini.

Board dapat beroperasi pada rentang tegangan 3,3V – 7V. jika tegangan kurang dari 3.3V maka Wemos tidak akan menyala atau data akan kacau karena kurangnya konsumsi daya. Sebaliknya jika tegangan lebih dari 5V maka Wemos akan terbakar dan rusak karenan kelebihan tegangan

Pin listrik pada Wemos D1 Mini :

5V : Dapat digunakan sebagai *input* atau *output* sumber daya sebesar 5V, jika daya *external* bukan dari USB maka *pin* ini untuk masukan daya, bila daya sudah menggunakan USB maka *pin* ini sebagai *output* tegangan

3,3V : Dapat digunakan sebagai *input* atau *output* sumber daya 3,3V, jika daya *external* bukan dari USB maka *pin* ini untuk masukan daya, bila daya sudah menggunakan USB maka *pin* ini sebagai *output* tegangan

GND : *Pin* untuk penetral *noise* atau juga berperan sebagai 0V, pada aplikasi ini.

Wemos ini memiliki konsumsi daya 3,3V, *clock speed* 160Mhz, konektivitas *wifi*, dan memori yang digunakan cukup besar yaitu 4 MB. Dalam operasi kerjanya Wemos ini dapat bekerja di rentang suhu antara 40 °C - 125 °C. Walaupun modul ini bukan Arduino, namun modul ini mendukung pemrograman menggunakan Arduino IDE, beserta *library* dan fungsi-fungsinya yang lain. Untuk koneksi ke *laptop* atau ke catu daya *microcontroller* Wemos ini menggunakan konektor *micro* USB yang umum digunakan untuk kabel data *smartphone* Android.

2.2.2 Microcontroller chipset pada microcontroller Wemos

Pada *microcontroller* Wemos memiliki 2 buah *chipset* yang digunakan sebagai otak kerja perangkat tersebut antara lain adalah :

a. *Chipset* ESP8266

Chipset ESP8266 adalah sebuah *chip microcontroller* yang memiliki fitur *wifi* yang mendukung *stack* TCP/IP. *Chip* ini memungkinkan *microcontroller* untuk terhubung ke jaringan *wifi* pada frekuensi 2.4GHz dan membuat koneksi TCP/IP hanya dengan *command* yang sederhana seperti gaya *hayes*. Dengan *clock* 80 MHz *chip* ini dibekali dengan 4MB *external* RAM, mendukung format IEEE 802.11 b/g/n sehingga tidak menyebabkan interferensi bagi yang lain. Dari sisi keamanan

chip ini sudah cukup aman digunakan karena mendukung enkripsi WEP dan WPA. *Chip* ini mempunyai 16 GPIO *pin* yang bekerja pada 3.3 Volt, dan 1 *pin* ADC dengan resolusi 10 *bit*.

b. Chipset CH340

Chipset CH340 adalah sebuah *chip* yang berfungsi untuk mengubah usb menjadi *serial interface* . Sebagai contohnya adalah aplikasi USB converter IrDA atau USB *converter to printer*. Dalam mode *serial interface*, *chip* ini digunakan untuk memperbesar sinyal *asynchronous serial interface* komputer atau mengubah perangkat *serial interface* umum untuk berhubungan dengan *bus* USB langsung.

2.2.3 Fitur – fitur Wemos

Ada beberapa fitur-fitur perangkat keras yang dapat ditemukan pada modul *microcontroller* Wemos antara lain sebagai berikut:

a. Pin Digital

Microcontroller Wemos secara fisik mempunyai *pin* digital berjumlah 9 *pin* yang dimulai dari d0 – d8. Namun secara program mempunyai 16 *pin* digital yang beberapa *pin* di definisikan menjadi 2 alamat I/O. *Pin* digital ini dapat digunakan menjadi *input* maupun *output* sama fungsinya dengan *pin* digital *input output* pada Arduino maupun *microcontroller* yang lain. Selain itu *pin* digital pada *microcontroller* Wemos sudah dapat digunakan untuk PWM (*Pulse Width Modulator*).

b. Pin analog

Microcontroller Wemos hanya mempunyai 1 buah *pin* analog yang dapat digunakan sebagai *input* untuk ADC yang memiliki 10 *bit* resolusi dengan nilai

tegangan maksimal 3.2 Volt. *Pin* ini juga dapat digunakan sebagai *pin* digital *input* *output*. Selain itu *pin* ini juga memiliki resistor *pullup* namun untuk menggunakan *pullup* ini cukup membuat repot karena ada beberapa aturan yang harus di lakukan terlebih dahulu.

c. Memori

Ada 3 jenis memori yang digunakan dalam *microcontroller* Wemos ini, antara lain :

1. RAM untuk menyimpan memori instruksi sebesar 64KB
2. RAM untuk menyimpan data sebesar 96KB
3. *External* QSPI *flash* untuk menimpa listing program sebesar 4MB

d. I2C

Inter Integrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didisain khusus untuk mengirim maupun menerima data. Sistem I2C terdiri dari saluran SCL (*Serial Clock*) dan SDA (*Serial Data*) yang membawa informasi data antara I2C dengan pengontrolnya.

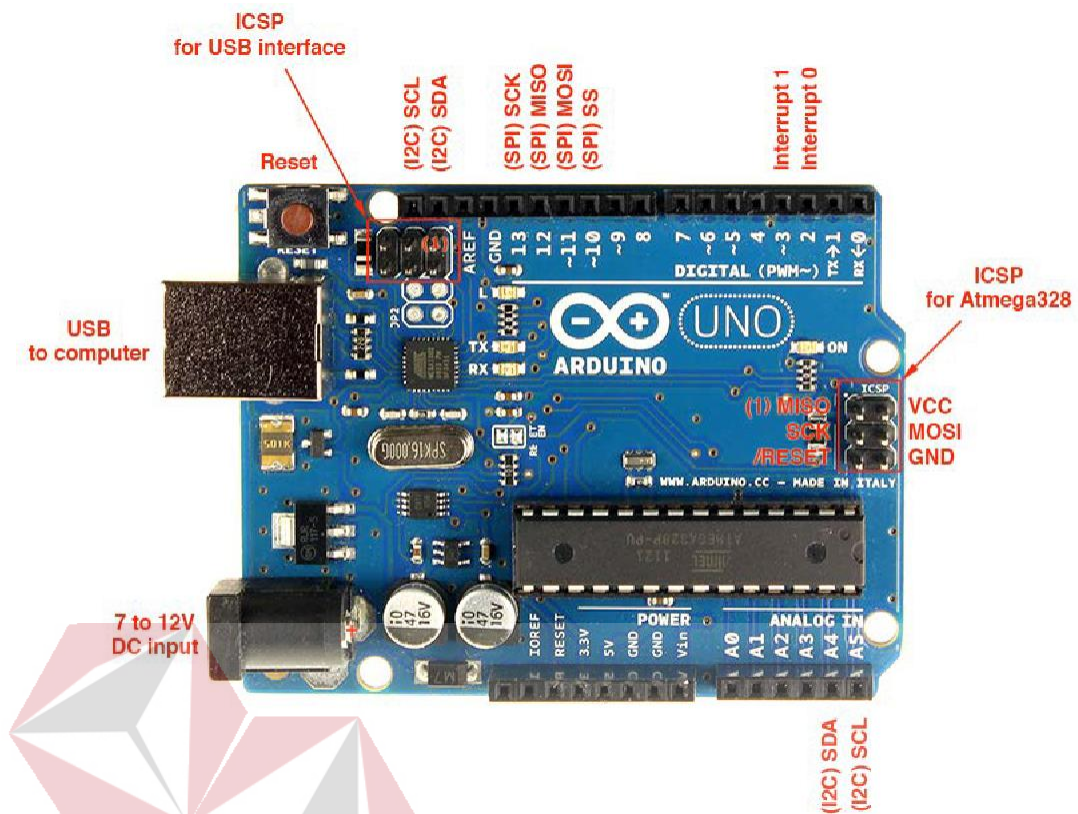
Microcontroller Wemos ini didukung dengan I2C yang berada pada D4 sebagai *serial data* dan D5 sebagai *serial clock*. Yang akan lebih memudahkan untuk perangkat ini mengendalikan *hardware* lain yang mendukung I2C pula.

Tabel 2. 2 Konfigurasi Pin Wemos

Pin	Fungsi	ESP-8266 Pin
TX	TXD	TXD
RX	RXD	RXD
A0	Analog input, max 3.3 input	A0
D0	I/O	GPIO16
D1	I/O, SCL	GPIO5
D2	I/O, SDA	GPIO4
D3	I/O, 10k Pull-up	GPIO0
D4	I/O, 10k Pull-up, Builtin_Led	GPIO2
D5	I/O, SCK	GPIO14
D6	I/O, Mosi	GPIO12
D7	I/O, Miso	GPIO13
D8	I/O, 10k Pull-Down, SS	GPIO15
G	Ground	GNF
5V	5V	-
3V3	3,3V	3,3V
RST	Reset	Reset

2.3 Arduino UNO

Arduino UNO adalah sebuah *board* mikrokontroler yang didasarkan pada ATmega328. Arduino UNO mempunyai 14 pin digital *input/output* (6 di antaranya dapat digunakan sebagai *output* PWM), 6 *input* analog, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah power jack, sebuah ICSP header, dan sebuah tombol reset. Arduino UNO memuat semua yang dibutuhkan untuk menunjang mikrokontroler, mudah menghubungkannya ke sebuah computer dengan sebuah kabel USB atau mensuplainya dengan sebuah adaptor AC ke DC atau menggunakan baterai untuk menggunakannya.



Gambar 2. 3 Board Arduino UNO

Tabel 2. 3 Karakteristik Arduino UNO

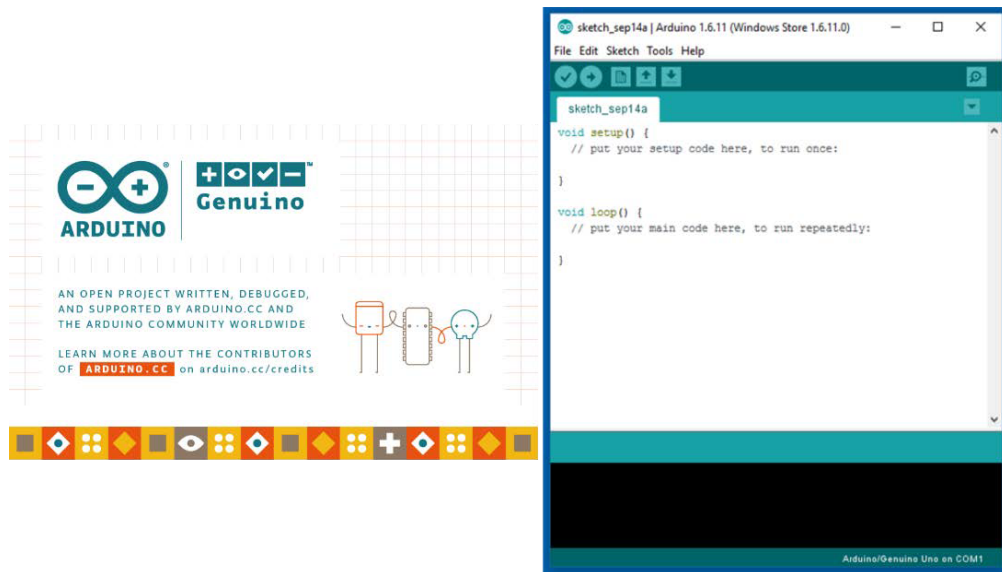
Mikrokontroler	ATmega328
Tegangan pengoperasian	5V
Tegangan input yang disarankan	7-12V
Batas tegangan input	6-20V
Jumlah pin I/O digital	14 (6 di antaranya menyediakan keluaran PWM)
Jumlah pin input analog	6
Arus DC tiap pin I/O	40 mA
Arus DC untuk pin 3.3V	50 mA
Memori Flash	32 KB (ATmega328), sekitar 0.5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

2.4 Arduino IDE

Software Arduino IDE adalah sebuah *software* yang digunakan untuk membuat program untuk memberi perintah kepada Arduino. Dengan menggunakan bahasa C++ yang dikembangkan oleh Arduino. Pemrograman dengan Arduino IDE dimudahkan lagi dengan banyaknya *library* yang disediakan. *Software* IDE Arduino terdiri dari 3 (tiga) bagian antara lain :

1. *Editor program*, untuk menulis dan mengedit program dalam bahasa *processing*. *Listing* program pada Arduino disebut *sketch*.
2. *Compiler*, sebuah modul yang berfungsi mengubah bahasa *processing* (kode program) menjadi kode biner karena kode biner merupakan bahasa program yang dipahami oleh mikrokontroler.
3. *Uploader*, sebuah modul yang berfungsi memasukkan kode biner kedalam memori mikrokontroler.

Perangkat lunak yang ditulis menggunakan Arduino IDE disebut *sketch*. *Sketch* ditulis pada editor teks dan disimpan dengan *extensifile* .ino. Struktur perintah pada Arduino secara garis besar terdiri dari dua bagian yaitu *void setup* dan *void loop*. *Void setup* berisi perintah yang akan dieksekusi hanya satu kali sejak Arduino dihidupkan sedangkan *void loop* berisi perintah yang akan dieksekusi berulang-ulang selama Arduino dinyalakan.



Gambar 2. 4 Tampilan Arduino IDE Dan Sketch

2.5 Liquid Crystal Display (LCD)

Display elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf maupun grafis. LCD adalah suatu jenis *display* elektronik. LCD merupakan media tampil yang menggunakan kristal cair sebagai penampil utama. LCD sudah digunakan diberbagai bidang misalnya alal-alat elektronik seperti televisi, kalkulator dan layar komputer. Salah satu jenis LCD (*Liquid Crystal Display*). Yaitu 2.4 inch TFT LCD Touch Screen Module Shield adalah salah satu shield yang dapat digunakan oleh Arduino Uno atau Arduino Mega untuk menampilkan visualisasi grafis, bahkan dilengkapi dengan LCD layar sentuh yang dapat mendeteksi jari tangan atau stylus. Jenis LCD TFT Module ini paling umum digunakan sebagai display unit dan *input* device pada proyek Arduino. LCD display ini dapat menampilkan data berupa Text, grafis ataupun gambar bitmap.



Gambar 2. 5 LCD TFT 2.4 Arduino Shield

Dan terdapat tabel masing – masing Pin TFT LCD 2.4 Arduino Shield.

Tabel 2. 4 Arduino *Pin Connections*

Arduino Pin	LCD Shield Pin	Use
3.3V	3.3V	Power
5V	5V	Power
GND	GND	Power
A0	LCD_RD	LCD Control
A1	LCD_WR TOUCH_YP	LCD Control / Touch Data
A2	LCD_RS TOUCH_XM	LCD Control / Touch Data
A3	LCD_CS	LCD Control
A4	LCD_RST	LCD Reset
D2	LCD_D2	LCD Data
D3	LCD_D3	LCD Data
D4	LCD_D4	LCD Data
D5	LCD_D5	LCD Data
D6	LCD_D6 / TOUCH XP	LCD Data/ Touch Data
D7	LCD_D7 / TOUCH YM	LCD Data / Touch Data
D8	LCD_D0	LCD Data
D9	LCD_D1	LCD Data
D10	SD_CS	SD Select
D11	SD_DI	SD Data
D12	SD_DO	SD Data
D13	SD_SCK	SD Clock

2.6 Router

Router adalah perangkat keras jaringan komputer yang dapat di gunakan untuk menghubungkan beberapa jaringan yang sama atau berbeda. *Router* adalah sebuah alat untuk mengirimkan paket data melalui jaringan atau internet untuk dapat menuju tujuannya, proses tersebut dinamakan *routing*.

Router memiliki fungsi utama untuk membagi atau mendistribusikan IP Address, baik itu secara statis ataupun DHCP (*Dynamic Host Configuration Protocol*) kepada semua komputer yang terhubung ke *router* tersebut. Dengan adanya IP Address yang unik yang dibagikan *router* tersebut kepada setiap komputer dapat memungkinkan setiap komputer untuk saling terhubung serta melakukan komunikasi, baik itu pada LAN, WLAN, atau internet.

2.7 Access Point

Access point adalah sebuah perangkat jaringan yang berisi sebuah *transceiver* dan antena untuk transmisi dan menerima sinyal ke dan dari *clients remote*. Dengan *access point* (AP) *clients wireless* bisa dengan cepat dan mudah untuk terhubung kepada jaringan LAN kabel secara *wireless*.



Gambar 2. 6 *Access Point*

2.8 Protokol TCP/IP

TCP/IP adalah sebuah perangkat lunak (*software*) yang digunakan untuk mengirimkan dan menerima informasi dari satu komputer ke komputer lain. TCP/IP terdiri dari dua komponen yaitu *Internet Protocol (IP)* dan *Transmission Control Protocol (TCP)*.

Informasi yang akan dikirimkan terbentuk dalam sebuah paket dengan ukuran tertentu, bila informasi yang akan dikirimkan berukuran besar maka informasi tersebut dibagi-bagi dalam beberapa paket. Kemudian paket-paket ini dikirimkan ke jaringan. Bila disetiap ujung jaringan terdapat komputer yang mengerti IP maka komputer tersebut dapat menerima paket tersebut. Komputer penerima akan menggabungkan lagi paket-paket yang diterimanya menjadi sebuah informasi yang siap disajikan.

TCP bekerja sama dengan protokol IP untuk memastikan bahwa informasi yang dikirimkan melewati sebuah jaringan diterima oleh komputer tujuan. TCP juga memastikan bahwa penggabungan paket-paket informasi di komputer tujuan berjalan dengan baik dan sesuai dengan informasi awal di komputer pengirim. TCP akan meminta pengiriman kembali paket-paket informasi ke komputer tujuan bila beberapa paket-paket informasi itu hilang atau rusak.

TCP/IP terdiri dari sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi data. Berkat prinsip ini, tugas masing-masing protokol menjadi jelas dan sederhana. Protokol yang satu tidak perlu mengetahui cara kerja protokol yang lain, sepanjang ia masih bisa saling mengirim dan menerima data.

Berkat penggunaan prinsip ini, TCP/IP menjadi protokol komunikasi data yang fleksibel. TCP/IP dapat diterapkan dengan mudah di setiap jenis komputer dan interface jaringan, karena sebagian besar isi kumpulan protokol ini tidak spesifik terhadap satu komputer atau peralatan jaringan tertentu. Agar TCP/IP dapat berjalan diatas *interface* jaringan tertentu, hanya perlu dilakukan perubahan pada protokol yang berhubungan dengan *interface* jaringan saja.



BAB III

METODE PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metode Penelitian



Gambar 3. 1 Blok Diagram Rancangan Tugas Akhir

Dalam pengerjaan Tugas Akhir ini metode penelitian yang digunakan adalah Metode penelitian rancang bangun *hardware* berupa wemos untuk penerima data dari *server* dan Arduino sebagai penerima dan pengolah data yang di kirim dari wemos, lalu data yang sudah di olah tersebut dikirim melalui jalur komunikasi I2C dan di tampilkan ke LCD Grafis. Pada rancang bangun *software* dengan membuat sebuah aplikasi untuk *input* data dari *user* dan mengirimkan data tersebut ke *Wemos*. Dengan ini penulis akan untuk menguji Arduino untuk menerima suatu informasi data yang di kirim dari *Wemos* dan mengolah informasi yang sudah

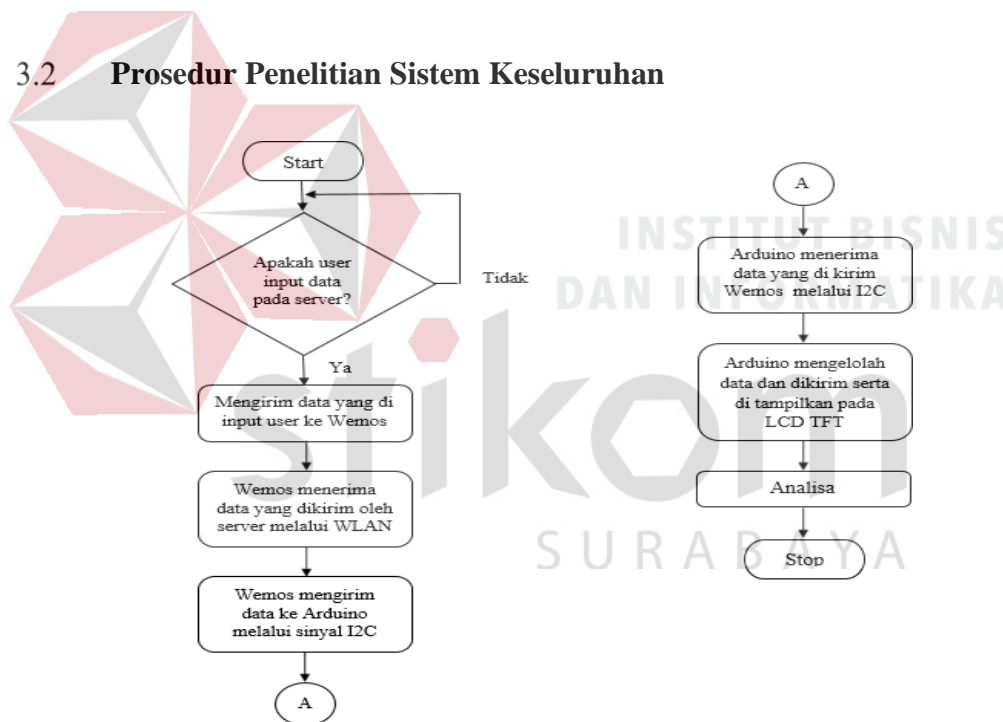
didapat sesuai dengan permasalahan yang terjadi. Studi kepustakaan dilakukan untuk mencari teori atau informasi dari buku, materi perkuliahan serta literatur dari internet, jurnal, dan artikel – artikel yang berkaitan dengan penelitian.

Penjelasan blok diagram pada rancangan penelitian yang menjelaskan tentang proses pengiriman data yang di *input* oleh *user* melalui *Wemos* menuju *Arduino*:

1. Proses awal yang akan dilakukan oleh *user* adalah menginput data menuju ke *server*. Data yang di inputkan oleh *user* yaitu berupa text IP tujuan, Alamat I2C, nama barang, harga barang, dan kode barang yang akan di munculkan pada LCD TFT. Alamat IP tujuan adalah alamat *Wemos* yang akan diproses, Alamat yang akan dituju berupa alamat I2C yang dimiliki oleh LCD. Alamat IP dan alamat I2C ini harus di berikan agar data yang di kirim tepat sampai tujuan.
2. Pada bagian berikutnya merupakan bagian *server* yang bertugas untuk mengolah *input* data yang diberikan oleh *user*, data *input* yang sudah di olah tersebut akan di kirim kepada *Wemos*. Data yang dikirim oleh *server* adalah berupa Alamat I2C, nama barang, harga barang, dan kode barang . *server* akan mengirimkan data tersebut kepada *Wemos* sesuai dengan IP tujuan yang telah di masukan oleh *user*.
3. Pada bagian *router*, Paket data yang akan dikirim oleh *server* pada *router* akan dilihat alamat IP tujuannya. Kemudian *router* akan menyebarkan broadcast kepada semua yang terhubung ke *Accses Point*. *Router* ini berfungsi untuk membuat jalan paket data yang diterima dan mengirimkan kepada IP tujuan.

4. Paket data yang telah dikirim akan ditangkap oleh Wemos sesuai dengan IP yang dituju. Wemos akan membuka paket data yang berisi alamat I2C, nama barang, harga barang, dan kode barang. Wemos akan mengambil data alamat I2C untuk mengirimkan data kembali menuju LCD TFT melalui module I2C yang ada pada Arduino, data tersebut berupa data yang ingin di tampilkan seperti nama barang, harga barang, dan barcode barang.
5. Melakukan pengujian dengan parameter *throughput*, paket *loss*, dan *delay* mulai dari paket dikirim dari *server* sampai tampil ke LCD TFT.

3.2 Prosedur Penelitian Sistem Keseluruhan



Gambar 3. 2 Flowchart Proses Penelitian Sistem Keseluruhan

1. Proses dimulai dengan *user* melakukan *input* data di komputer *server*, lalu selanjutnya data yang dimasukan oleh *user* akan dikirim oleh *server* menuju ke Wemos melalui jaringan.

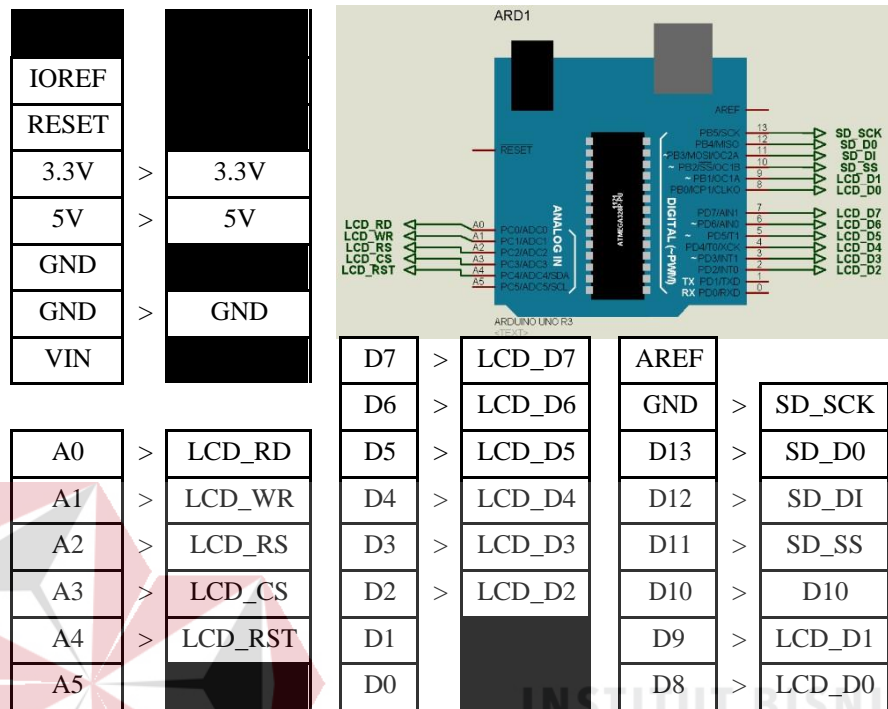
2. Setelah wemos menerima data dari *server* wemos akan menerjemahkan data tersebut.
3. Data yang dikeluarkan oleh wemos akan di terima modul LCD grafis untuk ditampilkan di layar LCD.
4. Lalu analisis yang akan dilakukan adalah apakah data yang dikirim oleh *server* sampai pada LCD grafis yang benar juga data yang dikirim benar tidak ada yang berubah atau hilang.

3.3 Perancangan Perangkat Keras

Perangkat keras adalah perangkat fisik yang berupa komponen elektronik yang mempunyai fungsi tertentu. Pada sistem ini perangkat keras dibagi menjadi 3 bagian, yaitu bagian sistem pengelola informasi atau *database*, bagian sistem jaringan dan komunikasi, dan bagian sistem elektronik untuk penampilan (*display*).

3.3.1 Rangkaian Arduino UNO Ke LCD TFT

Pada Gambar 3.3 dapat diketahui pin yang di hubungkan antara Arduino UNO dengan LCD TFT 2.4", dimana yang sebelah kiri dari panah (>) adalah pin dari arduino uno dan sebelah kanan adalah pin LCD TFT 2.4". Juga sudah disertai gambar Arduino UNO untuk mempermudah pembacaan,



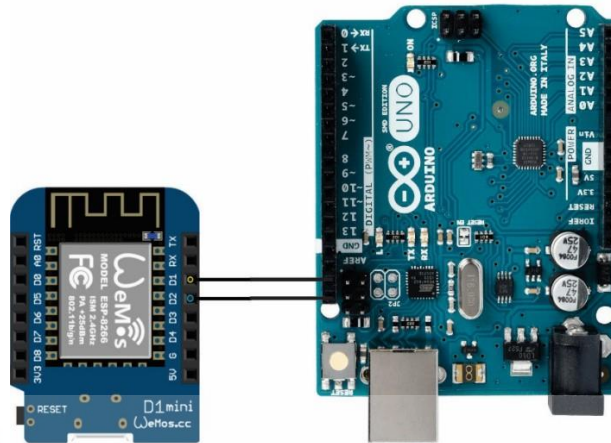
Gambar 3. 3 Rangkaian Arduino UNO Ke LCD TFT

3.3.2 Rangkaian Koneksi I2C Arduino UNO Ke Wemos

I2C menggunakan 2 *pin* yang digunakan untuk komunikasi data yaitu SCL dan SDA. Setiap *pin* mempunyai fungsi masing seperti SCL digunakan untuk sinkronisasi clock antara device dan SDA digunakan untuk jalur lintasan data.

Wemos mempunyai 9 digital *pin out* dimana yang akan digunakan adalah *pin* yang mendukung komunikasi I2C yaitu *pin* D1 dan D2. *Pin* D2 adalah sebagai *serial data* (SDA) ini digunakan untuk jalur data keluar masuk dari atau ke Wemos sedangkan *pin* D1 adalah sebagai *serial clock* (SCL) ini adalah sebagai jalur *clock* yang digunakan untuk pembacaan data.

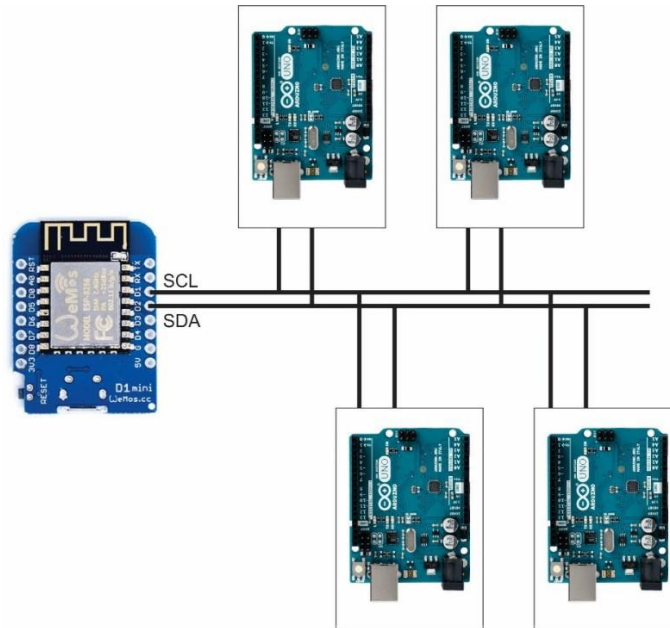
Arduino UNO mempunyai 32 *pin* I/O dimana antara 32 pin tersebut sudah ada jalur kusus yang digunakan untuk komunikasi I2C dan juga sudah dinamai dengan SDA dan SCL.



Gambar 3. 4 Rangkaian Koneksi I2C Arduino UNO Ke Wemos

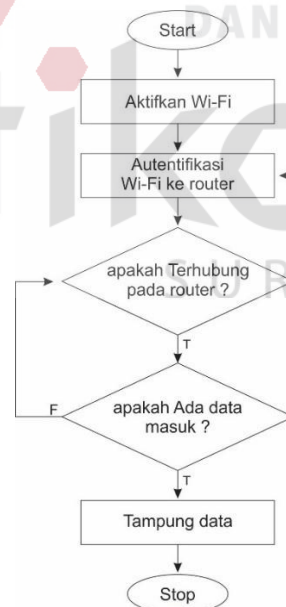
Pada Gambar 3.4 dapat diketahui bahwa *pin out* Wemos yang mendukung I2C berada di *pin* D1 dan D2. Pada Arduino sendiri pin yang mendukung I2C letaknya disendirikan dapat diperhatikan pada Gambar 3.5.

Jalur I2C dari Wemos lalu disambungkan secara paralel ke jalur I2C Arduino. Pada Gambar 3.4 penulis menggunakan 1 Arduino UNO yang akan digunakan sebagai contoh. Jalur I2C ini dapat dirangkai secara paralel perhatikan gambar 3.6 karena Wemos akan memberikan alamat pada setiap data yang dikirimkan. Apabila alamat yang dikirimkan Wemos dalam paket data cocok dengan alamat yang dimiliki I2C Arduino, maka data yang dikirimkan Wemos akan diterima dengan baik oleh Arduino.



Gambar 3. 5 Rangkaian Pararel I2C / BUS Koneksi

3.3.3 Algoritma Pengiriman Data Dari *Server* ke Wemos



Gambar 3. 6 Algoritma Pengiriman Data Dari *Server* Ke Wemos

Pada saat Wemos menerima data digunakan algoritma seperti Gambar 3.6 Awal mula *WiFi* yang dimiliki oleh Wemos dinyalakan, dimana default dari Wemos sendiri adalah *auto* aktif saat dinyalakan. Lalu Wemos akan melakukan *autentifikasi* dengan *router* yang SSID-nya dituju. Sebelum melakukan hal tersebut penulis memasukan *header* dan variabel yang digunakan untuk menyimpan *ssid* dan *password* untuk melakukan *autentifikasi*.

```
#include <ESP8266WiFi.h>
const char* ssid = "*****";
const char* password = "*****";
```

Setelah *header* dan *ssid* masuk, juga *password* sudah diisi, maka Wemos siap dihubungkan ke *router* dengan perintah *WiFi.begin(ssid,password)* Wemos akan memulai mengautentifikasi. Seperti pada *source code* berikut :

```
void setup()
{
  WiFi.begin(ssid, password);
}
```

Untuk memastikan apakah Wemos sudah terhubung atau belum, kita membutuhkan sinyal indikator untuk mengetahui status tersebut fungsi perintah yang penulis gunakan adalah dengan *WiFi.status() == WL_CONNECTED*, perintah tersebut adalah untuk melakukan pengecekan kondisi apakah *WiFi* sudah terhubung, bila status benar maka lampu indikator pada Wemos akan berkedip.

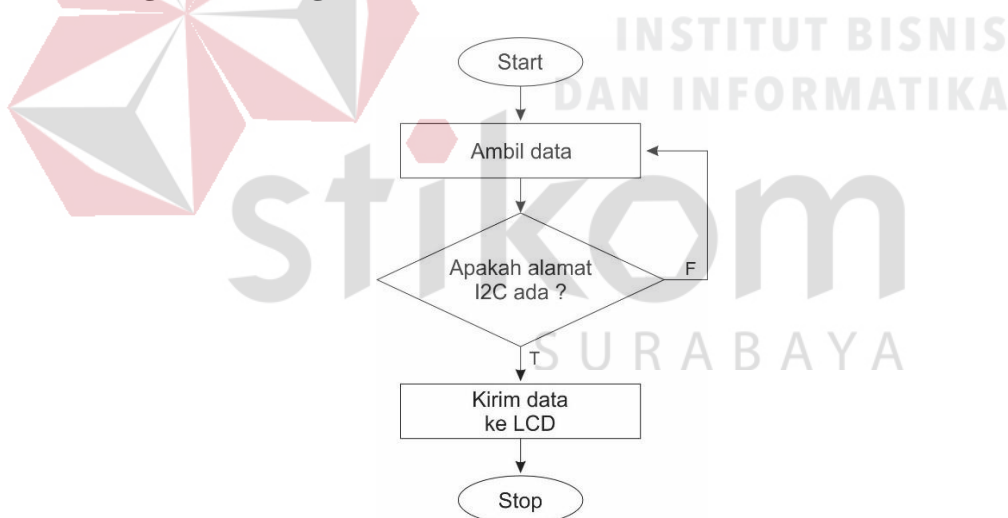
Perhatikan *source code* berikut :

```
IF (WiFi.status() == WL_CONNECTED)
{
  digitalWrite(LED_BUILTIN, LOW);
  delay(500);
  digitalWrite(LED_BUILTIN, HIGH);
  delay(500);
}
```

Untuk *source code* lengkapnya perhatikan di bawah ini :

```
#include <ESP8266WiFi.h>
const char* ssid = "*****";
const char* password = "*****";
void setup()
{
    WiFi.begin(ssid, password);
}
void loop()
{
    IF (WiFi.status() == WL_CONNECTED)
    {
        digitalWrite(LED_BUILTIN, LOW);
        delay(500);
        digitalWrite(LED_BUILTIN, HIGH);
        delay(500);
    }
}
```

3.3.4 Algoritma Pengiriman Dari Wemos Ke Arduino



Gambar 3. 7 Algoritma Pengiriman Data Dari Wemos Ke Arduino

Pada saat Wemos menampilkan data pada LCD TFT digunakan algoritma seperti Gambar 3.7. Awal mula pada program diberi *header* yang berfungsi untuk menjalankan I2C yang pertama adalah *wire.h* ini adalah *header* untuk membuka komunikasi I2C dan *header LiquidCristal_I2C* ini adalah *header* yang digunakan

untuk mengkonversi perintah *print* untuk LCD tanpa I2C menjadi data I2C untuk mengendalikan modul I2C 1602 . Berikut *source code* yang akan digunakan :

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

Setelah *header* dimasukkan langkah yang penulis ambil selanjutnya adalah instalasi variabel untuk I2C dalam instalasi variabel ini diikuti oleh alamat I2C, banyak karakter horisontal yang dimiliki LCD dan banyak karakter vertikal yang dimiliki LCD pada penelitian ini, penulis menggunakan LCD TFT. Dengan *source code* berikut :

```
LiquidCrystal_I2C LCD1(0x3E,16,2);
```

Variabel LCD1 untuk I2C tidak dapat digunakan sebelum diberikan perintah mulai langkah selanjutnya yang harus dilakukan adalah menjalankan perintah mulai untuk I2C tersebut, dalam penulisannya perintah mulai diikuti oleh alamat I2C, banyak karakter horisontal yang dimiliki LCD dan banyak karakter vertikal yang dimiliki LCD. Perhatikan *source code* di bawah :

```
LCD1.begin(0x3E,16,2);
```

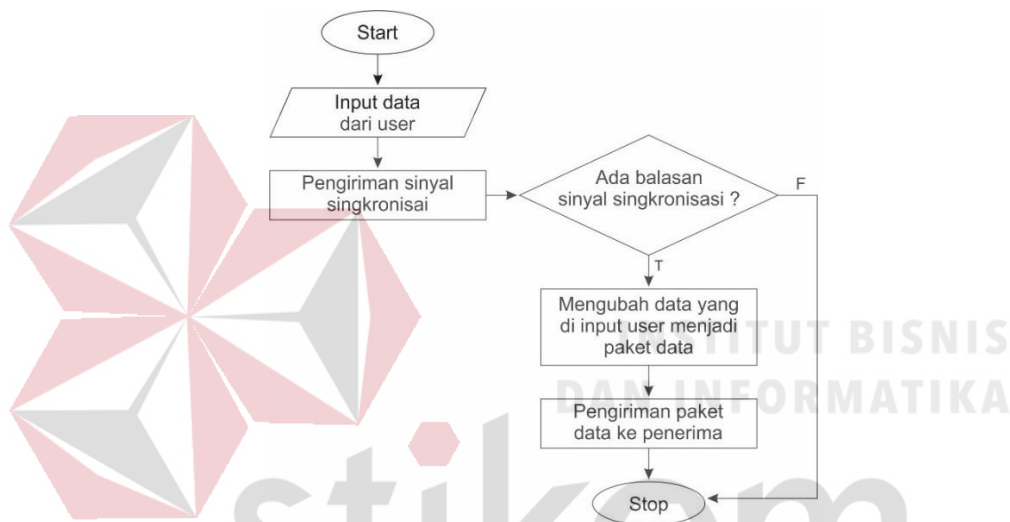
Setelah semua tahap dalam instalasi I2C LCD dilakukan berikutnya adalah mencoba komunikasi I2C dengan langsung mengirimkan data ke LCD, hal yang dapat ditampilkan dalam LCD adalah semua tipe data seperti *string*, *integer*, *double* / *float*, dan *char*. Perhatikan *source code* di bawah ini :

```
LCD1.print("LCD READY");
LCD1.print(d_barang);
LCD1.print(d_harga);
```

3.4 Perancangan Perangkat Lunak

Dari perancangan sistem tersebut, selain perencanaan *hardware* juga dibutuhkan perancangan perangkat lunak untuk menjalankan perancangan *hardware* yang telah dibuat, perangkat lunak ini berisi tentang informasi yang akan dikirimkan ke *hardware* melalui jaringan.

3.4.1 Algoritma Pengiriman Data



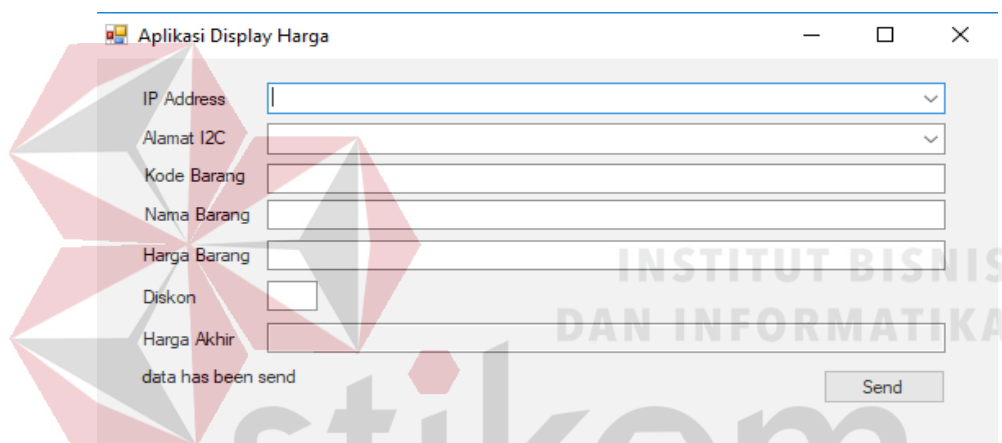
Gambar 3. 8 *Flowchart* Algoritma Pengiriman Data

Seperti yang dijelaskan oleh Gambar 3.8, data yang diinputkan *user* akan dibuatkan sebuah paket data dan dikirimkan ke penerima, namun sebelum mengirimkan *software* ini akan mengirimkan sinyal sinkronisasi yang fungsinya untuk memastikan bahwa penerima itu aktif. Jika penerima tidak dapat merespon sinyal sinkronisasi maka penerima di anggap tidak aktif dan data tidak dikirim.

Adapun informasi data yang akan dimuat dalam perangkat lunak atau aplikasi antara lain :

1. IP address tujuan.
2. I2C tujuan.
3. Nama Barang.
4. Harga barang.
5. Kode barang
6. Diskon.
7. Harga Akhir.

Tampilan perangkat lunak dapat dilihat pada Gambar di bawah ini :



Gambar 3. 9 Tampilan Perangkat Lunak

a. Sinkronisasi

Pada sinkronisasi penulis memanfaatkan fasilitas *ping* untuk mengetahui apakah alamat destinasi sedang aktif atau sedang dalam keadaan mati. Pada fungsi *ping software* akan mengirimkan sebuah paket ICMP ke alamat yang dituju, lalu jika alamat tujuan hidup maka *hardware* alamat tersebut akan mengirimkan sinyal *echo* atau sinyal *feedback*. Sinyal *feedback* ini yang akan menjadi acuan apakah proses dilanjutkan atau tidak.

Untuk menggunakan fasilitas *ping* awal mula penulis mendaftarkan *library network information* terlebih dahulu, ini digunakan untuk memunculkan fungsi *ping* dalam program.

```
using System.Net.NetworkInformation;
```

Untuk *ping* sendiri setelah *library* terdaftar penulis harus membuat variabel terlebih dahulu untuk menggantikan variabel *ping* disini penulis menggunakan variabel *isping* sebagai pemanggil, lalu pada baris kedua penulis membuat variabel baru yang diisi perintah untuk melakukan *ping* pada "*isping.send()*" hasil dari *ping* akan ditampung oleh variabel *reply*, perhatikan *source code* berikut:

```
Ping isping = new Ping();
PingReply reply = isping.Send(comboBox1.Text);
```

Pada saat ini hasil dari *ping* masih ditampung pada *reply* dan *reply* bukan *boolean* yang artinya datanya masih belum bisa dibaca. Untuk solusi saat ini, penulis menambahkan variabel untuk menyimpan hasil dari *ping*, dimana penulis membuat variabel baru berupa *pinging* yang digunakan untuk menampung status dari hasil *reply*. Perhatikan *source code* berikut:

```
bool pinging = false;
pinging = reply.Status == IPStatus.Success;
```

Berikut adalah *source code* lengkap pada bagian sinkronisasi:

```
Ping isping = new Ping();
bool pinging = false;

try
{
    PingReply reply = isping.Send(comboBox1.Text);
    pinging = reply.Status == IPStatus.Success;
}
catch
{
    pinging = false;
}
```

```

if ( pinging == false )
{
    MessageBox.Show("Check your IP address Destination..!!!", "Warning");
}

```

b. Pengiriman data

Pengiriman data yang dilakukan menggunakan protokol telnet, karena protokol telnet sendiri adalah protokol yang dibuat untuk keperluan komunikasi 2 arah antara *server* dan *client* dengan memanfaatkan *virtual terminal*. Dipilihnya telnet untuk protokol komunikasi yang digunakan adalah karena telnet support kepada wemos tidak seperti FTP yang harus melalui media parsing *web*. Pada pengiriman data berbasis text lebih cepat karena pengiriman bersifat langsung, tidak melalui media *web* jadi pengiriman lebih cepat dengan data yang kecil. Pengiriman data sendiri akan dikirimkan melalui media jaringan *Wi-Fi* yang menghubungkan antara Wemos dan *server*.

Untuk menggunakan fasilitas telnet awal mula penulis mendaftarkan *library net* dan *net.sockets* terlebih dahulu, *sockets* ini merupakan mekanisme komunikasi yang memungkinkan terjadinya pertukaran data antarprogram atau proses baik pada satu mesin atau antarmesin. Fasilitas *sockets* tersedia pada sistem operasi yang berkomunikasi dengan protokol TCP/IP. Seperti pada *source code* berikut :

```

Using System.Net;
using System.Net.Sockets;

```

Sedangkan untuk menggunakan fungsi telnet setelah mendaftarkan library berikutnya yang dilakukan penulis adalah pembuatan 2 variabel penting yaitu variabel *socket* dan *IpEndPoint*. Dimana *socket* berfungsi untuk membuat jalur dan membuka tutup akses, lalu *IpEndPoint* digunakan untuk pengalamatan jalur yang akan dibuka dan dikirimkan data oleh *socket*. Seperti pada *source code* berikut :

```
Socket sck = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
IPEndPoint localEndPoint = new IPEndPoint(IPAddress.Parse(combobox1.Text), int.Parse("23"));
```

Setelah variabel terbentuk, penulis menyiapkan data yang akan dikirimkan melalui program ini, format yang digunakan oleh pengirim dalam mengirimkan data adalah alamat_i2c, nama_barang, harga_barang. Setelah format data tersusun data akan di-convert atau di-encoding menjadi nilai ASCII karena pada protokol TCP/IP yang dikirimkan adalah data *byte*, ini juga dapat digunakan sebagai enkripsi data. Seperti pada *source code* berikut :

```
string nilai = combobox2.Text + "," + textBox4.Text + ",Rp " + textBox5.Text;
byte[] data = Encoding.ASCII.GetBytes(nilai);
```

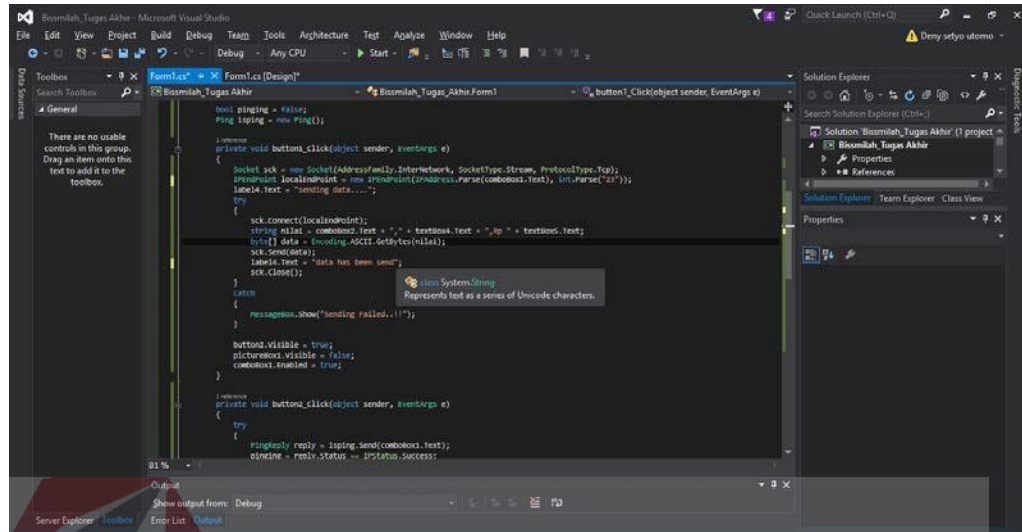
Setelah data terbentuk maka penulis siap untuk mengirimkan paket data tersebut, yang berperan di pengiriman data ini adalah *socket* yang akan membuka dan mengirimkan data lalu menutup jalur komunikasi. Perhatikan *source code* berikut :

```
sck.Connect(localEndPoint);
sck.Send(data);
sck.Close();
```

Berikut adalah *source code* lengkap pada bagian pengiriman data :

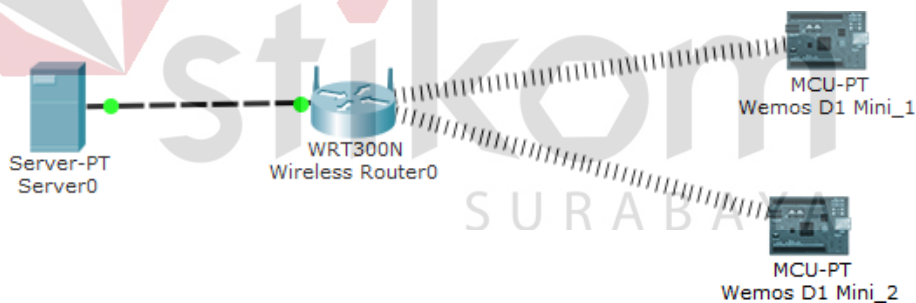
```
Socket sck = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
IPEndPoint localEndPoint = new IPEndPoint(IPAddress.Parse(combobox1.Text), int.Parse("23"));
label4.Text = "sending data...";
try
{
    sck.Connect(localEndPoint);
    string nilai = combobox2.Text + "," + textBox4.Text + ",Rp " + textBox5.Text;
    byte[] data = Encoding.ASCII.GetBytes(nilai);
    sck.Send(data);
    label4.Text = "data has been send";
    sck.Close();
}
catch
{
    MessageBox.Show("Sending Failed..!!");
}
```

Hasil dari penulisan source code dapat dilihat pada gambar 3.10 source code ditulis pada *software* Visual Studio.



Gambar 3. 10 Tampilan Source Code Pada Visual Studio

3.5 Perancangan Topologi Jaringan



Gambar 3. 11 Topologi Jaringan Sistem

Topologi jaringan yang akan digunakan dalam membuat sistem ini adalah seperti Gambar 3.11. Menggunakan 1 *Router Access Point* utama sebagai pengatur lalu lintas data dan penyebar sinyal WLAN untuk *client* koneksi melalui WLAN.

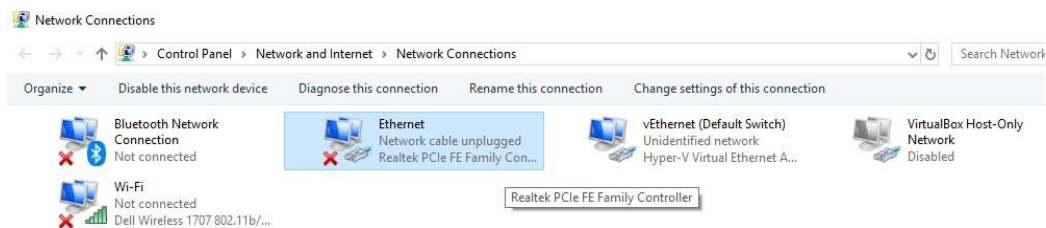
Model pengalamatan IP yang akan digunakan adalah antara *router* dan Komputer menggunakan alamat jaringan 192.168.100.0 dengan *subnet mask*

255.255.255.252 / 30. Pada sisi jaringan *server* hanya akan disediakan IP 2 buah yaitu untuk *router* dan Komputer saja. Ini digunakan untuk mengamankan jaringan *server* agar bila ada yang ingin terhubung ke jaringan *server* tidak bisa. Pada sisi *client* menggunakan 1 alamat jaringan yaitu 192.168.0.0 dan *subnet mask*-nya adalah 255.255.192.0 / 18. dengan subnet tersebut *router* akan menyediakan alamat IP Untuk *client* sebanyak 16.382 *hosts* mulai dari 192.168.1.2 sampai dengan 192.168.64.254. Ini digunakan untuk mempermudah komunikasi antar Wemos.

Untuk mewujudkan pembagian IP seperti diatas maka dibutuhkan tambahan *Access Point* untuk memperluas jangkauan WLAN. *Access Point* ini berfungsi untuk membagi data dan sinyal dari 1 *node* ke banyak *node*. Dengan *Access Point* ini 1 *node* dari *router* akan disebar ke *Access Point* yang lain. *Access Point* ini digunakan untuk menghubungkan koneksi antara Wemos ke jaringan. Setiap Wemos yang akan terhubung ke jaringan harus melewati *Access Point* yang SSID-nya sama dengan IP yang akan ditentukan secara *static*.

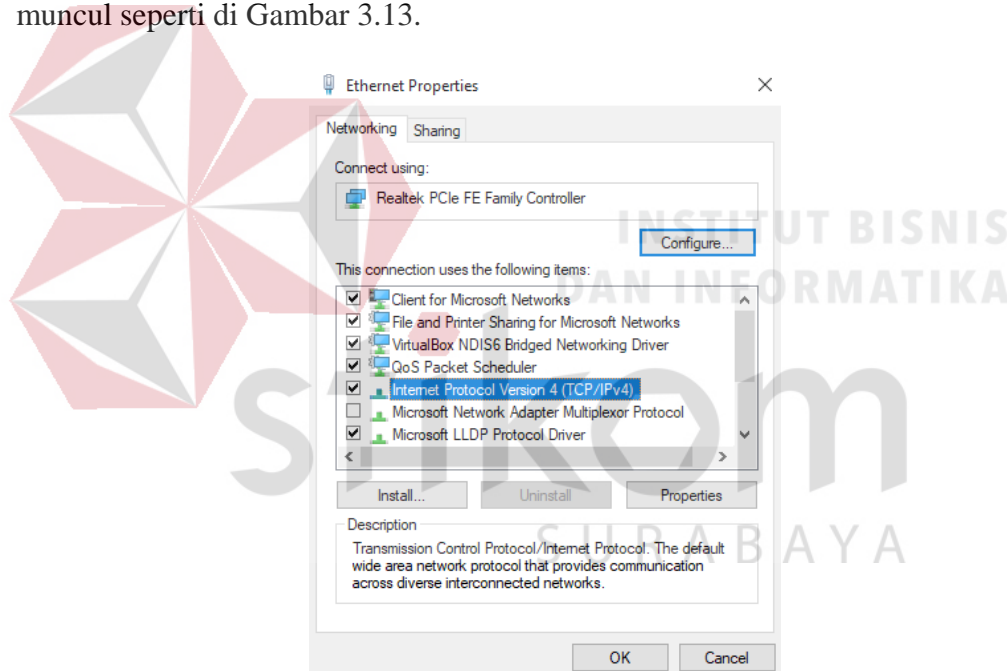
3.5.1 Konfigurasi IP Server

Untuk mewujudkan jaringan seperti Gambar 3.11 dibutuhkan konfigurasi pada bagian *server* terutama pada bagian IP Address dan *subnet mask*. Dalam implementasinya *server* diberi IP 192.168.100.1 dengan *subnet mask* 255.255.255.252 / 30. Untuk melakukan pengaturan IP tersebut menggunakan windows adalah dengan cara masuk ke *control panel* komputer lalu masuk ke bagian *network and internet* pilih bagian *network connections*. Maka akan muncul Gambar seperti Gambar 3.12.



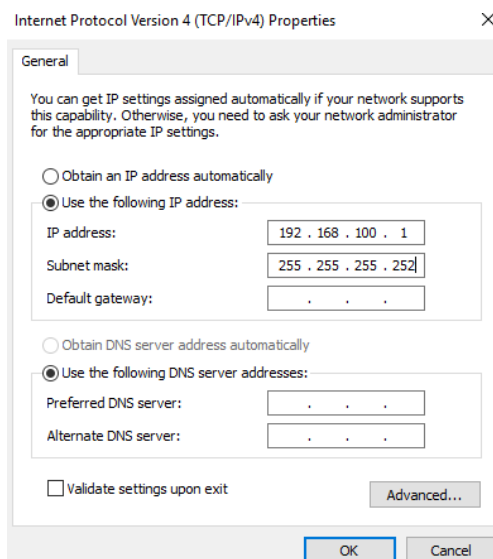
Gambar 3. 12 Tampilan *Network connections*

Setelah muncul seperti Gambar 3.12 pilih ethernet yang tersambung ke router langsung. Klik kanan pada *eternet* yang terpilih lalu klik *properties*. Akan muncul seperti di Gambar 3.13.



Gambar 3. 13 Tampilan *Ethernet Properties*

Setelah muncul seperti di Gambar 3.13 klik 2 kali pada *Internet Protocol Version 4 (Ipv4)* atau klik 1 kali lalu klik button *properties* akan muncul seperti Gambar 3.14.



Gambar 3. 14 Tampilan Ipv4 Properties

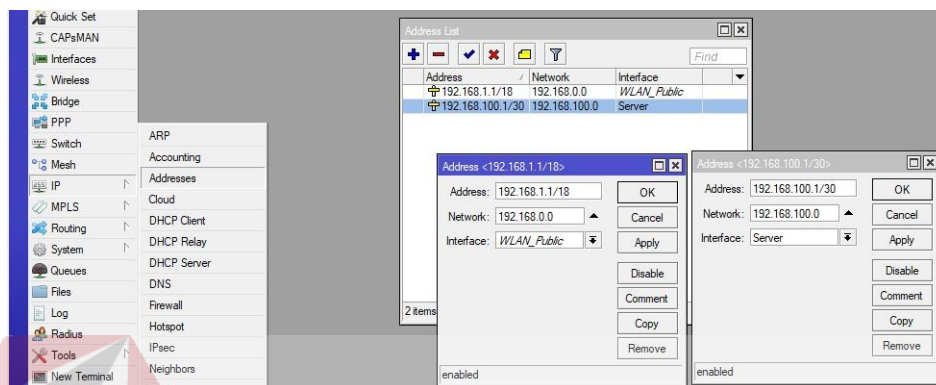
Pada Gambar 3.14 akan dilakukan setting IP Address pada topologi ini PC digunakan sebagai *server* maka yang akan di isi adalah pada bagian IP Address dan *Subnet mask*. IP diisi dengan 192.168.100.2 dan subnet mask adalah 255.255.255.252 / 30.

Setelah langkah tersebut selesai maka IP komputer kita adalah 192.168.100.2 langkah selanjutnya klik oke lalu tutup *window*.

3.5.2 Konfigurasi *Routing* dan IP Router

Untuk menghubungkan antara jaringan Wemos dan jaringan server diperlukan sebuah *router* agar kedua jaringan tersebut dapat di routing. Pada penelitian ini *router* yang digunakan oleh penulis adalah *router* Mikrotik. Untuk mewujudkan topologi jaringan seperti pada Gambar 3.15 *router mikrotik* harus di setting agar dapat membuat jaringan dan menghubungkan jaringan.

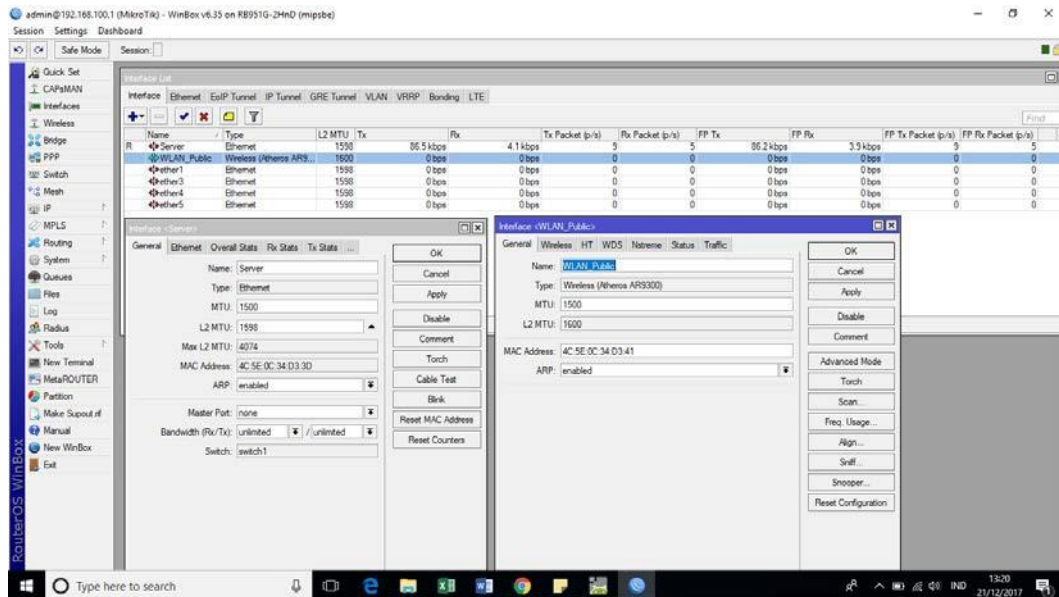
Adapun langkah awal yang harus dilakukan adalah melakukan setting IP mikrotik pada Gambar 3.15 jaringan dibagi menjadi 2 yaitu jaringan LAN untuk server yang memiliki alamat jaringan 192.168.100.0 dan jaringan WLAN untuk Wemos yang memiliki alamat jaringan 192.168.0.0.



Gambar 3. 15 Setting IP LAN Dan WLAN Mikrotik

Pada Gambar 3.15 dapat dilihat bahwa LAN disetting dengan IP 192.168.100.1/30 dan WLAN di setting IP 192.168.1.1/18. Artinya LAN hanya akan menyediakan 2 host 1 digunakan untuk gateway dan satunya lagi digunakan untuk komputer. Sedang pada WLAN menyediakan 16.382 host.

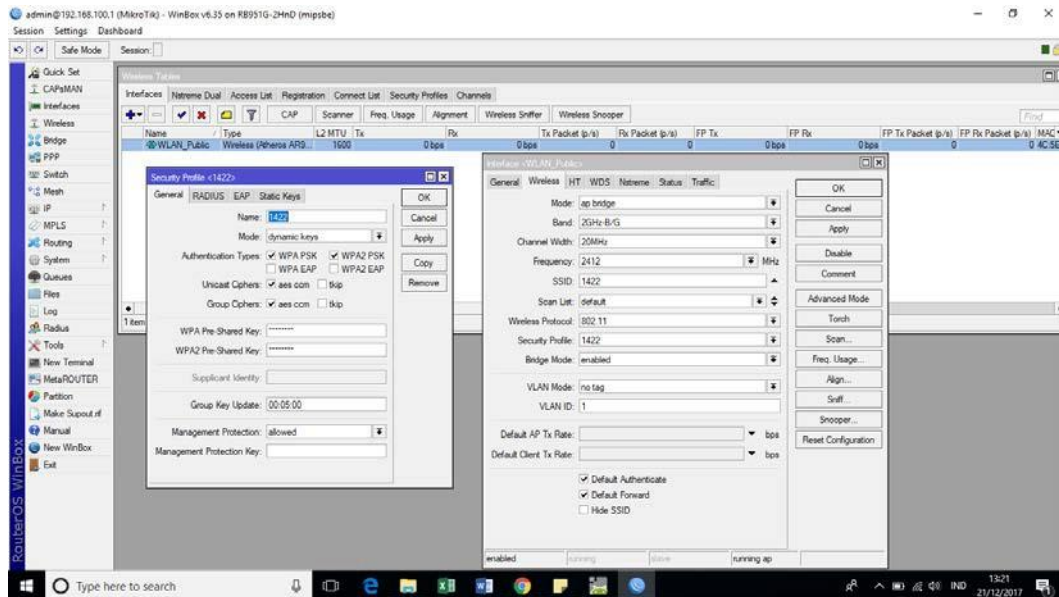
Setelah IP di setting selanjutnya adalah melakukan konfigurasi interface dimana konfigurasi ini berguna untuk membedakan nama interface agar lebih mudah dipahami. Juga untuk menyalakan *Wireless* Mikrotik karena settingan default adalah *Wireless* dalam keadaan mati.



Gambar 3. 16 Konfigurasi *Interface Router* Mikrotik

Pada Gambar 3.16 dapat dilihat bahwa interface LAN diberi nama *server* dan pada jaringan WLAN diberi nama *WLAN_Public*. Lalu untuk mengaktifkan *Wireless* Mikrotik cukup mengklik interface *WLAN_Public* lalu klik *icon* centang maka *Wireless mikrotik* sudah aktif.

Setelah *WLAN* aktif maka selanjutnya adalah melakukan setting konfigurasi *WLAN* agar dapat digunakan sebagai *access point* juga melakukan konfigurasi keamanan *WLAN* agar tidak mudah dimasuki orang dengan cara memberi keamanan pada jaringan *WLAN*.

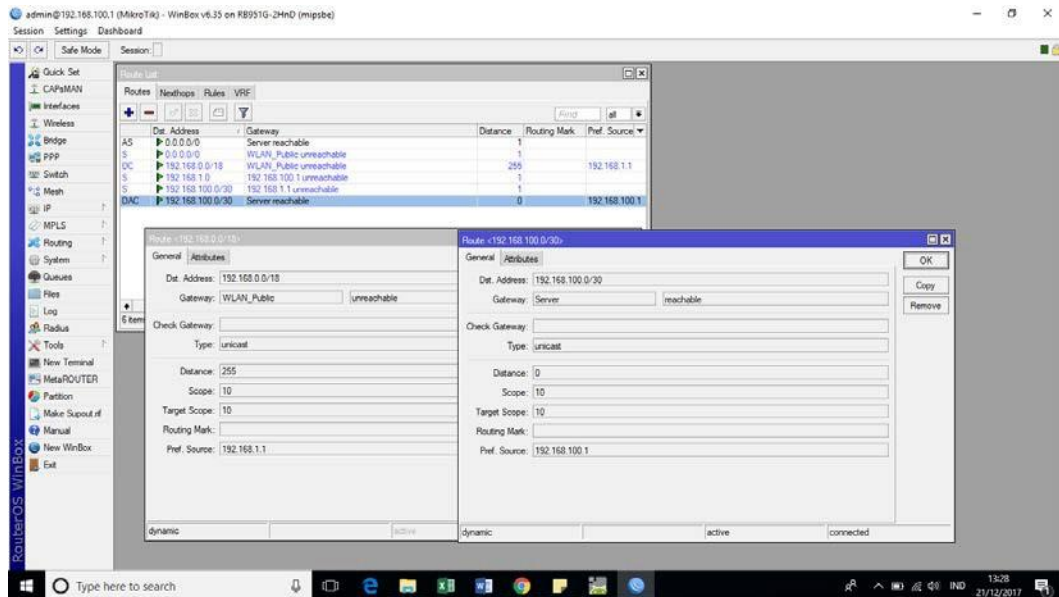


Gambar 3. 17 Konfigurasi WLAN Dan Keamanan WLAN

Pada Gambar 3.17 dapat dilihat bahwa pengaturan WLAN dibuat mode ap bridge agar dapat menjadi *access point* dan nama SSID yang dibuat adalah 1422, artinya nantinya *client* akan mencari *access point* dengan nama 1422 untuk masuk ke jaringan WLAN.

Dari Gambar 3.17 pula dapat dilihat bahwa pengamanan *WiFi* menggunakan WPA PSK dan WPA2 PSK dan diberi password, artinya *client* yang ingin memasuki *access point* ini diharuskan mengisi password sebelum masuk. Ini berguna untuk pengamanan agar yang tidak mempunyai kepentingan tidak dapat masuk kedalam jaringan.

Setelah WLAN dan LAN aktif ini tidak dapat langsung berkomunikasi karena berada di jaringan berbeda. Adapun hal yang harus dilakukan selanjutnya adalah melakukan routing antara jaringan WLAN dan Jaringan LAN.



Gambar 3. 18 Setting *Routing* Mikrotik

Pada Gambar 3.18 dapat dilihat pengaturan routing yaitu dengan mengisikan destinasi *address* adalah alamat jaringan yang dituju dan gateway adalah alamat yang akan kita lewati menuju ke jaringan yang kita tuju. Untuk routing *server* adalah destinasi *address* menuju WLAN_Public dan untuk WLAN_Public adalah destinasi *address* menuju *server*. Kedua jaringan ini harus saling di routing agar dapat saling berkomunikasi.

3.5.3 Konfigurasi IP *Static* pada Wemos D1 mini

Pengalaman IP pada Wemos murni menggunakan *source code* untuk konfigurasinya yang harus di tentukan pertama adalah alamat IP Address local *Wemos*, IP subnet mask, dan IP gateway. Pada penelitian ini penulis menggunakan alamat jaringan adalah 192.168.0.0 dan IP gateway 192.168.1.1 lalu menyediakan host sebanyak 16.382 artinya bahwa IP yang disediakan untuk *Wemos* adalah sebanyak 16.379. Hasil tersebut adalah setelah host dikurangi 3 karena untuk alamat

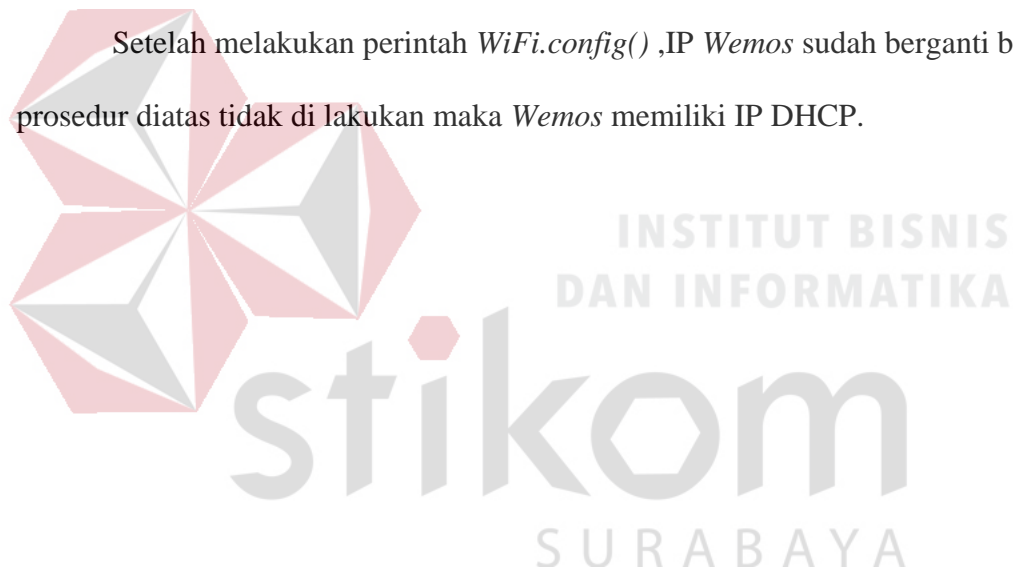
jaringan, alamat broadcast, dan alamat gateway. Untuk range alamat IP yang dapat digunakan adalah antara 192.168.1.2 – 192.168.64.254. perhatikan *source code* berikut :

```
IPAddress localIP(192.168.1.100);  
IPAddress subnet(255, 255, 192, 0);  
IPAddress gateway(192.168.1, 1);
```

Setelah menentukan IP *Wemos* yang selanjutnya dilakukan adalah mengubah konfigurasi yang ada di *Wemos*. Karena dengan hanya mengisi IP tidak dapat mengubah IP *Wemos* harus ada yang mengeksekusi perubahan.

```
WiFi.config(localIP, gateway, subnet);
```

Setelah melakukan perintah *WiFi.config()* ,IP *Wemos* sudah berganti bila prosedur diatas tidak di lakukan maka *Wemos* memiliki IP DHCP.



BAB IV

HASIL PENGUJIAN DAN PENGAMATAN

Dalam bab ini penulis akan menguraikan dan menjelaskan beberapa hasil pengujian dan hasil penelitian Tugas Akhir ini. Pengujian yang dilakukan meliputi pengujian perangkat lunak dan kinerja keseluruhan sistem, serta analisa pengiriman data dari PC ke Wemos lalu dari Wemos ke LCD. Perlengkapan alat yang digunakan dalam pengujian ini dapat dilihat di Gambar 4.1.



Gambar 4. 1 Tampilan Peralatan Yang Digunakan Untuk Pengujian

4.1 Pengujian Wemos

Dalam pengujian Wemos dilakukan dengan menuliskan program pada *sketch* Arduino IDE lalu di *upload* pada *Wemos*.

4.1.1 Tujuan Pengujian Wemos

Pengujian ini dilakukan untuk memastikan Wemos masih berfungsi dengan baik dan tidak dalam keadaan rusak.

4.1.2 Alat Yang Digunakan Pengujian Wemos

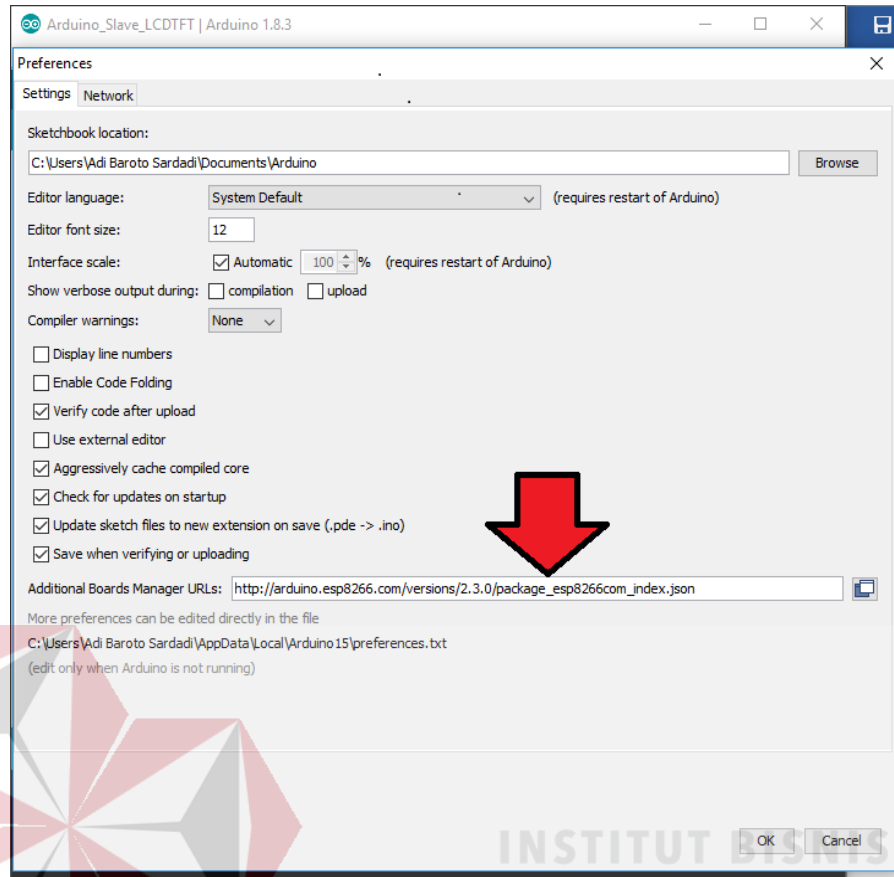
Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

- a. *Power supply 5V*
- b. Kabel data USB *mikro*
- c. Wemos D1 Mini
- d. Komputer / Laptop
- e. *Software* Arduino IDE

4.1.3 Prosedur Pengujian Wemos

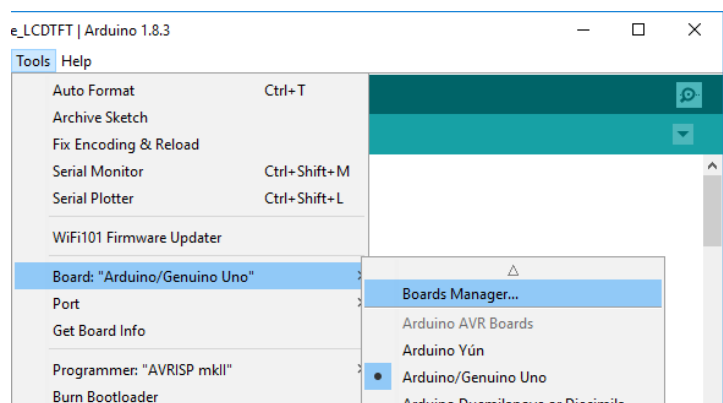
Prosedur pengujian :

1. Menyalakan komputer / laptop yang akan digunakan untuk menguji Wemos.
2. Membuka *software* Arduino IDE.
3. Melakukan instalasi *library* Wemos pada Arduino IDE.
 - a. Untuk mengatur *download board* Wemos D1 Mini masuk pada menu file lalu memilih *preferences*, masukan alamat url berikut http://arduino.esp8266.com/versions/2.3.0/package_esp8266com_index.js kedalam kolom Additional *Board Manager* Url lalu memilih OK, sebagai contoh Gambar 4.2.



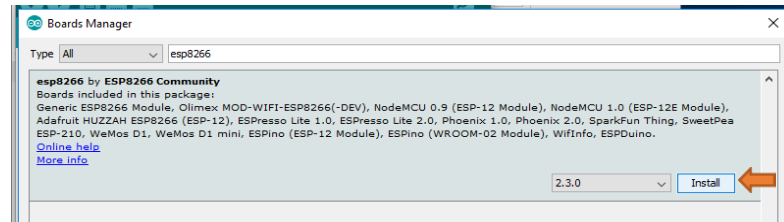
Gambar 4.2 File Preferences

- b. Setelah selesai masukan URL kemudian membuka *tools* memilih *board* lalu memilih *board manager*. Seperti Gambar 4.3.



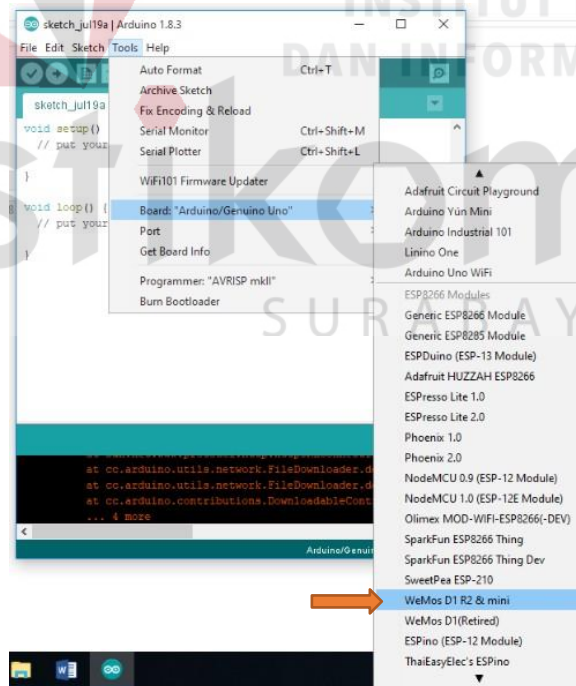
Gambar 4.3 Board Manager

- c. Setelah muncul window *board* manager menuliskan pada kolom “esp8266” lalu menekan enter, setelah pencarian selesai memilih tombol *install*, lalu menunggu hingga proses *download* dan *install* selesai. Seperti Gambar 4.4.



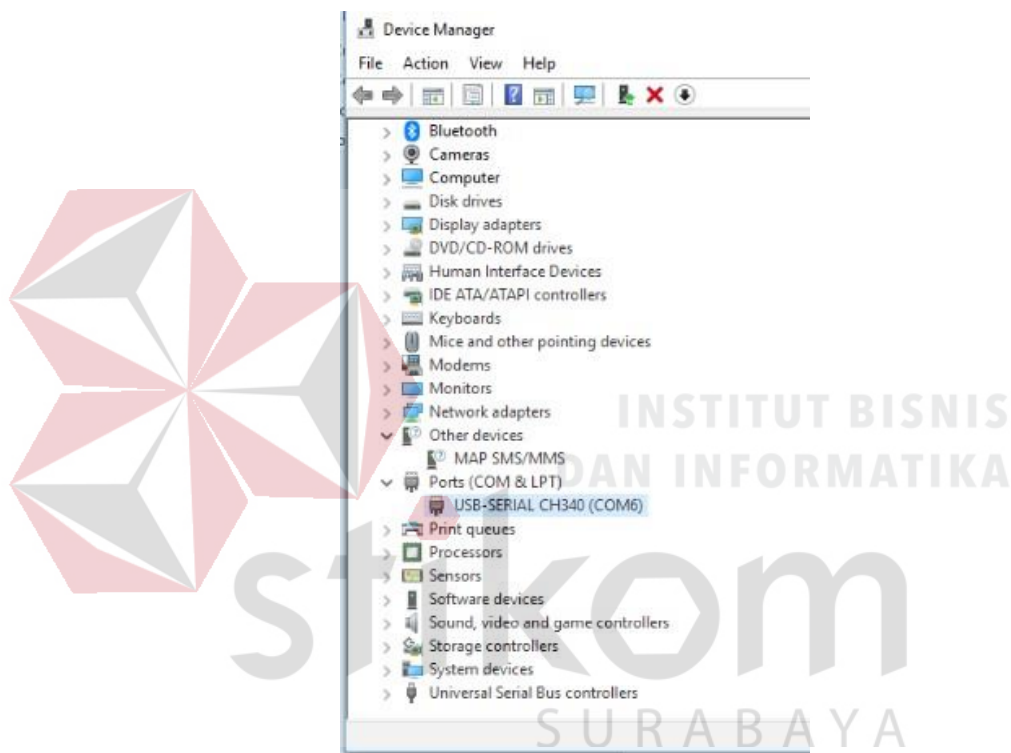
Gambar 4. 4 Pencari Dan *Install Board* Wemos.

- d. Setelah instalasi selesai kemudian mengecek pada menu *tools* memilih *board* lalu mencari Wemos D1 Mini, apabila ketemu Arduino IDE sudah *support* Wemos dan memiliki *library* pada *board*. Seperti pada Gambar 4.5.



Gambar 4. 5 Pengecekan *Board* Wemos D1 Mini Sudah Terinstall

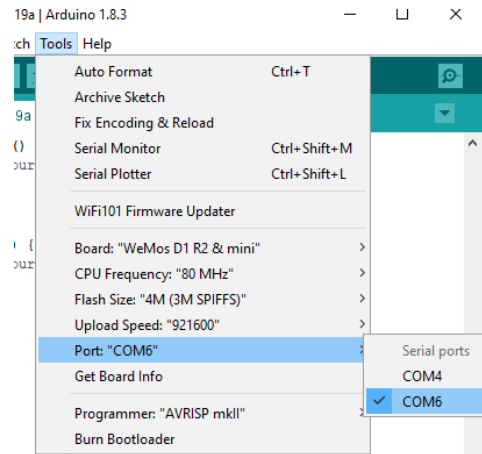
4. Menancapkan Kabel data USB *mikro* pada Wemos D1 Mini dan sisi USB dihubungkan kepada USB laptop.
5. Menyalin *Source code* dengan judul “pengujian Wemos menggunakan led blink” pada lampiran.
6. Membuka *device manager* pada laptop, melihat COM *port* Wemos berada di COM nomor berapa. Seperti pada Gambar 4.6.



Gambar 4. 6 Tampilan *Device Manager*

Dari Gambar 4.6 dapat di lihat bahwa Wemos berada di COM 6 pada laptop penulis.

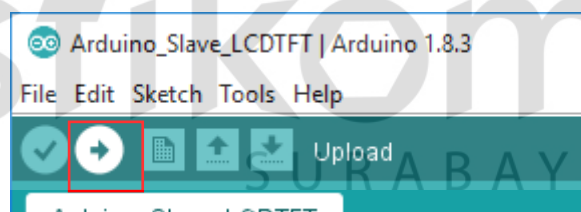
7. Menyesuaikan *port* Wemos pada Arduino IDE dengan *device manager*. Seperti pada Gambar 4.7.



Gambar 4. 7 Setting *Port* Wemos

Dari Gambar 4.7 dapat dilihat bahwa pada Arduino IDE sudah di setting *port* komunikasi Wemos berada di COM6.

8. Meng-*Upload* program yang telah dibuat menuju ke Wemos dengan mengklik tombol *upload* yang bergambar panah ke kanan di bawah menu. Seperti pada Gambar 4.8.



Gambar 4. 8 *Icon Upload*

Pada Gambar 4.8 dapat dilihat bahwa setelah *icon upload* dipilih maka Arduino IDE melakukan proses *upload* program menuju ke Wemos melewati *port* komunikasi.

```

Uploading...
Archiving built core (caching) in: C:\Users\DENYSE-1\AppData\Local\Temp\arduino_cache_353959\core\core_esp8266_esp8266_d1_mini
Sketch uses 222209 bytes (21%) of program storage space. Maximum is 1044464 bytes.
Global variables use 31576 bytes (38%) of dynamic memory, leaving 50344 bytes for local variables. Maximum is 81920 bytes.
Uploading 226352 bytes from C:\Users\DENYSE-1\AppData\Local\Temp\arduino_build_73978/Blink.ino.bin to flash at 0x00000000
..... [ 36% ]
..... [ 72% ]
.....

```

Gambar 4. 9 Proses *Upload* Program Ke Wemos

4.1.4 Hasil Pengujian Wemos

Setelah melakukan pengujian kepada Wemos di dapatkan hasil seperti pada Gambar 4.9 dan Gambar 4.10.

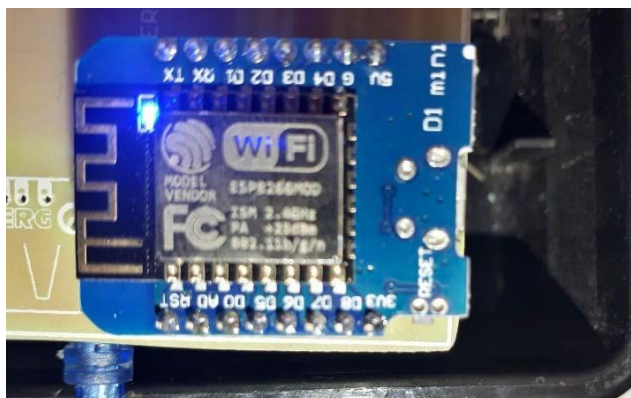
```

Done uploading.
Archiving built core (caching) in: C:\Users\DENYSE-1\AppData\Local\Temp\arduino_cache_353959
Sketch uses 222209 bytes (21%) of program storage space. Maximum is 1044464 bytes.
Global variables use 31576 bytes (38%) of dynamic memory, leaving 50344 bytes for local va
Uploading 226352 bytes from C:\Users\DENYSE-1\AppData\Local\Temp\arduino_build_73978/Blink
..... [ 36% ]
..... [ 72% ]
..... [ 100% ]

```

Gambar 4. 10 *Upload* Program Ke Wemos Sukses

Pada Gambar 4.10 menunjukan bahwa program yang di *upload* menuju Wemos berhasil sampai 100%, hal ini menunjukan Wemos dalam keadaan sudah dapat dimasuki program.



Gambar 4. 11 Hasil Dari *Upload* Program

Pada Gambar 4.11 menunjukkan bahwa program yang telah dibuat dapat berjalan sesuai yang diinginkan, ditunjukkan dengan menyalnya lampu led yang berada di Wemos. Ini menunjukkan bahwa Wemos dapat berjalan sesuai dengan mestinya, juga dapat disimpulkan bahwa Wemos dalam keadaan baik dan siap untuk digunakan.

4.2 **Pengujian *Scaning Access Point* Oleh Wemos**

Dalam pengujian *scaning access point* oleh Wemos dilakukan dengan menuliskan program pada *sketch* Arduino IDE yang memerintahkan Wemos untuk mencari *access point* yang berada di radius jangkauan *WiFi* Wemos.

4.2.1 **Tujuan Pengujian *Scaning Access Point* Oleh Wemos**

Pengujian ini dilakukan untuk memastikan Wemos dapat mencari *access point* yang dituju dan memastikan keadaan *WiFi* Wemos dalam keadaan baik.

4.2.2 **Alat Yang Digunakan Pengujian *Scaning Access Point* Oleh Wemos**

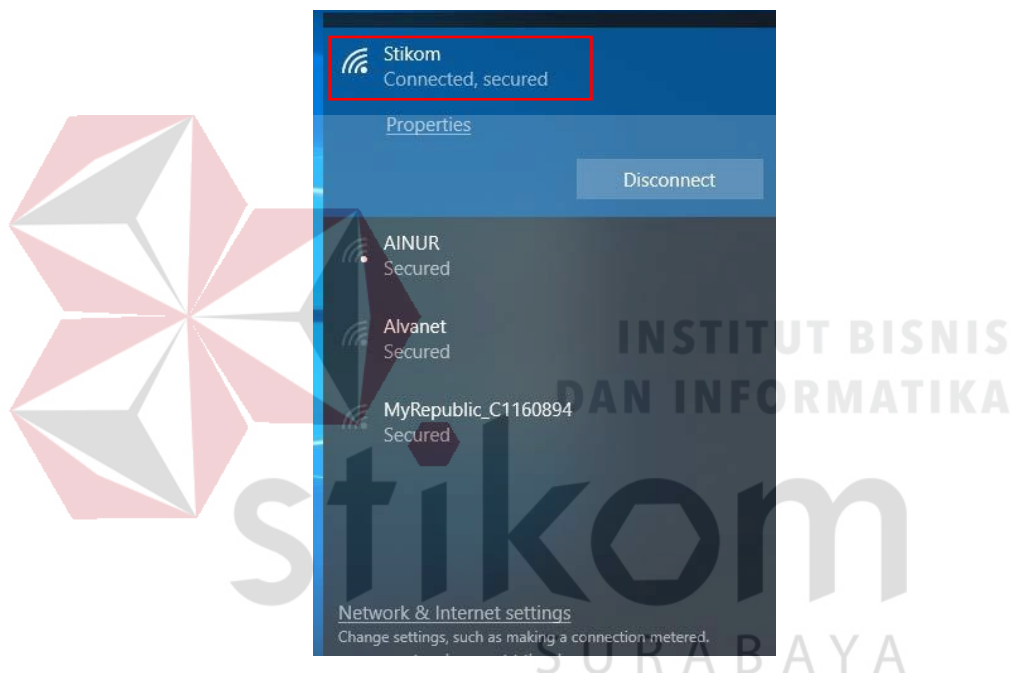
Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

- a. *Power supply* 5V
- b. *Akses Point*
- c. *Software* Arduino IDE
- d. Laptop / Komputer
- e. Kabel data *mikro* USB
- f. Wemos D1 Mini

4.2.3 Prosedur Pengujian *Scanning Access Point* Oleh Wmos

Prosedur pengujian :

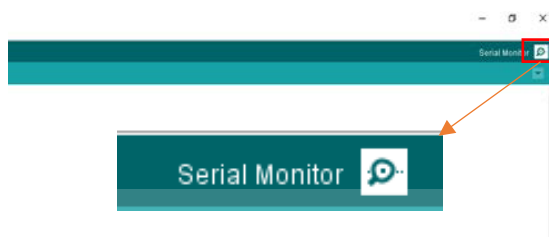
1. Menyalakan *access point*.
2. mengaktifkan laptop dan menyalakan *WiFi* pada laptop.
3. Membuka *WiFi* dan memastikan jaringan *Wireless* yang disebar oleh *access point* terlihat oleh laptop sebelum melakukan pengujian seperti pada Gambar 4.12.



Gambar 4. 12 Tersedia Jaringan WLAN

4. Menancapkan Kabel data USB *mikro* pada Wemos D1 Mini dan sisi USB dihubungkan kepada USB laptop.
5. Membuka *software* Arduino IDE.
6. Menyalin *Source code* dengan judul “Pengujian *scanning Access Point* Wemos” pada lampiran.

7. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.
8. Meng-*upload* program yang telah dibuat menuju ke Wemos.
9. Setelah meng-*upload* selesai membuka *serial monitor* yang berada disebelah kanan pada Arduino IDE yang memiliki *icon* seperti kaca pembesar seperti pada Gambar 4.13.



Gambar 4. 13 *Icon Serial Monitor*

4.2.4 Hasil Pengujian *Scanning Access Point* Oleh Wemos

Setelah melakukan pengujian *scanning Access Point* Wemos maka didapatkan hasil pada *serial monitor* seperti pada Gambar 4.14.

```

COM6
scan start
scan done
2 networks found
1: Stikom (-54)*
2: WARKOP KERAMAT (-63)*

```

Gambar 4. 14 Hasil *Scanning Access Point* Oleh Wemos

Pada Gambar 4.14 menunjukkan hasil dari pantauan *serial monitor* Wemos melalui *port* COM6 , dari gambar tersebut membuktikan Wemos dapat menemukan *access point* yang telah disebar oleh *access point* dan dari hasil ini juga dapat disimpulkan bahwa kondisi *WiFi* yang dimiliki Wemos bekerja dengan baik.

4.3 Pengujian Menghubungkan *WiFi Wemos* Pada *Access Point*

Dalam pengujian menghubungkan *WiFi Wemos* pada *access point* ini dilakukan dengan menuliskan program pada *sketch* Arduino IDE yang memerintahkan Wemos untuk menghubungkan diri kepada *access point* yang telah ditentukan.

4.3.1 Tujuan Pengujian Menghubungkan *WiFi Wemos* Pada *Access Point*

Pengujian ini dilakukan untuk memastikan Wemos dapat menghubungkan diri kepada *access point* yang ditentukan serta dapat berkomunikasi di jaringan yang dibuat oleh *access point* tersebut.

4.3.2 Alat Yang Digunakan Pengujian Menghubungkan *WiFi Wemos* Pada *Access Point*

Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

- a. *Power supply* 5V
- b. *Akses Point*
- c. *Software* Arduino IDE
- d. Laptop / Komputer
- e. Kabel data *mikro* USB
- f. Wemos D1 Mini

4.3.3 Prosedur Pengujian Menghubungkan *WiFi* Wemos Pada *Access Point*

Prosedur pengujian :

1. Menyalakan *access point*.
2. Mengaktifkan laptop dan menyalakan *WiFi* pada laptop.
3. Menghubungkan laptop pada *access point* yang sudah disediakan.
4. Membuka *command prompt*.
5. Menuliskan “ipconfig” lalu menekan enter pada *command prompt*.
6. Melihat IP yang didapatkan oleh laptop dengan memperhatikan adapter *WiFi* terdapat *Ipv4 Address* yang menjadi alamat IP komputer yang diberikan oleh *access point*. Seperti pada Gambar 4.15.

```

Command Prompt
C:\Users\Adi Baroto Sardadi>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 12:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix . :
    Link-local IPv6 Address . . . . . : fe80::219c:8415:1dbd:ae68%18
    IPv4 Address. . . . . : 192.168.43.210
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.43.1

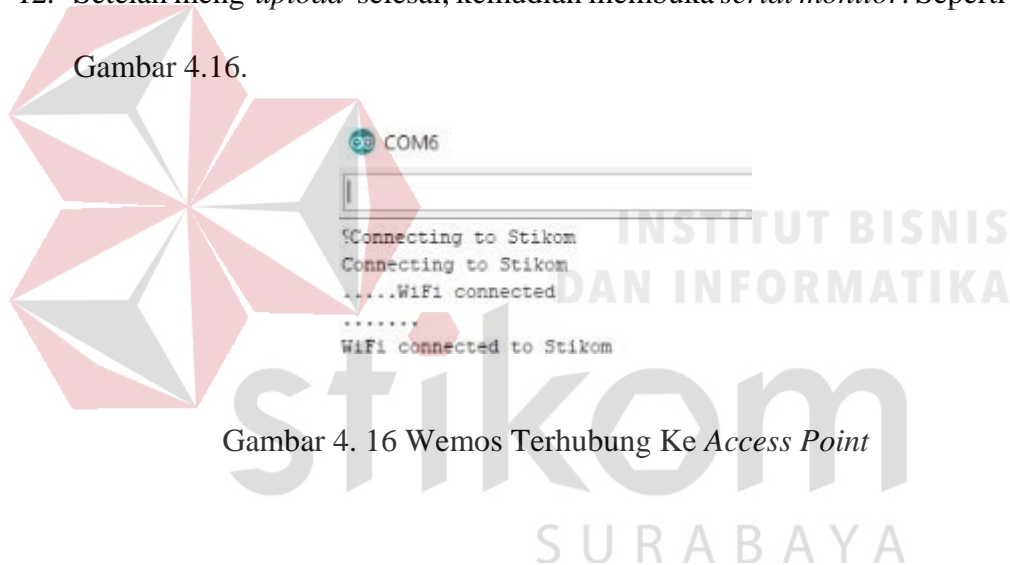
Ethernet adapter Bluetooth Network Connection:
  
```

Gambar 4. 15 Alamat IP Yang Didapat Dari *Access Point*

Dari Gambar 4.15 dapat dilihat bahwa setelah menuliskan “ipconfig” lalu menekan enter akan muncul *IP configuration*. Lalu dari gambar tersebut juga dapat dilihat bahwa laptop mendapatkan IP 192.168.43.210.

7. Menancapkan Kabel data USB *mikro* pada Wemos D1 Mini dan sisi USB dihubungkan kepada USB laptop.
8. Membuka *software* Arduino IDE.
9. Menyalin *Source code* dengan judul “Pengujian *connecting WiFi to access point*” pada lampiran.
10. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.
11. Meng-*Upload* program yang telah dibuat menuju ke Wemos.
12. Setelah meng-*upload* selesai, kemudian membuka *serial monitor*. Seperti pada

Gambar 4.16.



Gambar 4. 16 Wemos Terhubung Ke *Access Point*

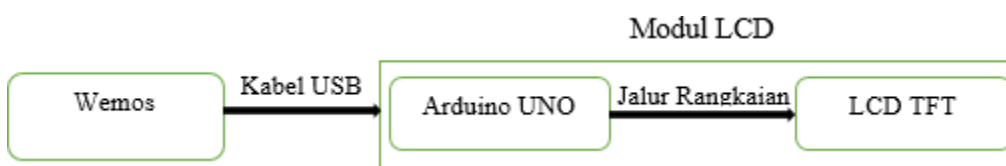
Dilihat dari Gambar 4.16 menunjukkan Wemos terhubung pada *access point* .

4.3.4 Hasil Pengujian Menghubungkan *WiFi* Wemos Pada *Access Point*

Dilihat dari Gambar 4.16 dapat disimpulkan bahwa Wemos dapat terhubung ke jaringan WLAN yang di buat oleh *access point* dan mendapatkan *IP address* dari *access point*, ini menunjukkan *WiFi* Wemos dalam keadaan baik dan dapat digunakan untuk komunikasi.

Bila dilihat dari Gambar 4.16 dapat di simpulkan juga bahwa Wemos dapat berkomunikasi dengan laptop melauai jaringan WLAN, ini dikarenakan Wemos memberi balasan kepada laptop saat laptop megirimkan sinyal ping, hal ini menunjukkan bahwa Wemos dapat berkomunikasi melalui *Wireless*.

4.4 Pengujian *Scaning* Dan Pengiriman I2C Wemos Menuju LCD



Gambar 4. 17 Diagram Blok Pengujian *Scaning* Dan Pengiriman Data Dari Wemos Menuju LCD

Pada Gambar 4.17 dapat diketahui bahwa Wemos akan mengirimkan data menuju ke Arduino lalu oleh Arduino akan menerjemahkan data I2C menjadi tampilan di LCD TFT.

Dalam pengujian ini dilakukan dengan menuliskan program pada *sketch* Arduino IDE yang memerintahkan Wemos mengirimkan data ke LCD menggunakan protokol I2C. Data yang dikirim oleh Wemos akan di tampilkan di LCD TFT.

4.4.1 Tujuan Pengujian *Scaning* Dan Pengiriman I2C Wemos Menuju LCD

Pengujian ini dilakukan untuk memastikan Wemos D1 Mini dapat mengirimkan sinyal I2C dan dapat diterima oleh Arduino untuk di terjemahkan menjadi tampilan *display* di LCD TFT.

4.4.2 Alat Yang Digunakan Pengujian *Scaning* Dan Pengiriman I2C Wemos Menuju LCD

Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

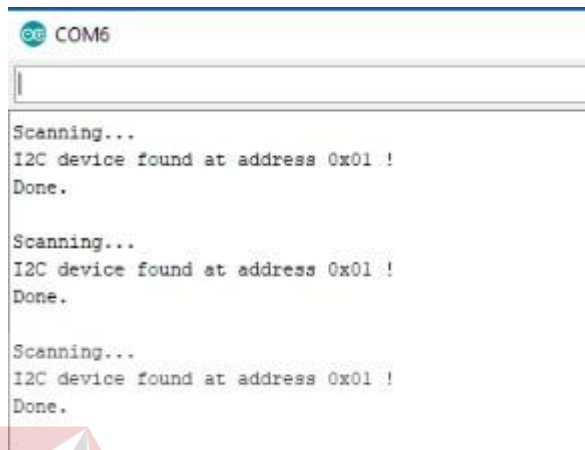
- a. *Power supply* 5V
- b. *Software* Arduino IDE
- c. Laptop / Komputer
- d. Kabel data *mikro* USB
- e. Wemos D1 Mini
- f. Kabel USB *male to male*
- g. Arduino UNO
- h. LCD TFT

4.4.3 Prosedur Pengujian *Scaning* Dan Pengiriman I2C Wemos Menuju LCD

Prosedur pengujian :

1. Menyalakan laptop atau komputer.
2. Menghubungkan Kabel data USB *mikro* pada Wemos D1 Mini dan sisi USB dihubungkan kepada USB laptop.
3. Menghubungkan kabel USB *male to male* yang satu menuju ke Wemos yang sisi lain menuju ke modul LCD.
4. Membuka *software* Arduino IDE.
5. Menyalin *Source code* dengan judul “Pengujian *scaning* I2C” pada lampiran.

6. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.
7. Meng-*Upload* program yang telah dibuat menuju ke Wemos.
8. Setelah meng-*upload* selesai, membuka *serial monitor*. Seperti Gambar 4.18.



```

COM6
Scanning...
I2C device found at address 0x01 !
Done.

Scanning...
I2C device found at address 0x01 !
Done.

Scanning...
I2C device found at address 0x01 !
Done.

```

Gambar 4. 18 Hasil *Serial Monitor Scaning* I2C

Dari Gambar 4.18 dapat dilihat bahwa setelah Wemos melakukan *scanning* Wemos menemukan alamat I2C LCD yang terhubung. Dari gambar tersebut juga bisa di simpulkan jika Wemos dapat digunakan untuk komunikasi I2C karena dapat memberi sinyal triger dan menerima sinyal respon dari perangkat yang terhubung oleh dirinya menggunakan I2C.

4.4.4 Hasil Pengujian Pengujian *Scaning* Dan Pengiriman I2C Wemos Menuju LCD

Dari Gambar 4.19 dapat disimpulkan bahwa Wemos dapat mengirimkan dan menerima sinyal I2C dengan contoh mencari alamat I2C perangkat yang terhubung pada dirinya.



Gambar 4. 19 Hasil Pengiriman Data I2C Wemos Menuju LCD TFT

Dari Gambar 4.19 dapat disimpulkan pengujian pengiriman data I2C Wemos menuju LCD berhasil hal ini dikarenakan data yang dikirim Wemos menuju LCD ditampilkan semua secara utuh dan tidak ada data yang hilang.

4.5 Pengujian Jarak Jangkauan WiFi Wemos



Gambar 4. 20 Diagram Blok Pengujian Jangkauan WiFi Wemos

Dalam pengujian ini dilakukan percobaan pengukuran jarak jangkauan Wemos. Jika dilihat dari Gambar 4.20 di gunakan 2 Wemos 1 digunakan sebagai *access point* dan yang lain sebagai *client* akan mencari Wemos AP. Pada sisi Wemos client akan diberi program untuk *scanning WiFi* yang dibuat oleh Wemos *access point*. Jika Wemos client masih dapat mendeteksi Wemos AP maka Wemos masih dapat terhubung. Jarak yang akan di gunakan untuk pengujian bebas penghalang adalah 4m, 6m, 11m, dan 16m. Untuk pengukuran dengan 1 penghalang menggunakan jarak 3m, 5m, 7m, dan 10m.

4.5.1 Tujuan Pengujian Jarak Jangkauan *WiFi Wemos*

Pengujian ini dilakukan untuk mengetahui jarak jangkauan Wemos untuk menentukan jarak efektif penggunaan Wemos untuk berkomunikasi.

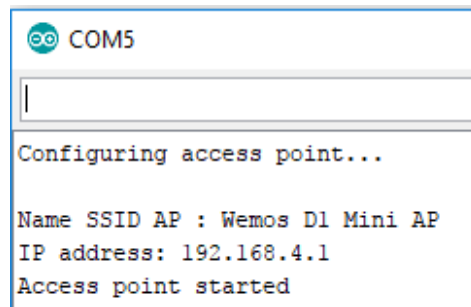
4.5.2 Alat Yang Digunakan Pengujian Jarak Jangkauan *WiFi Wemos*

Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

- a. 2 buah *Power supply* 5V
- b. *Software* Arduino IDE
- c. Laptop / Komputer
- d. 2 buah Kabel data *mikro* USB
- e. 2 buah Wemos D1 Mini

4.5.3 Prosedur Pengujian Jarak Jangkauan *WiFi Wemos*

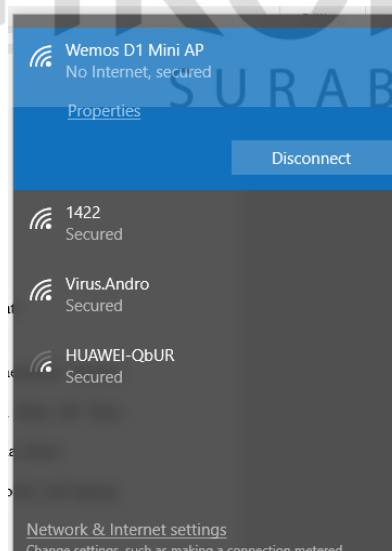
1. Menyalakan laptop atau komputer.
2. Menghubungkan Kabel data USB *mikro* pada Wemos D1 Mini yang akan menjadi *access point* dan sisi USB dihubungkan kepada USB laptop.
3. Membuka *software* Arduino IDE.
4. Menyalin *Source code* dengan judul “Pengujian Wemos sebagai *access point*” pada lampiran.
5. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.
6. Meng-*Upload* program yang telah dibuat menuju ke Wemos.
7. Setelah proses *upload* selesai, membuka *serial monitor*. Seperti Gambar 4.21.



Gambar 4. 21 Wemos Berhasil Menjadi *Access Point*

Dari Gambar 4.21 menunjukkan Wemos telah berhasil membuat dirinya menjadi *access point* dengan nama SSID adalah Wemos D1 Mini AP. Dari gambar di atas juga menunjukkan bahwa Wemos siap menerima client.

8. Melepaskan sambungan USB Wemos yang menuju ke laptop dan memindahkan pada catu daya USB 5V sebagai pengganti daya yang awalnya berasal dari laptop.
9. Membuka *WiFi* pada laptop, memastikan AP Wemos dapat terdeteksi oleh *WiFi* laptop. Seperti pada Gambar 4.22.



Gambar 4. 22 *Access Point* Wemos Terdeteksi Oleh Laptop

Dari Gambar 4.22 terbukti bahwa *access point* Wemos terdeteksi oleh laptop, selain itu dari Gambar 4.22 juga terbukti bahwa *access point* yang dibuat oleh Wemos dapat dihubungkan dengan client pada contoh ini client adalah laptop.

10. Menghubungkan Kabel data USB *mikro* pada Wemos D1 Mini yang akan menjadi client dan sisi USB dihubungkan kepada USB laptop.
11. Membuka *software* Arduino IDE.
12. Menyalin *Source code* dengan judul “Pengujian *scanning Access Point* Wemos” pada lampiran.
13. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.
14. Meng-*Upload* program yang telah dibuat menuju ke Wemos.
15. Setelah proses *upload* selesai, membuka *serial monitor*. Seperti Gambar 4.23.



Gambar 4. 23 AP Yang Dibuat Wemos Terlihat Oleh Wemos Client

Dari Gambar 4.23 dapat dilihat bahwa Wemos client dapat menemukan *access point* yang di buat oleh Wemos AP. Dari gambar tersebut pula juga dapat di simpulkan bahwa antara Wemos AP dan Wemos client berada pada jarak yang baik dan dapat terhubung.

16. Buat jarak antara Wemos AP dan Wemos client sampai Wemos client tidak dapat menemukan Wemos AP.

4.5.4 Hasil Pengujian Jarak Jangkauan WiFi Wemos

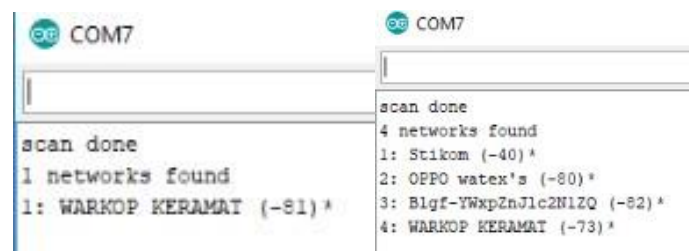
Dari pengujian jarak yang dilakukan oleh penulis maka penulis mencatat hasil dari percobaan di atas seperti pada Tabel 4.1.

Tabel 4. 1 Hasil Pengujian Jarak Jangkauan Wemos

Jarak	Keterangan
Bebas Penghalang	
4m	Tersedia
6m	Tersedia
11m	Tersedia
16m	Tidak tersedia
Menggunakan Penghalang	
3m	Tersedia
5m	Tersedia
7m	Tersedia
10m	Tidak tersedia

Dari hasil percobaan dapat di simpulkan jarak efektif Wemos dalam berkomunikasi maksimal tanpa penghalang adalah sejauh 11 meter dan dengan penghalang maksimal 7 meter. Dari data di atas jangkauan *range* WiFi Wemos dengan jarak efektif adalah 5 meter.

Hasil dari *scanning* di atas dapat dilihat pada Gambar 4.24



```

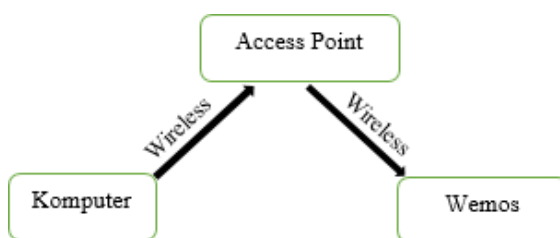
COM7
scan done
1 networks found
1: WARKOP KERAMAT (-81) *

COM7
scan done
4 networks found
1: Stikom (-40) *
2: OPFO watek's (-80) *
3: Blgf-YWxpZnJlc2N1ZQ (-82) *
4: WARKOP KERAMAT (-73) *

```

Gambar 4. 24 Hasil *Scanning* WiFi (Kanan AP Terlihat Dan Kiri AP Tidak Terlihat)

4.6 Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet



Gambar 4. 25 Diagram Blok Pengujian Penerimaan Data Dari Pc Menuju Wemos

Dalam pengujian ini dilakukan pengiriman data dari PC menuju ke Wemos menggunakan protokol telnet. Dari Gambar 4.25 dapat dilihat arah data adalah mulai dari komputer menuju ke *access point* lalu oleh *access point* akan dilanjutkan kepada Wemos.

Dalam pengujian ini akan dikirim data sejumlah 1000 menuju ke Wemos sebanyak 10 kali dan data yang diterima Wemos akan dimonitor melalui *serial monitor* pada Arduino IDE serta di hitung waktu lama pengiriman data.

4.6.1 Tujuan Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet

Pengujian ini dilakukan untuk memastikan Wemos dapat menerima data secara lengkap dan cepat serta tidak ada paket loss saat pengiriman.

4.6.2 Alat Yang Digunakan Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet

Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

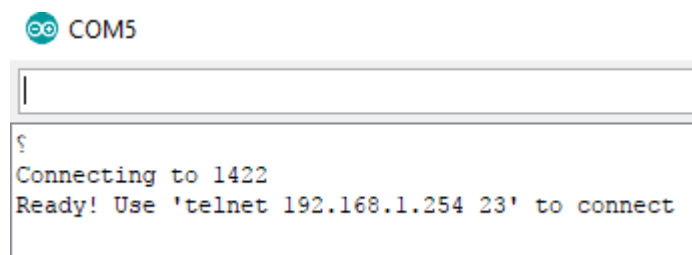
- a. *Power supply 5V*

- b. *Akses Point* (Jaringan WLAN)
- c. *Software* Arduino IDE
- d. Laptop / Komputer
- e. Kabel data *mikro* USB
- f. Wemos D1 Mini
- g. *Software* pengirim data Wemos

4.6.3 Prosedur Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet

Prosedur pengujian :

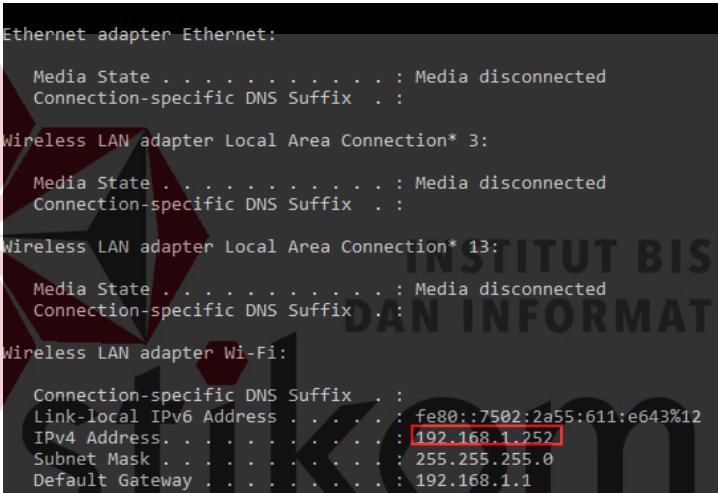
1. Menyalakan laptop atau komputer.
2. Menghubungkan Kabel data USB *mikro* pada Wemos D1 Mini yang akan menjadi *access point* dan sisi USB dihubungkan kepada USB laptop.
3. Membuka *software* Arduino IDE.
4. Menyalin *Source code* dengan judul “Pengujian penerimaan data melalui telnet” pada lampiran.
5. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.
6. Meng-*Upload* program yang telah di buat menuju ke Wemos.
7. Setelah proses *upload* selesai, membuka *serial monitor*. Seperti Gambar 4.26.



Gambar 4. 26 Wemos Siap Untuk Telnet

Dari Gambar 4.26 dapat disimpulkan bahwa Wemos sudah terhubung dengan *access point* dan sudah siap untuk menerima data. Dari gambar tersebut juga dapat diketahui bahwa alamat untuk Wemos berkomunikasi terdapat pada IP 192.168.1.254 dan *Port* 23.

8. Menyambungkan *WiFi* laptop kepada *access point* yang sama dengan Wemos yang akan menerima data
9. Membuka *command prompt* dan melihat IP yang diterima oleh laptop berada pada 1 jaringan dengan Wemos penerima. Seperti pada Gambar 4.27.



```

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 3:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 13:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

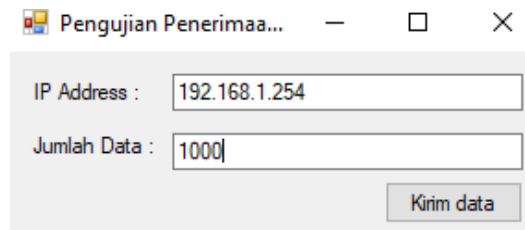
Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::7502:2a55:611:e643%12
    IPv4 Address. . . . . : 192.168.1.252
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1
  
```

Gambar 4. 27 IP Yang Didapatkan Laptop Dari *Access Point*

Dari Gambar 4.27 dapat diketahui bahwa antara laptop dan Wemos berada di 1 jaringan yang sama yaitu di jaringan 192.168.1.0.

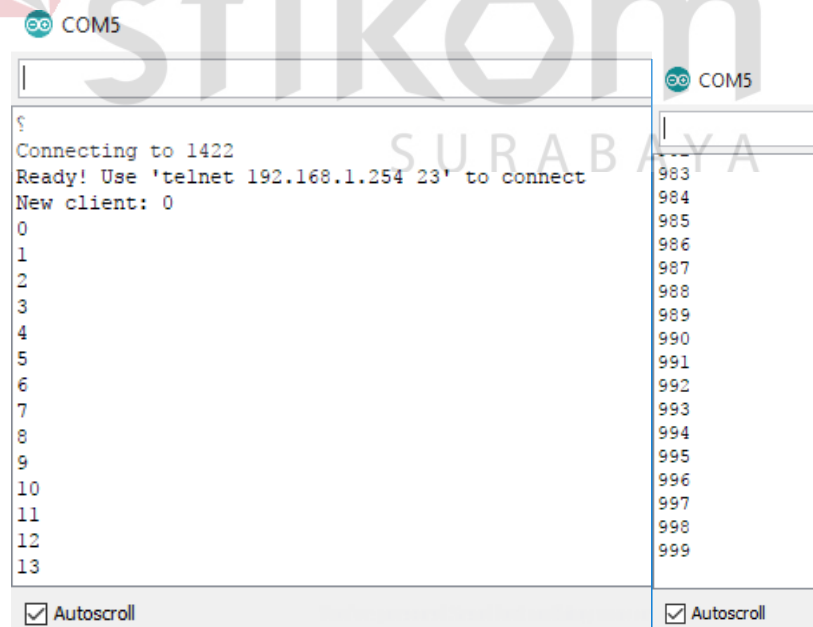
10. Melakukan ping pada IP Wemos melalui *command prompt* bila Wemos membalas ping maka pengujian pengiriman data siap di mulai.
11. Membuka *software* pengujian penerimaan data Wemos. Seperti Gambar 4.28



Gambar 4. 28 *Software* Pengujian Penerimaan Wemos

Pada Gambar 4.28 adalah gambar tampilan *software* untuk pengujian penerimaan Wemos. Terdapat 2 kotak kolom yang harus di isi sebelum melakukan pengiriman data. Kolom pertama adalah *IP Address* ini berfungsi untuk menuliskan alamat IP yang akan dikirimkan data, pada kolom kedua adalah alamat I2C, Setelah semua terisi mengklik kirim data maka *software* ini akan mengirimkan data sebanyak jumlah data menuju ke *IP address*.

12. Setelah *software* mengirimkan data, membuka kembali *serial monitor* pada Arduino IDE. Perhatikan Gambar 4.29.



Gambar 4. 29 Data Yang Diterima Wemos (Kiri Awal Data Kanan Akhir Data)

Dari Gambar 4.29 bahwa data yang diterima Wemos adalah mulai dari 0 – 999 artinya pengiriman yang dilakukan adalah berhasil karena Wemos menerima 1000 data tanpa loss.

4.6.4 Hasil Pengujian Penerimaan Data Wemos Dari PC Melalui Telnet

Dari Gambar 4.29 dapat dilihat bahwa saat pengiriman data tidak ada data yang loss hal ini dapat disimpulkan bahwa pengiriman data Wemos 100% berhasil. Data dari 10 percobaan dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Hasil 10 Kali Percobaan Pengiriman Data

Percobaan ke :	Jumlah data	Waktu pengiriman (ms)	Keterangan
1	1000	1230	Data Sesuai
2	1000	1250	Data Sesuai
3	1000	1150	Data Sesuai
4	1000	1190	Data Sesuai
5	1000	1350	Data Sesuai
6	1000	997	Data Sesuai
7	1000	967	Data Sesuai
8	1000	1054	Data Sesuai
9	1000	1069	Data Sesuai
10	1000	956	Data Sesuai

Dari Tabel 4.2 dapat di simpulkan bahwa dari 10 percobaan pengiriman data kesepuluh percobaan berhasil karena tidak ada data loss. Untuk waktu yang ditempuh dalam pengiriman 1000 data memiliki waktu 1190ms atau bisa di katakan 1 detik, dari hal ini dapat disimpulkan bahwa pengiriman data dapat dikatakan sangat cepat dikarenakan dalam mengirim 1000 data hanya memerlukan waktu 1 detik.

4.7 Pengujian Sistem

Dalam pengujian sistem ini yang akan di uji adalah sistem pada Gambar 3.1.

Adapun cara pengujianya akan dibuat 3 skenario pengujian antara lain :

Skenario 1 : Laptop akan mengirim data kepada Wemos sebanyak 20x lalu akan dilihat apakah dalam percobaan tersebut apakah ada kesalahan.

Skenario 2 : Laptop akan mengirim data kepada 1 Wemos dan 1 LCD, mulai dari LCD kosong sampai terisi akan dilihat apakah ada kesalahan dalam percobaan tersebut.

Skenario 3 : Laptop akan mengirim data kepada 1 Wemos dan 1 LCD, mulai dari LCD berisi data sebelumnya sampai berubah ke data yang baru akan dilihat apakah ada kesalahan dalam percobaan tersebut.

4.7.1 Tujuan Pengujian Sistem

Pengujian sistem ini dilakukan untuk mengetahui apakah sistem dapat berjalan dengan lancar dalam menjalankan tugasnya dan untuk mengetahui apakah sistem ini mempunyai kesalahan dalam pelaksanaannya.

4.7.2 Alat Yang Digunakan Pengujian Sistem

Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

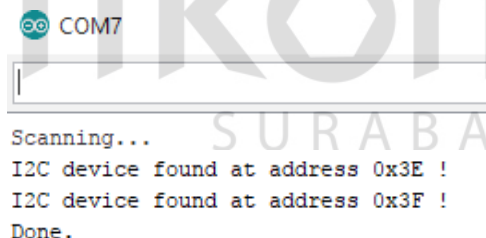
- a. 3 buah *Power supply* 5V
- b. *Software* Arduino IDE
- c. *Software* Product Price Display Wemos
- d. Laptop / Komputer

- e. 2 buah Kabel data *mikro* USB
- f. 2 buah Wemos D1 Mini
- g. 2 Buah LCD TFT
- h. 2 Buah Arduino UNO
- i. Kabel LAN

4.7.3 Prosedur Pengujian Sistem

Prosedur pengujian :

1. Menyalakan laptop atau komputer.
2. Menghubungkan Kabel data USB *mikro* pada Wemos D1 Mini ke 1.
3. Menghubungkan Wemos ke 1 dengan modul LCD ke 1 menggunakan USB.
4. Menghubungkan modul LCD ke 1 dengan LCD ke 2 menggunakan USB.
5. Membuka *software* Arduino IDE.
6. Melakukan *scanning address* I2C untuk mendapatkan alamat I2C modul LCD.



```

COM7
Scanning...
I2C device found at address 0x3E !
I2C device found at address 0x3F !
Done.

```

Gambar 4. 30 Hasil *Scaning* Wemos Ke 1

Dari Gambar 4.30 dapat diketahui bahwa terdapat 2 modul LCD di Wemos ke

1. Dimana LCD ke 1 beralamat 0x3E dan LCD 2 beralamatkan 0x3F.

7. Menyalin *Source code* dengan judul “Pengujian sistem” pada lampiran.
8. Memastikan *port* COM untuk komunikasi Wemos antara di Arduino IDE dan *device manager* sama.

9. Meng-*Upload* program yang telah di buat menuju Wemos ke 1.
10. Setelah proses *upload* selesai, membuka *serial monitor*. Seperti Gambar 4.31.

```

ESP8266 : Connecting to Stikom
.....Wemos Ready

```

Gambar 4. 31 Wemos Ke 1

Dari Gambar 4.31 menunjukkan Wemos sudah siap digunakan untuk pengiriman data melalui telnet.

11. Melepaskan USB yang terhubung pada laptop pindah ke power supply 5V.
12. Melihat pada layar modul LCD bila di layar sudah tampil “*Ready*” maka LCD sudah siap untuk menampilkan data. Seperti pada Gambar 4.32.



Gambar 4. 32 Tampilan LCD Menunjukkan *Ready*

Dari Gambar 4.32 diketahui bahwa modul LCD siap untuk digunakan dikarenakan pada tampilan LCD sudah menunjukkan tampilan *Ready*.

13. Membuka *software* Product Price Display Wemos.
14. Setelah dibuka, kemudian mengisi data yang akan dikirim kepada LCD 1, yaitu IP *Adress*, Alamat I2C Arduino, Kode Barang, Nama Barang, Harga Barang, Diskon Barang. Seperti pada Gambar 4.33.

Gambar 4. 33 Tampilan *Software* Aplikasi Display Harga

Dari Gambar 4.33 diketahui bahwa kiriman ditujukan kepada Wemos ke 1 karena alamat IP menunjukkan Wemos ke 1. Dari alamat I2C menunjukkan data akan di kirim kepada LCD ke 1 karena alamat I2C menunjuk ke alamat LCD ke 1. Lalu yang akan di kirim adalah data berupa nama barang yaitu Aqua dan harga barang adalah 3000 ini menunjukkan LCD ke 1 digunakan untuk *display* Aqua yang memiliki harga Rp. 3000. Setelah selesai data benar lalu mengklik *send*, maka data akan di kirimkan pada Wemos ke 1.

15. Data yang dikirim diterima oleh Wemos ke 1 dan ditampilkan pada LCD ke 1.

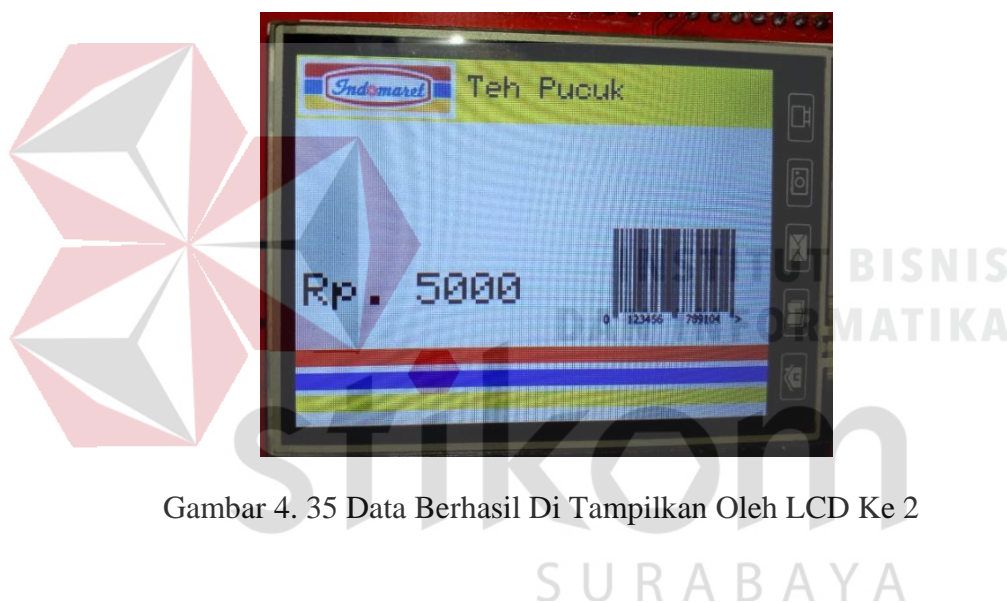
Seperti pada Gambar 4.34.



Gambar 4. 34 Data Berhasil Di Tampilkan Oleh LCD Ke 1

Dari Gambar 4.34 dapat dilihat bahwa LCD ke 1 menampilkan data yang di kirim melalui *software* Aplikasi Display Harga. Dari gambar tersebut juga dapat disimpulkan bahwa pengiriman data berhasil.

16. Melakukan perubahan alamat I2C menjadi 3 untuk melakukan pengiriman data pada Wemos ke 2 LCD ke 2. Lalu pada nama barang di ganti dengan Teh Pucuk dan pada harga barang diganti menjadi 5000. Lalu perhatikan apakah LCD ke 2 menampilkan data nama barang dan harga barang yang sesuai dengan yang dikirim. Seperti pada gambar 4.35.



Gambar 4. 35 Data Berhasil Di Tampilkan Oleh LCD Ke 2

17. Menghubungkan Wemos ke 2 dengan modul LCD ke 2 menggunakan USB.
18. Membuka *software* Arduino IDE.
19. Melakukan *scanning address* I2C untuk mendapatkan alamat I2C modul LCD.
20. Meng-*Upload* program menuju Wemos ke 2.
21. Setelah proses *upload* selesai, kemudian membuka *serial monitor*,
Memastikan Wemos kedua mendapat IP yang berbeda dengan IP pertama.
22. Melepas USB yang terhubung pada laptop dan dipindah ke power supply 5V.
23. Melakukan langkah 17 – 23 untuk Wemos ke 2 dan LCD ke 2.

24. Setelah semua Wemos dan LCD dapat digunakan atau dalam kondisi *ready* kemudian melakukan skenario 3 percobaan.

4.7.4 Hasil Pengujian Sistem

Dari 3 skenario percobaan didapatkan hasil dari skenario seperti dapat dilihat pada Tabel 4.3.

Tabel 4. 3 Hasil Percobaan Skenario

No.	Data Dikirim			Tujuan		Keterangan
	Kode Barang	Nama barang	Harga Barang	Wemos ke -	LCD ke -	
1	1234567	Indomie goreng	Rp 3.000	1	1	Sesuai
2	2345678	Teh Pucuk 330ml	Rp 4.000	2	3	Sesuai
3	3456789	Aqua	Rp 3.000	2	3	Sesuai
4	3214656	Kepal Milo	Rp 20.000	1	3	Sesuai
5	3334567	Donat Kentang	Rp 5.000	1	2	Sesuai
6	4477788	Sarimi Isi 3	Rp 5.500	1	1	Sesuai
7	012313	Nasi Goreng	Rp 10.000	1	2	Sesuai
8	5385422	Forvita 200g	Rp 9.900	2	3	Sesuai
9	9948373	Tissu Paseo	Rp 9.500	2	1	Sesuai
10	8746888	Martabak spesial	Rp 30.000	1	3	Sesuai
11	7584992	Terang Bulan	Rp 35.000	1	1	Sesuai
12	2345634	Sempol	Rp 12.900	1	2	Sesuai
13	4529786	Roti Goreng	Rp 20.000	2	3	Sesuai
14	2897638	Susu Sapi 100ml	Rp 32.000	2	1	Sesuai
15	8883339	Softex	Rp 31.250	2	3	Sesuai
16	2228837	Tissu Multi	Rp 11.150	1	1	Sesuai
17	6667788	Bebelac Gold 3	Rp 114.500	1	2	Sesuai
18	9993388	SUN Bubur	Rp 6.000	2	3	Sesuai
19	3398644	Nutrilon 3	Rp 12.500	2	2	Sesuai
20	6668877	Tabung Gas	Rp 18.600	1	1	Sesuai

Dari tabel percobaan di atas dapat memenuhi 3 skenario yang di ujikan.

Dapat dipisah sebagai berikut :

1. Pada saat pengiriman nomor 1 -5 didapati bahwa ini sesuai dengan skenario kedua dimana *user* harus mengisi data mulai dari LCD kosong atau *ready* menjadi menampilkan data yang di kirim.

Adapun hasil dari pengujian tersebut adalah seperti ditunjukkan oleh Tabel 4.3.

Data berhasil di kirim ke alamat yang tepat.

2. Pada saat pengiriman nomor 6 -10 didapati bahwa ini sesuai dengan skenario ketiga dimana *user* harus mengganti data yang di tampilkan LCD dari data lama menjadi data baru.

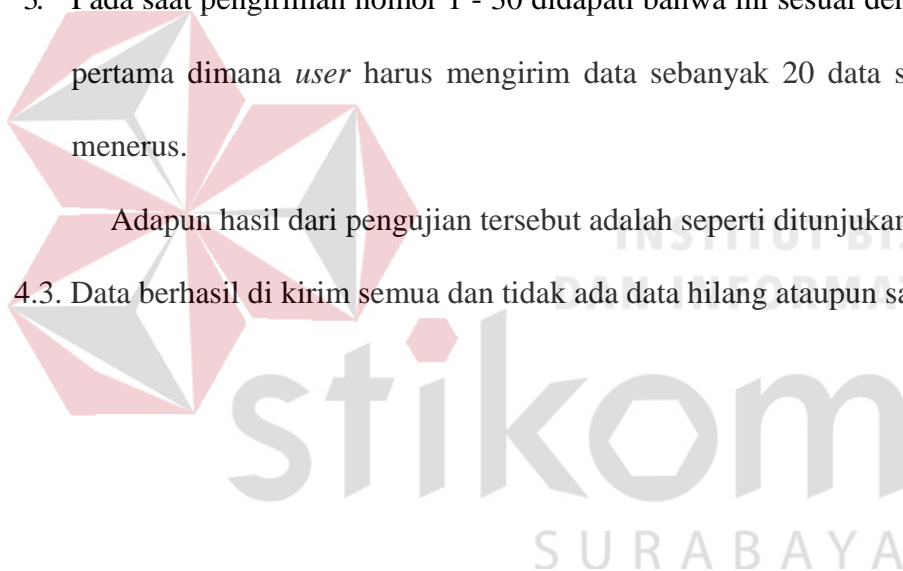
Adapun hasil dari pengujian tersebut adalah seperti ditunjukkan oleh Tabel 4.3.

Data berhasil di kirim dan tampilan LCD berubah sesuai data baru yang dikirim.

3. Pada saat pengiriman nomor 1 - 30 didapati bahwa ini sesuai dengan skenario pertama dimana *user* harus mengirim data sebanyak 20 data secara terus - menerus.

Adapun hasil dari pengujian tersebut adalah seperti ditunjukkan oleh Tabel

- 4.3. Data berhasil di kirim semua dan tidak ada data hilang ataupun salah alamat.



BAB V

PENUTUP

5.1 Kesimpulan

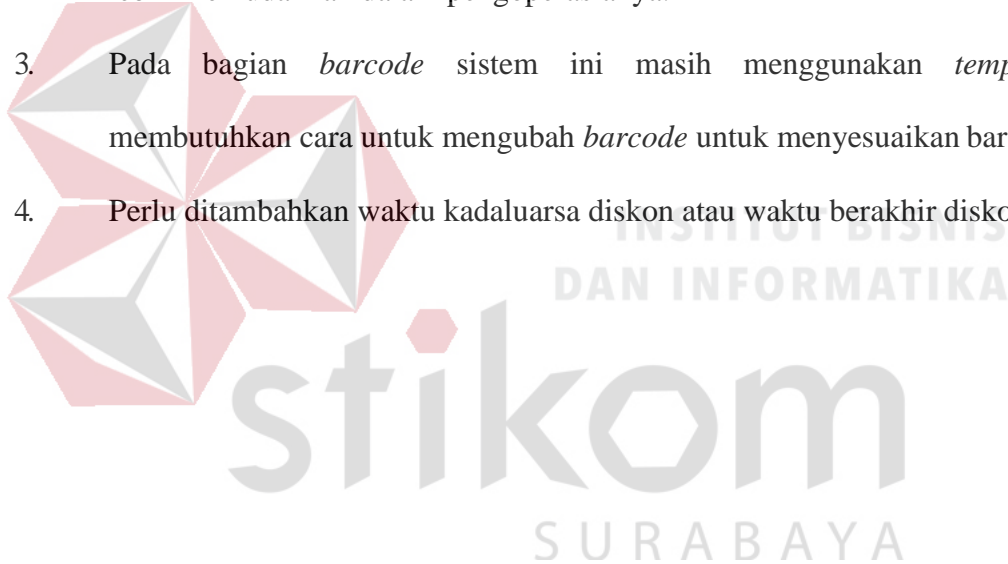
Kesimpulan yang didapat dari analisis dan percobaan sistem yang telah di teliti tersebut antara lain sebagai berikut :

1. Perbedaan harga barang diminimalisir dengan menggunakan sistem Rancang Bangun Alat Display Harga Secara Otomatis dengan Menggunakan LCD Grafis ini dikarenakan pada saat pengujian tidak pernah ada data yang tidak terkirim semua data dapat terkirim dengan sempurna dengan waktu yang sangat cepat yaitu kurang dari 1 detik.
2. *Software* yang digunakan dalam sistem ini cukup mudah pada tampilan *software* dapat dilihat pada Gambar 4.30. Dimana *user* cukup mengetik alamat yang dituju dan data yang akan dikirim lalu klik kirim data maka label harga di rak barang akan langsung berubah sesuai dengan data yang dikirim.
3. Dari desain sistem ini juga dapat dikatakan *portable* karena dapat di lepas dan diganti pada setiap modulnya dengan mudah. Serta untuk sambunganya sendiri hanya cukup menggunakan kabel USB *male to male* yang langsung di tancapkan saja.
4. Rancang Bangun Alat Display Harga Secara Otomatis dengan Menggunakan LCD Grafis dapat memudahkan swalayan dalam melakukan operasionalnya juga dapat membuat sebuah swalayan lebih modern dan canggih.

5. Pada pelaksanaanya sistem ini akan dapat meminimalisir perbedaan harga antara di rak dan kasir karena sistem sudah tercentral karena ketika harga di kasir dirubah maka *display* di rak pasti juga akan ikut berubah.

5.2 Saran

1. Perlu ditambahkan fitur *auto scanning* pada I2C agar ketika modul LCD diganti tidak perlu melakukan registrasi ulang terutama melalui program.
2. Pada sisi *software* sistem ini membutuhkan sebuah *software inventory* yang lebih memudahkan dalam pengoperasiannya.
3. Pada bagian *barcode* sistem ini masih menggunakan *template*, membutuhkan cara untuk mengubah *barcode* untuk menyesuaikan barang.
4. Perlu ditambahkan waktu kadaluarsa diskon atau waktu berakhir diskon.



DAFTAR PUSTAKA

- Agung, Rizky. 2013. Tutorial Cara Setting Mikrotik RB750 via Winbox. [Online], (<https://mikrotikindo.blogspot.co.id/2013/04/tutorial-cara-setting-mikrotik-rb750-winbox.html>, diakses 25 Mei 2018)
- Basu Swastha, dan Irawan. 2003. *Manajemen Pemasaran Modern*. Yogyakarta, : Liberty Yogyakarta
- Fauzan, Abdul. 2016. Proyek Arduino 4 [online], (<http://www.charisfauzan.net/2016/04/proyek-arduino-4-animasi-grafik.html> Diakses 6 Juni 2018)
- Hidayatullah, Priyanto dan Jauhari Kairul Kawastara. 2015. *Pemrograman WEB(HTML / CSS / JavaScRIPt / PowerDesigner / XAMPP / CodeIgniter / JQuery)*. Bandung : Informatika
- Kadir, Abdul. 2003. *Jaringan Komputer*. Yogyakarta : Andi
- Peraturan Menteri Perdagangan Republik Indonesia. 2013. [online], (<http://www.kemendag.go.id/files/regulasi/2015/09/28/73m-dagper92015-id-1446169056.pdf>), Diakses tanggal 1 Februari 2018)
- Sianipar. 2017. *100 Kasus Pemrograman Visual C#*. Jakarta:Andi
- Suhartanto, D & Nuralia, A. 2001. *Citra Supermarket: Pengaruhnya Terhadap Perilaku Konsumen*: Jurnal Kajian Bisnis No.23/Mei-Agustus
- Utomo, Deny Setyo. 2017. *Product Price Display Using Wemos*. Surabaya : Institut Bisnis dan Informatika Stikom Surabaya
- Wiki.Wemos.cc. 2017. *D1 Mini*, [online], (https://wiki.Wemos.cc/products:d1:d1_mini#version), Diakses tanggal 1 Juli 2018)