

BAB 2

DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian Sugiono, Tutuk Indriyani dan Maretha Ruswiansari pada tahun 2017 yang berjudul “Kontrol Jarak Jauh Sistem Irigasi Sawah Berbasis *Internet Of Things (IoT)*” membahas tentang sistem kontrol pintu air irigasi secara jarak jauh melalui aplikasi android, sehingga pada aplikasi tersebut pengguna dapat menutup maupun membuka pintu air dan mengetahui ketinggian air tersebut [4]. Pada penelitian ini menggunakan komponen seperti *Wemos D1 ESP8266* sebagai mikropengendali, sensor ultrasonik *HC-SR04* berfungsi untuk mengukur ketinggian air, motor servo dan lain-lain. Hasil dari penelitian ini alat bekerja dengan baik, ditandai pada sistem kedua portal yang digerakan oleh motor servo yang mengambil data dari dari sensor ultrasonik ketika ketinggian air kurang dari 3 cm portal akan terbuka dan ketika ketinggian air 3 cm portal akan tertutup. Dari penelitian tersebut untuk menutupi kekurangan penulis menambahkan sensor kelembaban agar mengetahui nilai kelambaban tanah persawahan.

Pada penelitian Wiranto, Budi Indra Setiawan dan Satyanto Krido Saptomo pada tahun 2014 yang berjudul “Sistem Kontrol Irigasi Otomatis Nirkabel” membahas tentang sistem irigasi otomatis yang mampu menyediakan air untuk tanaman dengan kondisi yang sesuai dengan kebutuhan. Area yang luas pada sistem irigasi ini menggunakan sistem *wireless sensor network (WSN)* [5]. Kelebihannya diantaranya mudahnya *download* data dan perangkat mudah untuk diinstalasi sehingga memudahkan dalam memonitoring wilayah irigasi. Komponen yang digunakan perangkat *Xbee/Xbee pro* yang berguna sebagai sistem nirkabel menggunakan gelombang radio sebagai sistem komunikasinya dan perangkat modem GSM untuk sistem komunikasi SMS. Tujuannya dari penelitian ini meningkatkan efisiensi sistem irigasi kontrol nirkabel (*wireless*), sedangkan manfaatnya memberikan sistem irigasi otomatis alternatif untuk wilayah dengan ketersediaan air yang terbatas. Dari penelitian ini penulis merancang

menggunakan NodeMCU ESP8266 karena penulis mengharapkan perancangan ini dapat berkomunikasi menggunakan internet.

Penelitian Sudirman Sirait, Satyanto K. Saptomo, M. Tanuar J. dan Purwanto pada tahun 2015 yang berjudul “Rancang Bangun Sistem Otomatisasi Irigasi Pipa Lahan Sawah Berbasis Tenaga Surya” membahas tentang rancangan irigasi yang bertujuan untuk pemberian air secara optimal dengan sistem kontrol otomatis sehingga menjaga permukaan air pada lahan persawahan dengan ketinggian tertentu sesuai dengan kebutuhan tanaman, sehingga data meningkatkan produktivitas dan efisiensi penggunaan air irigasi pada lahan persawahan [6]. Sistem ini dibuat dengan memanfaatkan teknologi digital, mikrokontroler dan jaringan sensor. Komponen yang digunakan mikrokontroler Arduino Uno ATmega328P sebagai sistem kendali untuk menggerakkan sistem aktuasi kran air listrik *Valworx 561086* berdasarkan kelembapan tanah dan tinggi muka air pada lahan persawahan yang dideteksi oleh sensor. Pada penelitian ini sistem kontrol otomatis dapat berfungsi dengan baik. Dari penelitian ini penulis menggunakan NodeMCU ESP8266 agar perancangan dapat melakukan monitoring dan kontroling menggunakan internet.

Penelitian Bambang Tri Wahyu Utomo dan Hasan Saifudi pada tahun 2014 yang berjudul “Prototyping sistem Monitoring Ketinggian Air Dan Pengendalian Pintu Air Pada Jaringan Irigasi Berbasis Mikrokontroler ATmega16 Dengan Menggunakan *Short Message Service* (SMS)” membahas tentang sistem monitoring ketinggian air dan pengendalian pintu air pada irigasi menggunakan ATmega16 dengan notifikasi berupa *Short Message Service* (SMS) [7]. Pada penelitian ini komponen yang digunakan ATmega16 sebagai pengendali mengatur motor DC, sensor ketinggian air, *driver ULN2003*, motor DC, LCD (*Liquid Crystal Display*), modem SMS *WAVECOM M13026B* sebagai perintah pembuka pintu air / data informasi ketinggian air via SMS, *Keypad* dan lain-lain. Sistem kerja untuk pengendalian pintu air dilakukan dengan cara mengirimkan sms dengan kode PNT00 ke mikrokontroler sehingga dapat terbuka maupun tertutup pintu air tersebut, sedangkan untuk mengetahui ketinggian air tersebut mengirimkan

sms dengan kode *PAIR* ke mikrokontroler. Pada hasil penelitian ini dapat memberikan informasi ketinggian air dengan cara *request* sms ke terhubung dengan mikrokontroler ATmega16 secara otomatis akan membalas sms yang berisi mengenai informasi ketinggian air dan pintu air tersebut. Dari penelitian ini penulis menggunakan NodeMCU ESP8266 agar perancangan dapat melakukan monitoring dan kontroling menggunakan internet.

Penelitian Aiman Fakhri pada tahun 2017 yang berjudul “Sistem Monitoring Otomasi Irigasi Dengan Penjadwalan Menggunakan Logika *Fuzzy*” membahas tentang sistem penjadwalan buka tutup pintu air menggunakan logika *fuzzy*. Logika *fuzzy* berfungsi untuk menentukan berapa lama waktu irigasi yang diperlukan dan mengatur penjadwalan pengiriman data ketinggian air [8]. Komponen yang digunakan pada penelitian ini seperti Arduino Uno sebagai mikrokontroler, sensor ultrasonik yang berfungsi untuk mengukur ketinggian air, *raspberry pi*, *wireless dongle*, servo dan lain-lain. Pada penelitian ini sistem yang dibuat telah mampu menunjukkan akurasi yang baik dan sistem monitoring telah berjalan menggunakan fasilitas dari *freeboard.io* dan logika *fuzzy* berhasil diterapkan. Dari penelitian tersebut untuk menutupi kekurangan penulis menambahkan sensor kelembaban agar mengetahui nilai kelembaban tanah persawahan dan menggunakan NodeMCU ESP 8266 sebagai mikrokontroler.

Penelitian Muammar Michael, Agung Rahmadi, H. Aji dan M. Gazi pada tahun 2018 yang berjudul “Survei Pengaruh Ketinggian Genangan Air Pada Tanaman Padi di Daerah Bandung Timur” membahas tentang pengaruh pertumbuhan padi terhadap genangan air . Pengukuran ketinggian genangan air dari permukaan tanah hingga muka air menggunakan penggaris yang dilakukan secara acak berdasarkan petakan-petakan sawah yang ada. Dengan mengambil 6 sampel petakan sawah yang berbeda diperoleh hasil ketinggian genangan air pada tanaman padi berkisar dari -2 cm sampai 2 cm. Dan studi literature menunjukkan ketinggian genangan air yang optimum untuk tanaman padi adalah -2,5 cm hingga 7,5 cm [9]. Dari penelitian ini dapat diambil kesimpulan bahwa ketinggian air dengan batas 5 cm merupakan batas optimum untuk tanaman padi pada area persawahan.

2.2 DASAR TEORI

2.2.1 Masa Pertumbuhan Tanaman Padi

Dalam pertumbuhan tanaman padi terdapat beberapa tahap meliputi tahap awal pertumbuhan atau perkembangan akar, tahap pembentukan anakan, tahap bunting atau pembentukan bulir padi, tahap pembungaan dan terakhir menjelang masa panen. Masa pertumbuhan perlu adanya pengelolaan air yang baik semisal pada tanaman padi yang memiliki umur pendek [10]. Berikut contoh pengelolaan air :

- a. Pada saat setelah tanaman padi ditanam atau pemupukan awal, selama 3 hari sawah tidak diairi tetapi sawah tersebut dibiarkan begitu saja dalam keadaan macak-macak atau jenuh air
- b. Pada saat umur tanaman padi empat hari sampai dengan empat belas hari setelah tanam sawah tersebut diberikan air sampai tergenang sedikit demi sedikit hingga ketinggian air mencapai 7 cm sampai dengan 10 cm.
- c. Pada saat lima belas hari sampai dengan tiga puluh hari setelah masa tanam sawah digenangi air secara terus-menerus hingga mencapai ketinggian antara 3 cm sampai dengan 5 cm. Pada fase ini kekurangan air akan mengakibatkan berkurangnya jumlah anakan. Dan juga genangan yang terlalu tinggi akan mengakibatkan menghambatnya pertumbuhan anakan, oleh sebab itu pada fase sangat diperhatikan dari faktor ketinggian maupun tingkat kelembaban tanah pada sawah tersebut.
- d. Pada saat tiga puluh hari setelah ditanam, air pada area petak sawah dikeluarkan dan dibiarkan macak-macak. Dan pada masa ini dilakukan pemberian pupuk susulan yang pertama.
- e. Pada saat tiga puluh sampai dengan lima puluh hari setelah masa ditanam, petakan sawah digenangi dengan ketinggian air mencapai 5 cm sampai dengan 10 cm

- f. Pada saat lima puluh hari setelah ditanam, dilakukan pengeringan dan dibiarkan macak-macak, setelah itu dilakukan pemberian pupuk susulan kedua.
- g. Pada saat lima puluh hari setelah ditanam, dilakukan penggenangan dengan ketinggian 10 cm smpai pada masa berbunga secara serempak. Apabila terjadi kekurangan air pada fase ini akan melemahkan pembentukan malai dan pembuahans sehingga banyak adanya butir padi yang kosong atau tidak terisi calon beras. Selanjutnya air yang ada pada area petak sawah dikeluarkan lagi dari petakan dan setidaknya dibiarkan selama 5 hari dalam keadaan macak-macak. Penggenangan dilanjutkan lagi sampai ketinggian 10 cm hingga gabah berisi penuh.
- h. Pada saat lima belas hari sebelum padi dipanen, maka area petak sawah dikeringkan.



Gambar 2.1 Area Persawahan[11]

2.2.2 NodeMCU ESP8266

NodeMCU ESP8266 adalah suatu *platform internet of things* yang bersifat *opensource*, yang terdiri dari perangkat keras *System On Chip* ESP8266 buatan *Esperessif System*. NodeMCU dapat dikatakan sebagai *board* Arduino yang terkoneksi dengan ESP8266. NodeMCU telah terdapat ESP8266 ke dalam sebuah *board* yang sudah terintegrasi dengan berbagai *feature* sebagai mikropengendali dan kapasitas akses terhadap *wifi* dan chip komunikasi berupa USB to *serial*. Sehingga dalam pemrograman hanya membutuhkan kabel data USB [12].



Gambar 2.2 NodeMCU ESP8266

2.2.3 Sensor Ultrasonik HC-SR04

Sensor ultrasonik merupakan sensor yang bekerja berdasarkan pantulan gelombang suara dan berfungsi untuk mendeteksi keberadaan suatu objek yang berada di depan sensor tersebut. Sensor ini memiliki frekuensi kerja mulai 40 KHz sampai dengan 400KHz dan terdiri dari unit penerima dan unit pemancar. Struktur dari unit pemancar merupakan kristal *piezoelectric* yang dihubungkan dengan mekanik jangkar dan hanya dihubungkan dengan diafragma penggetar [13].



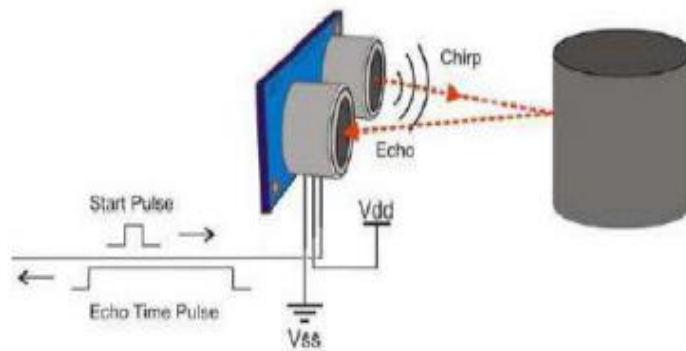
Gambar 2.3 Sensor Ultrasonik HC-SR04 Tampak Depan[13]



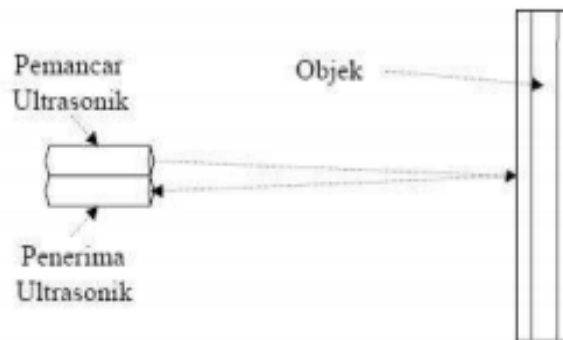
Gambar 2.4 Sensor Ultrasonik HC-SR04 Tampak Belakang[13]

Prinsip kerja dari sensor ini kontraksi yang terjadi diteruskan pada diafragma penggetar sehingga menimbulkan gelombang ultrasonik yang dipancarkan ke udara dengan arti tempat sekitarnya. Pantulan

gelombang ultrasonik tersebut akan terjadi apabila terdapat sebuah objek yang berada di depan sensor dan pantulan gelombang ultrasonik yang terjadi akan diterima kembali pada unit penerima. Kemudian pada unit penerima terjadi getaran pada diafragma penggetar dan efek *piezoelectric* akan menghasilkan tegangan bolak-balik dengan frekuensi yang sama. Prinsip kerja dan pemantulan sensor ultrasonik dapat dilihat pada gambar 2.5 dan 2.6



Gambar 2.5 Prinsip Kerja Sensor Ultrasonik[13]



Gambar 2.6 Prinsip Pemantulan Sensor Ultrasonik[13]

Untuk mengetahui presentase *error* pada sensor ultrasonik dapat digunakan metode persamaan untuk mencari tingkat akurasi dengan rumus :

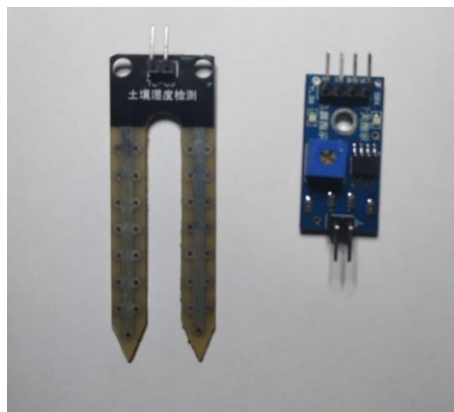
$$Error = \frac{Jp - Js}{Js} \times 100 \% \quad (2.1)$$

Dimana Jp merupakan tinggi pembacaan sensor dan Js merupakan tinggi air sebenarnya.

2.2.4 Sensor *Soil Moisture* YL-69

Soil Moisture Sensor YL-69 merupakan suatu sensor yang berguna dapat mendeteksi tingkat kelembaban didalam tanah. Sensor ini sederhana, tetapi memiliki fungsi yang ideal untuk memantau suatu area persawahan atau tingkat air pada tanaman perkebunan. Sensor yang terdiri dua *probe* ini berfungsi untuk melewatkan arus melalui tanah yang kemudian membaca resistansinya untuk memperoleh nilai tingkat kelembaban tanah tersebut [14]. Sehingga semakin banyaknya air mengakibatkan tanah lebih mudah untuk menghantarkan listrik (resistansi kecil), sedangkan sebaliknya apabila tanah tersebut kering mengakibatkan sulitnya dalam menghantarkan listrik (resistansi besar). Oleh karena itu, sensor ini sangat berguna mengetahui tingkat kelembaban pada tanaman maupun area irigasi persawahan atau perkebunan.

Spesifikasi pada sensor kelembaban suhu tanah memiliki tegangan *input* sebesar 3.3 *volt* atau 5 *volt*, tegangan *output* sebesar 0 sampai dengan 4.2 *volt*, arus sebesar 35 mA dan mempunyai *value range* ADC sebesar 1024 bit yang dimulai dari 0 sampai dengan 1023 bit dengan pin digital terdapat *high low output*, kemudian pin yang digunakan yaitu pin analog.



Gambar 2.7 *Soil Moisture* Sensor YL-69

2.2.5 Motor Servo

Merupakan suatu motor yang dapat diatur dan dikendalikan menggunakan pulsa. Motor ini terdiri dari beberapa komponen seperti sebuah motor DC, potensiometer, serangkaian *gear* dan rangkaian

kendali. Dimana potensiometer memiliki fungsi untuk membaca sudut digital *encoder* dari putaran servo dan motor servo ini beroperasi secara *close loop* yang kemudian poros motor dihubungkan pada rangkaian kendali, apabila putaran poros belum mencapai posisi yang diperintahkan ada coding maka rangkaian kendali akan memperbaiki posisi sampai diposisi yang diperintahkan [15].

Terdapat dua jenis motor servo yaitu motor servo *continuous* dan motor servo standar. Perbedaan dari kedua jenis ini yaitu pada motor servo standar hanya dapat bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga defleksi keduanya sebesar 180° , sedangkan motor servo *continuous* yang pada bagian *feedback*-nya dilepas sehingga dapat bergerak dua arah dengan tanpa batasan defleksi sudut putar mencapai sudut 360° .

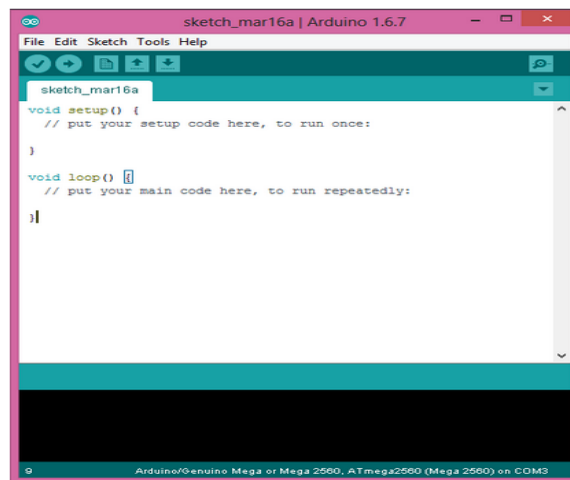
Motor servo standar ini mempunyai spesifikasi menahan posisi apapun antara 0° dan 180° , catu daya maksimal sebesar 6 VDC, Torsi sebesar 3.40 kg-cm, dikendalikan melalui *pulse-width modulation*, memiliki berat sebesar 44 gram, menerima empat sekrup pemasangan, diproduksi untuk *parallax* secara eksklusif oleh Futaba. Sedangkan motor servo *continuous* memiliki spesifikasi rotasi kontinyu 2 arah sampai 360, memiliki catu daya maksimal sebesar 6 VDC, kecepatan putar maksimal 50 RPM, dikendalikan melalui *pulse-width modulation*, memiliki berat 42.5 gram, menerima empat sekrup pemasangan, diproduksi untuk *parallax* secara eksklusif oleh Futaba. Sedangkan bentuk dari motor servo dapat dilihat pada gambar 2.9



Gambar 2.8 Motor Servo

2.2.6 Perangkat Lunak Arduino IDE

Software Arduino IDE (*Integrated Development Environment*) merupakan *software* yang berfungsi untuk membuat program pada Nodemcu ESP8266. Dalam arduino IDE ini terdapat *sketch*, *sketch* adalah program yang ditulis pada *software* ini. *Sketch* ditulis dalam suatu editor teks yang disimpan dalam *file* dengan ekstensi *.ino* [16]. Pada Arduino IDE ini terdapat fitur-fitur yaitu *message box* berwarna hitam yang digunakan untuk menampilkan status seperti *error message*, *compile* dan *upload* program, sedangkan *board* yang terkonfigurasi beserta *COM Ports* dapat ditemukan dibagian bawah kanan. Fitur *Compile/Verify* berfungsi untuk mengecek *sketch* yang telah dibuat ada atau tidaknya kekeliruan dalam segi sintaks atau tidak, apabila terdeteksi kesalahan dalam *sketch* ini maka sintaks yang dibuat akan *dicompile* kedalam bahasa mesin. Dan *upload* berfungsi untuk mengirimkan program yang telah *diverify* ke *Arduino board*. Gambar 2.9 dibawah ini merupakan tampilan *software* Arduino IDE

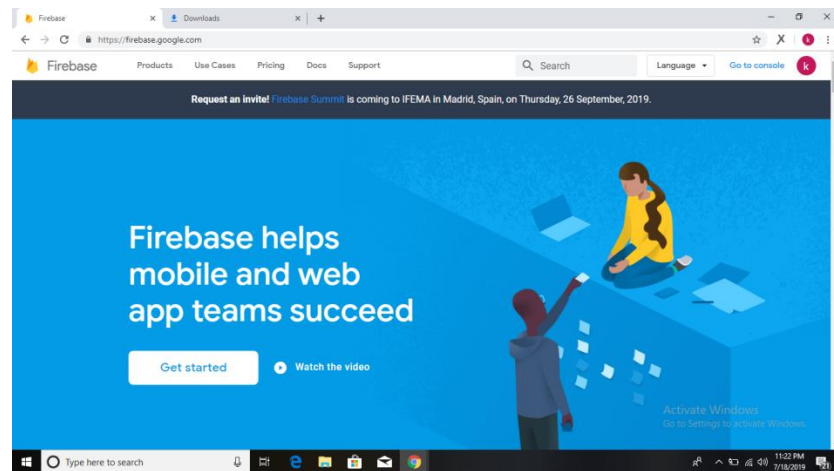


Gambar 2.9 Software Arduino IDE

2.2.7 Firebase Realtime Database

Firebase Realtime Database merupakan salah satu layanan yang dimiliki oleh Google dan merupakan *cloud database*. Dengan data yang disimpan dalam format *JSON* dan disinkronkan secara *realtime* ke setiap pengguna yang terhubung. Dengan data disimpan dalam format *JSON* dapat berbagi satu *instance Realtime Database* walaupun lintas

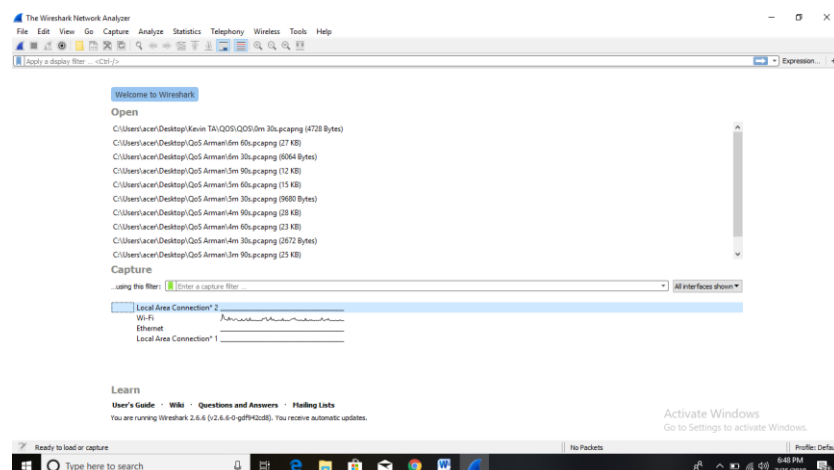
platform seperti Android dengan iOS dan secara otomatis menerima pembaharuan data [17].



Gambar 2.10 Tampilan *Firebase*

2.2.8 *Wireshark*

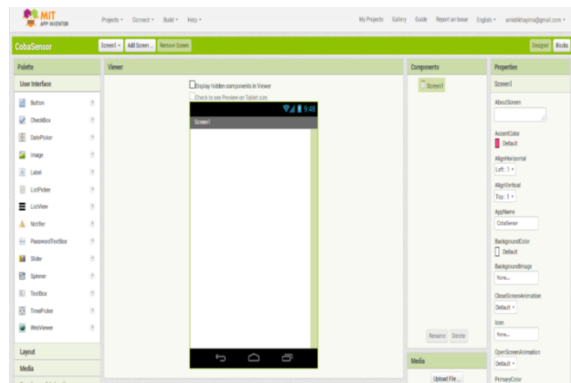
Wireshark merupakan *tool open source* yang memiliki fungsi untuk melakukan analisis dan memecahkan suatu masalah jaringan, penggunaan aplikasi ini memungkinkan untuk mengetahui adanya masalah di jaringan. *Wireshark* berfungsi pemeriksaan lanjut dari ratusan protokol, pengambilan data secara langsung dan melakukan analisis *offline*, *browser* paket tiga panel standar dan multi *platform* seperti : Windows, Linux, macOS, Solaris, FreeBSD, NetBSD dan banyak lainnya [18].



Gambar 2.11 Tampilan *Wireshark*

2.2.9 MIT App Inventor

MIT App Inventor merupakan sebuah *tool* untuk membuat aplikasi android sehingga memudahkan pengguna karena berbasis visual *block programming* jadi bisa membuat aplikasi tanpa kode satupun [16]. *Design View* terdiri dari *platte*, *viewer*, *component*, *properties* dan media. Fitur *Block* berfungsi untuk mengatur jalannya sebuah program yang dibuat. Fitur *design* berfungsi untuk merancang bentuk dari tampilan aplikasi yang akan dibuat.



Gambar 2.12 Tampilan MIT App Inventor

2.2.10 Smartphone

Smartphone merupakan telepon yang menyediakan berbagai macam fitur yang berada diatas dan diluar kemampuan sederhana sekedar hanya untuk membuat panggilan telepon [19]. *Smartphone* terus berkembang sebagai perangkat komunikasi maupun perangkat tangan menjadi lebih canggih.



Gambar 2.13 Smartphone

2.2.11 Quality of Service (QoS)

Quality of Service merupakan kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan menyediakan kapasitas jaringan, mengatasi *jitter* dan *delay*. QoS bertujuan membantu pengguna mengetahui kinerja dari aplikasi-aplikasi yang berbasis jaringan. QoS ini mengacu pada kemampuan suatu jaringan untuk menyediakan layanan yang maksimal pada trafik jaringan melalui teknologi yang berbeda-beda [20]. Sedangkan parameter-parameter QoS sebagai berikut :

a. *Delay*

Merupakan akumulasi waktu yang dibutuhkan untuk mengirim paket sampai ke tujuan. *Delay* mempengaruhi kualitas dari QoS karena menyebabkan waktu sampainya paket lebih lama mencapai tujuan. Rumus menghitung *delay* sebagai berikut :

$$\text{Rata - rata delay} = \frac{\text{Total delay}}{\text{Total paket yang diterima}} \quad (2.2)$$

Tabel 2.1 Standar Nilai *Delay* Berdasarkan ITU G.114

| <i>Delay</i> (ms) | Kualitas |
|-------------------|-----------------------------|
| 0-150 | Baik |
| 150-400 | Cukup, masih dapat diterima |
| > 400 | Buruk |

b. *Throughput*

Merupakan kecepatan mengirim data efektif yang diukur dalam *bit per second* (bps). Rumus menghitung *throughput* sebagai berikut :

$$\text{Throughput} = \frac{\text{Jumlah data yang dikirim}}{\text{Waktu pengiriman data}} \times 8 \quad (2.3)$$

c. *Packet Loss*

Bisa disebut paket yang hilang. Merupakan paket yang hilang pada saat melakukan pengiriman. Dapat diketahui dengan rumus sebagai berikut :

$$\text{Packet Loss} = \frac{\text{Paket data yang dikirim} - \text{paket data yang diterima}}{\text{paket data yang dikirim}} \times 100\% \quad (2.4)$$

Tabel 2.2 Standar *Packet Loss* Berdasarkan ITU G.114

| Packet Loss (%) | Kualitas |
|------------------------|----------------------|
| 0 | Baik |
| 1-5 | Dapat diterima |
| >10 | Tidak dapat diterima |