

# Perancangan Kendali Lampu Berbasis Android

Immanuel Warangkiran<sup>(1)</sup>, Ir. S.T.G Kaunang, MT.<sup>(2)</sup>, Arie. S.M Lumenta, ST, MT.<sup>(3)</sup>, Arthur. M Rumagit ST, MT.<sup>(4)</sup>

(1)Mahasiswa (2)Pembimbing 1 (3)Pembimbing 2 (4)Pembimbing 3

nuel\_other@yahoo.co.id<sup>(1)</sup> s.t.g.kaunang@unsrat.ac.id<sup>(2)</sup> arie.lumenta@gmail.com<sup>(3)</sup>

arthur\_rumagit@unsrat.ac.id<sup>(4)</sup>

Jurusan Teknik Elektro-FT, UNSRAT, Manado-95115

## Abstract

*Smartphone with android operation system are getting more available in the markets with quite low price. Android operation system it self has open source operation system which can be modified. This boosts the interest of creating software to fill in the needs of the people in everyday life.*

*The aim of this last project is to create and implimate to control lamps that can be switch on or off from a smartphone based on android by using wireless LAN and to make android as multifunction cellular, not only as telecommunication device but as a device that can control any hardware.*

*Based on the final experiment the system work well. The application lamps control can be applied in the operation system android, and can be controlled to switch on or switch off the hard drives in this case three lamps. The communication between the hard drives lamps and the smartphone android worked well. The range of the application system can control the hard drives lamps, it depends on the specification of the wireless tools between the receiver and the sender.*

**Keywords:** *Android, Lamp, Smartphone, Wireless LAN.*

## Abstrak

*Smartphone dengan sistem operasi Android semakin banyak tersedia di pasaran dengan harga yang semakin terjangkau. Sistem operasi Android sendiri bersifat sistem operasi open source yang dapat dimodifikasi sesuai dengan keperluan. Hal ini menumbuhkan minat untuk dapat membuat perangkat lunak yang bermanfaat dalam memenuhi kebutuhan manusia sehari-hari.*

*Dalam Tugas Akhir ini bertujuan merancang dan mengimplementasikan, yaitu mengontrol lampu, mengaktifkan dan menonaktifkannya lewat smartphone berbasis android dengan memanfaatkan Wireless LAN dan menjadikan android sebagai perangkat selular yang multifungsi, di samping alat komunikasi tapi juga sebagai perangkat yang dikomunikasikan untuk mengendalikan sebuah perangkat keras.*

*Berdasarkan hasil pengujian yang dilakukan, sistem dapat bekerja dengan baik sesuai dengan tujuan. Program aplikasi Lampu Control yang dibuat dapat dijalankan di sistem operasi Android, dan dapat mengaktifkan dan menonaktifkan perangkat driver lampu, dimana memakai 3 buah lampu. komunikasi data antara perangkat driver lampu dengan smartphone android berjalan dengan baik, Jarak jangkauan sistem aplikasi kendali dapat mengontrol perangkat driver lampu tergantung spesifikasi perangkat wireless yang digunakan baik dari sisi penerima dan pengirim.*

**Kata kunci:** *Android, Lampu, Smartphone, Wireless LAN.*

## I. PENDAHULUAN

Ponsel pintar dengan sistem operasi *Android* semakin banyak tersedia di pasaran dengan harga yang semakin terjangkau. Sistem operasi *Android* sendiri bersifat sistem operasi *open source* yang dapat dimodifikasi sesuai dengan keperluan. Hal ini menumbuhkan minat untuk dapat membuat perangkat lunak yang bermanfaat dalam memenuhi kebutuhan manusia sehari-hari.

Dengan memanfaatkan perangkat keras yang sudah terdapat di ponsel *Smartphone Android* berupa pengaturan *wifi* maka dapat membuat aplikasi berdasarkan fungsi tersebut. Selain itu dalam proses pembuatan aplikasi tentunya tidak berbayar dan dapat dilakukan dengan bebas atau *open source* dan memiliki tampilan aplikasi yang bisa dimengerti pemakaiannya oleh pengguna.

Berdasarkan penelitian dan studi kasus yang dilakukan dalam pembuatan tugas akhir ini, ada beberapa pokok pikiran yang menjadi acuan sehingga penulis membuat tugas akhir Perancangan Kendali Lampu Berbasis *Android*.

## II. LANDASAN TEORI

### A. Ponsel Pintar

Ponsel pintar adalah perkembangan dari *handphone* yang ditambahi fitur - fitur seperti pada personal komputer fitur - fitur seperti *email*, *personal organizer*, dan juga konektivitas tambahan seperti *wifi* dan *bluetooth* yang dapat diinstall di *device*. Dari segi arsitektur *device* sendiri sudah dilengkapi dengan inputan seperti QWERTY miniatur *keyboard* dan *touchscreen*.

Aplikasi pada ponsel pintar dikembangkan oleh operator dari *device* itu sendiri ataupun pihak ketiga yang ikut mengembangkan untuk kepentingan komersial.

Ponsel pintar pertama diberi nama *Simon* yang dikembangkan oleh IBM pada tahun 1992 dan terpilih sebagai *product of the year* oleh COMDEX. *Simon* direlease pada tahun 1993 oleh *BellSouth*, selain fitur *telephone* dan SMS *Simon* dilengkapi dengan *calendar*, *address book*, *world clock*, *notepad*, *e-mail*, *fax*, dan *games*. Setelah itu banyak prodak sejenis yang dikeluarkan oleh berbagai *vendor* berbeda seperti *Nokia*.

Sistem operasi yang digunakan pada ponsel pintar berbeda-beda tetapi yang paling banyak digunakan saat ini adalah sistem operasi yang berbasis *Android* dari *google*.

#### B. Pengembangan mobile aplikasi

*Mobile* aplikasi adalah aplikasi pada perangkat atau *device* yang dapat dibawa kemanapun seperti *handphone*, ponsel pintar, PDA, dll. Aplikasi yang ditawarkan dapat bermacam – macam seperti *office based application* ( word, excel, dll) , *accounting application*, *game* , dll.

Seiring perkembangan dari dunia telekomunikasi khususnya dibidang jaringan banyak aplikasi mobile berbasis *wireless networking*. Aplikasi ini memanfaatkan infrastruktur jaringan yang ada di *device* tersebut seperti *wifi*, *bluetooth*, *infrared*, ataupun lewat GPRS. Dengan begitu memudahkan kita untuk melakukan komunikasi selain lewat jaringan GSM ataupun CDMA dari para operator.

#### C. Arsitektur wireless aplikasi mobile

Sejak diperkenalkannya teknologi paket data di atas jaringan *wireless* (seperti GPRS), memberikan banyak peluang baru kepada pengembang aplikasi maupun *content developer* untuk mengembangkan berbagai aplikasi layanan komunikasi bergerak. Bagi para pengembang aplikasi, yang sangat dibutuhkan sekarang adalah adanya kesepakatan mengenai arsitektur terbuka dan tersedianya *Application Programming Interfaces* (API). 3G adalah teknologi yang dirumuskan berdasarkan kesepakatan para ahli yang berkompeten di bidangnya. Organisasi *International Telecommunication Union* (ITU) telah mengeluarkan rekomendasi sistem 3G yang tertuang dalam *International Mobile Telecommunications 2000* (IMT-2000).

Dunia telah mendapat pelajaran dari masa lalu dengan diterapkannya arsitektur pengembangan teknologi dan aplikasi yang bersifat tertutup dan tidak fleksibel serta hanya memungkinkan kesempatan untuk pengembangan teknologi.

Dengan arsitektur yang bersifat vertikal seperti di atas, pengembangan teknologi hanya dapat dilakukan pada teknologi yang bersangkutan dan tidak dapat diterapkan (tidak kompatibel) pada teknologi lain. Hal inilah yang menyebabkan teknologi dan aplikasi yang berkembang bersifat *vendor-technology oriented* yang terutama terjadi di perusahaan besar yang ingin menciptakan pasar sendiri terhadap pelanggan yang membutuhkan produknya. Sebagai contoh, di masa lalu pelanggan dari jaringan CDMA tidak dapat mengakses SMS dari jaringan GSM, dan teknologi paket data CDPD hanya dapat dijalankan di jaringan TDMA (D-AMPS), atau juga pelanggan telepon *fixed-line* memiliki mesin penjawab yang berbeda dengan *voice-mail* yang terdapat pada telepon seluler. Dengan model arsitektur demikian, pengembang aplikasi dari pihak ketiga juga akan sangat sulit untuk berperan dan ikut bermain. Keterbatasan ini menjadi lebih terasa pada saat

sekarang dimana internet telah dikembangkan dengan arsitektur terbuka.

#### D. Android

Menurut Safaat Nazruddin (2011) *Android* adalah sistem operasi untuk telepon seluler yang berbasis Linux. *Android* menyediakan *platform* yang bersifat *open source* bagi para pengembang untuk menciptakan sebuah aplikasi. Awalnya, Google Inc. mengakuisi *Android Inc.* Yang mengembangkan *software* untuk ponsel yang berada di Palo Alto, California Amerika Serikat. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, yaitu konsorsium dari 34 perusahaan *hardware*, *software*, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Telepon pertama yang memakai sistem operasi *Android* adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis.

Dari segi arsitektur sistem, *Android* merupakan sekumpulan *framework* dan *virtual machine* yang berjalan di atas kernel linux. *Virtual machine Android* bernama *Dalvik Virtual Machine (DVM)*, *engine* ini berfungsi untuk menginterpretasikan dan menghubungkan seluruh kode mesin yang digunakan oleh setiap aplikasi dengan *kernel linux*. Sementara untuk *framework* aplikasi sebagian besar dikembangkan oleh google dan sebagian yang lain dikembangkan oleh pihak ketiga (*developer*). Beberapa *framework* yang dikembangkan oleh *Android* sendiri misalnya fungsi untuk telepon seperti panggilan telepon, sms, *video call*. Untuk *browser Android* menggunakan *google chrome*.

Aplikasi yang dikembangkan *Android* dibuat dengan menggunakan kode *Java* seperti halnya *J2ME* yang telah lama digunakan pada *platform* perangkat selular umumnya. Namun secara siklus program memiliki perbedaan mendasar antara *J2ME* dengan *Java* yang ada pada *Android*. Kode *Java* pada *Android* lebih dekat dengan *J2SE*. hingga saat ini *Android* telah banyak digunakan pada produk *smartphone* seperti Samsung, LG, SonyEricson, Nexian dan juga HTC. Dengan dukungan *Software Development Kit (SDK)* dan *Application Programming Interface (API)* dari google memberikan kemudahan bagi pihak ketiga (*developer*) untuk membangun aplikasi yang dapat berjalan pada sistem operasi *Android*.

Selain itu terdapat metode baru dalam mengembangkan aplikasi di dalam sistem operasi *Android* menggunakan *Native Development Kit (NDK)*. *NDK* ini memungkinkan *developer* untuk mengembangkan aplikasi di dalam sistem operasi *Android* menggunakan bahasa pemrograman *C* atau *C++*.

Menurut King C, Ableson (2011) *Android* memiliki empat komponen. Meliputi *activity*, *Broadcast Receiver* , *service* dan *content provider*. Komponen aplikasi dapat disebut juga sebagai elemen-elemen aplikasi yang bisa dikembangkan pada platform *Android*. Perlu diketahui bahwa untuk membangun sebuah aplikasi pada *Android* bahasa pemrograman yang digunakan adalah *Java*, seperti halnya *J2ME* aplikasi pada perangkat selular, *Java* yang dikembangkan pada

sistem operasi *Android* memiliki struktur yang berbeda. Program aplikasi yang dikembangkan pada *Android* tidak memiliki fungsi *main*. Karakteristik lain adalah, bahwa semua aplikasi di *Android* dapat menggunakan *object* yang dibangun oleh aplikasi yang lain. Contoh, jika ingin menggunakan sebuah *object scrollbar*, tidak harus membuatnya sendiri, namun bisa saja memanggil *object* yang berada pada aplikasi lainnya.

#### Activity

*Activity* merupakan bagian yang paling umum dari empat komponen *Android*. Suatu *activity* yang biasanya satu layar dalam aplikasi pengguna. Setiap *activity* diimplementasikan sebagai satu class yang memperluas dasar kelas *activity*. Kelas akan menampilkan *user interface* yang terdiri dari *views* dan merespon kejadian. Kebanyakan aplikasi terdiri dari beberapa layar. Sebagai contoh, sebuah aplikasi pesan teks mungkin memiliki satu layar yang menunjukkan daftar kontak untuk mengirim pesan, layar kedua untuk menulis pesan ke kontak yang dipilih dan layar lain untuk memeriksa pesan lama atau pengaturan. Masing-masing layar akan diimplementasikan sebagai suatu *activity*. Berpindah ke layar lain berarti memulai aktivitas baru. Dalam beberapa kasus suatu *activity* dapat mengembalikan nilai ke *activity* sebelumnya, misalnya *activity* yang memungkinkan pengguna mengambil foto akan kembali pada foto yang dipilih ke pemanggil. Ketika layar baru akan dibuka, layar sebelumnya akan berhenti dan diletakkan di *history stack*. Pengguna dapat menavigasi mundur melalui layar sebelumnya dalam *history*.

#### Content Provider

Sebuah *content provider* mengatur sekumpulan data aplikasi yang terbagi (*shared*). Kita bisa menyimpan data di file sistem, sebuah *database SQLite*, di *web*, atau di metode penyimpanan data lainnya yang bisa diakses oleh aplikasi kita. Melalui *content provider*, aplikasi lain bisa memberikan *query* atau bahkan bisa memodifikasi, tentunya jika *content provider* mengizinkan aksesnya. Sebagai contoh, sistem *Android* menyediakan *content provider* yang mengatur informasi kontak *user*. Misalnya, aplikasi apa saja yang memiliki ijin bisa memberikan *query* kepada sebagian data untuk membaca dan menulis informasi tentang orang tertentu. *Content provider* juga bisa digunakan untuk menulis dan membaca data pribadi yang tidak dibagikan .

#### E. Wireless LAN

Menurut Arifin Zaenal (2005) Teknologi *wireless LAN* melakukan proses pengiriman data dengan menggunakan frekuensi radio sebagai media perantaranya. Teknologi ini diregulasi oleh aturan yang sama seperti radio AM/FM. *Federal communication commission* (FCC) merupakan organisasi internasional yang meregulasi penggunaan *device wireless LAN*. Sebaliknya, IEE (*Institute of Electrical & Electric Engineers*) membuat dan mengelola standarisasi *device wireless*.

Ada tiga pita (*band*) frekuensi yang dapat digunakan secara bebas dalam dunia industri, medis, dan ilmiah, antara lain frekuensi 900 MHz, 2,4 GHz, dan 5,2

GHz. Di antara ketiga band, perangkat-perangkat *wireless* saat ini banyak menggunakan frekuensi 2,4 GHz.

### III. METODOLOGI PENELITIAN

#### A. Tempat dan Waktu Penelitian

Dalam pelaksanaan tugas akhir ini penulis mengambil tempat pada Ruang Laboratorium Sistem Komputer (LSK) Jurusan Teknik Elektro Universitas Sam Ratulangi, perpustakaan Fakultas Teknik UNSRAT dan di rumah tinggal penulis. Waktu dan lama perancangan sampai pembuatan berlangsung selama  $\pm 4$  bulan, dimulai dari awal bulan Maret 2012 sampai akhir bulan juli 2012.

#### B. Bahan dan Peralatan

Pada perancangan sistem ini, dibutuhkan beberapa Alat, bahan, serta program aplikasi pendukung, yang dikelompokkan menjadi 2 bagian, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*).

#### Hardware

Laptop, Wifi210, Arduino Uno, Tablet Android Aedupac, Kabel USB, Balon Lampu.

#### Software

Sistem Operasi: Windows 7, Microsoft Visio, IDE Arduino , IDE Eclipse, Putty, Notepad, Sun Java 1,6, SDK android.

#### C. Prosedur Penelitian

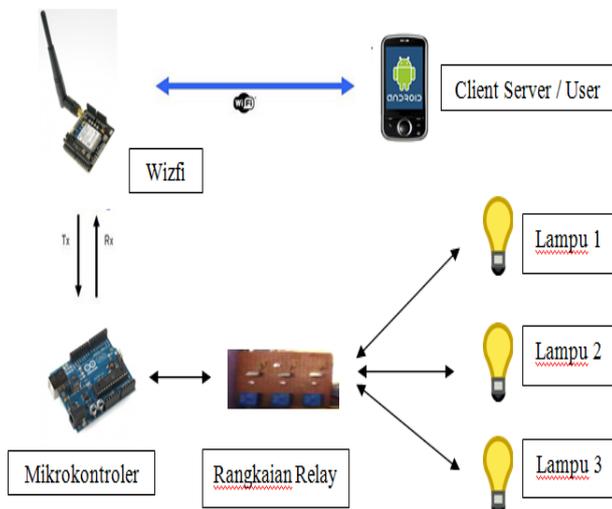
Prosedur yang dilakukan dalam perancangan kendali lampu berbasis *android* adalah sebagai berikut:

Sebelum melakukan penelitian, penulis terlebih dahulu melakukan studi literatur. Penulis mencari materi-materi yang berhubungan dengan pembuatan kendali lampu.

Setelah mendapatkan informasi yang dibutuhkan, maka penulis mencari program-program pendukung dan perangkat keras dalam pembuatan tugas akhir. Melakukan perancangan *hardware* dan perancangan program aplikasi, setelah itu mengadakan pengujian sistem.

#### D. Desain Sistem

Sistem terdiri dari dua bagian besar, yaitu sebuah aplikasi perangkat lunak yang akan digunakan untuk sistem pengontrolan berbasis *android* dan sebuah sistem perangkat keras yang berperan dalam sisi mekanisme sistem. Aplikasi perangkat lunak dibuat dengan menggunakan program aplikasi *eclipse* yang memungkinkan *user* untuk dapat mengakses tombol-tombol yang digunakan untuk mengendalikan, menyalakan atau mematikan perangkat listrik berupa lampu dari *smartphone* yang memiliki sistem operasi *android*. Sistem perangkat keras menggunakan beberapa komponen penting yaitu sebuah *PC/Laptop*, sebuah mikrokontroler, *wifi*, rangkaian *relay*, dan sebuah *smartphone android*. Sistem ini (gambar 1) berkerja dimana *smartphone* yang berisikan sistem operasi *android* dikendalikan untuk menyalakan tiga lampu.



Gambar 1. Diagram Sistem

Pada *smartphone android* di install aplikasi yang sudah di buat untuk mengendalikan, mengaktifkan atau menonaktifkan, setelah itu dihubungkan dengan jaringan *local wireless* yang ada di *wizfi*. *Wizfi* dan mikrokontroler di setting melalui *PC/Laptop* agar dapat berhubungan dengan *smartphone android* melalui jaringan *local wireless*, dan juga bisa berhubungan dengan rangkaian *relay* untuk mengendalikan, menyalakan / mematikan perangkat listrik berupa lampu.

#### E. Perancangan sistem Hardware.

Bagian *hardware* khususnya pada rangkaian *device*, mikrokontroler dan *wizfi* berperan penting dalam menerima input dari *server*, dan mengirimkan perintah tersebut pada rangkaian *relay* sesuai dengan perintah yang dikirimkan oleh *server*.

Sistem ini menggunakan *device* controller berupa rangkaian *integrated circuit*, menggunakan mikrokontroler *arduino uno*, *wizfi* dan rangkaian *relay* lampu. Mikrokontroler berperan dalam mengirimkan perintah dan untuk menyelaraskan tegangan yang keluar dari USB port agar setara dengan tegangan yang diperlukan oleh rangkaian *relay* lampu. Mikrokontroler ini juga berfungsi sebagai otak dari sistem kendali. Dalam rangkaian mikrokontroler *Arduino uno* ini mempunyai 14 pin digital input/output diantaranya 6 pin (dapat digunakan sebagai *PWM output*), 6 pin *analog*. *Wizfi* digunakan sebagai jembatan untuk dapat menghubungkan mikrokontroler, rangkaian *relay* lampu, dan *user*.

Pada sistem kali ini terlebih dahulu mikrokontroler dan *wizfi* dihubungkan. setelah keduanya terhubung, pin 5, 6, 7 pada *wizfi* digunakan untuk menghubungkan *wizfi* dengan rangkaian *relay* lampu. Pin 5, 6, 7 adalah sebagai output dari *wizfi* dimana ketiga pin yang tersebut, dipakai untuk mengendalikan tiga *relay* yang ada pada rangkaian, untuk menyalahkan dan mematikan lampu.

#### F. Perancangan Program Aplikasi.

Pada perancangan sistem *server* ini menggunakan hardware dengan spesifikasi *processor* AMD E-350 1,6GHz, memory 2 GB, dan beberapa komponen *standard* lainnya. Selain itu sistem ini

menggunakan beberapa *software* aplikasi yang di *download* secara gratis untuk mendukung kinerja sistem ini. Berikut adalah *software – software* yang dibutuhkan dalam perancangan sistem *server* ini :

#### Sistem Operasi.

Dalam perancangan sistem ini, sistem operasi yang digunakan adalah *Microsoft Windows 7*. Selain sistem operasi ini sudah dikenal luas oleh *server* pada umumnya, sistem ini dapat diandalkan karena kompatibel dengan aplikasi pendukung lainnya seperti : *eclipse*, *youwave*, *putty*, *android 1.0*.

#### Eclipse Galileo.

*Eclipse galileo* adalah program aplikasi yang dapat digunakan secara gratis dan legal. Dalam sistem ini *Eclipse Galileo* harus di install terlebih dahulu kemudian menambahkan beberapa aplikasi penunjang program ini didalamnya. *Eclipse Galileo* ini merupakan program aplikasi yang di gunakan untuk membuat aplikasi baik yang digunakan oleh *user*.

#### Android 1.0.

*Android 1.0* adalah program aplikasi yang didapat bersamaan dengan pembelian mikrokontroler *adruino*, ataupun bisa di *download* di *internet* secara gratis. Program aplikasi ini digunakan untuk membuat program pada mikrokontroler.

#### Putty.

*Putty* adalah *software remote console* atau terminal yang digunakan untuk meremote komputer dengan terhubungnya menggunakan *Port SSH* dan sebagainya.

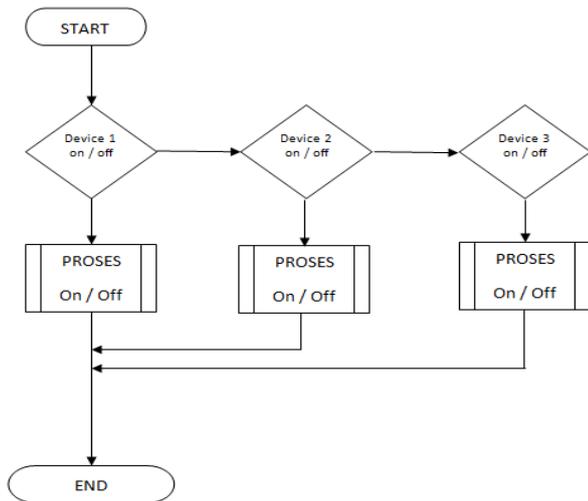
#### Youwave.

*Youwave* adalah program aplikasi yang digunakan untuk menjalankan aplikasi *android* di PC atau di laptop. *Youwave* biasa disebut juga sebagai *emulator android*. Kegunaannya dalam perancangan sistem ini adalah untuk menguji apakah program aplikasi *android* yang dibuat bisa berjalan dengan baik, dan bisa digunakan oleh *user*.

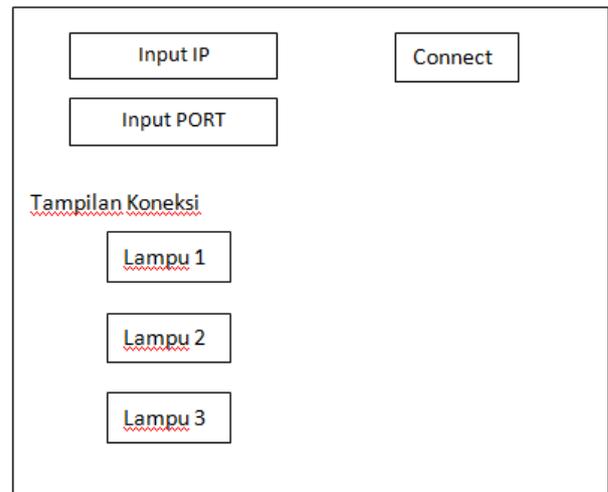
#### G. Algoritma Program.

Program yang nantinya akan di akses oleh *user* dapat direpresentasikan oleh *flowchart* pada gambar 2 *flowchart* algoritma program .

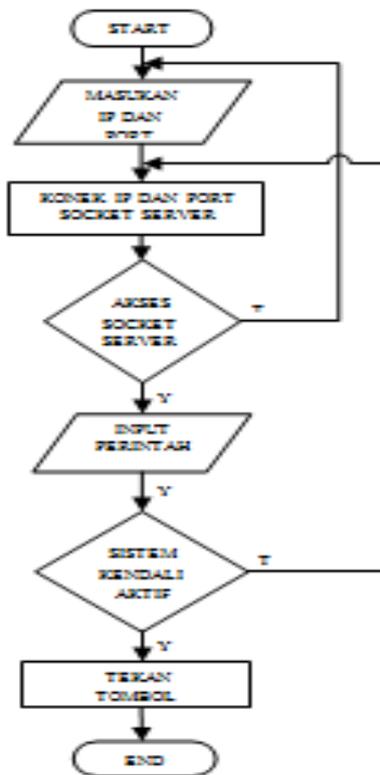
Dari *user* pertama (gambar 3) akan diarahkan ke halaman utama. Kemudian halaman utama akan menyajikan tampilan yang akan di akses dan tombol – tombol perintah memasukan *IP* dan *PORT* selanjutnya perintah mengaktifkan perangkat listrik berupa lampu. *user* akan diberikan kebebasan untuk mengaktifkan atau menonaktifkan perangkat tersebut. Jika *user* memilih untuk menekan tombol pertama kali maka program akan mengeksekusinya dengan menyalahkan / mengaktifkan perangkat listrik. Apabila *user* menekan kedua kalinya pada tombol tersebut maka program akan mengeksekusinya dengan mematikan / menonaktifkan.



Gambar 2. Flowchart Algoritma Program.



Gambar 4. Tampilan Utama Lampu Control.

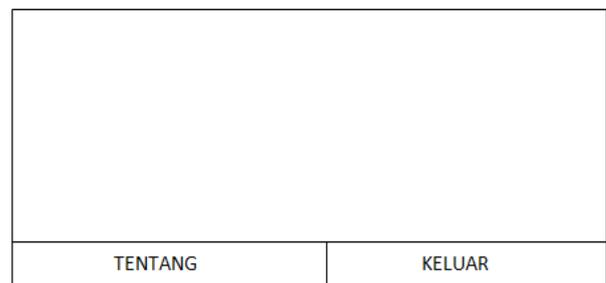


Gambar 3. Flowchart Program Aplikasi.

H. Perancangan Antar Muka Aplikasi Kendali Lampu Berbasis Android.

Aplikasi ini diberi nama lampu control (gambar 4). Secara Umum Arsitektur User Interface (UI) pada aplikasi android adalah user interface yang meliputi Activity dan user interface yang terdiri dari komponen. Semua yang berhubungan dengan user interface pada aplikasi android biasanya berada pada lokasi res/layout/filename.xml. dimana coding java untuk memanggilnya yang kita kenal dengan R.Layout.filename.

Pada option menu (gambar 5) terdapat dua pilihan yaitu (tentang) dan (keluar). Ketika kita memilih tentang maka ditampilkan info dari aplikasi android ini.



Gambar 5. Tampilan Menu Lampu Control.

IV. PENGUJIAN DAN PEMBAHASAN

Tugas akhir yang dibuat adalah perancangan kendali lampu berbasis android.

A. Pendukung Sistem

Untuk menyelesaikan tugas akhir ini digunakan beberapa pendukung sistem yang terdiri dari software dan hardware.

Software

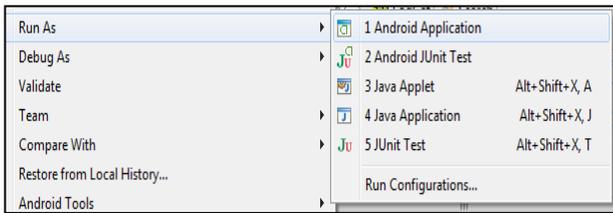
Dalam pembuatan perancangan kendali lampu berbasis android. digunakan aplikasi program utama yaitu IDE Eclipse dan beberapa program penunjang yaitu IDE arduino sebagai tool untuk pemrograman dan konfigurasi mikrokontroler dan wizfi, serta ponsel dengan sistem operasi android 2.2 (froyo).

Hardware

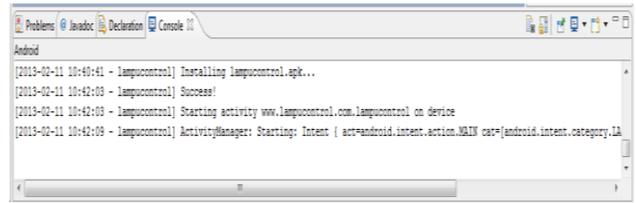
Selain software yang digunakan pada pembuatan sistem ini, juga digunakan hardware untuk mengoperasikan sistem ini. Hardware yang digunakan adalah rangkaian relay lampu yang dihubungkan dengan mikrokontroler dan wizfi.

B. Instalasi Program Aplikasi Lampu Control Pada Perangkat Ponsel Android.

Aplikasi ini dapat di install di sistem android.



Gambar 6. Kompilasi aplikasi android.



Gambar 8. Tampilan Activity Console Emulator.



Gambar 7. Tampilan Pada Emulator Android.



Gambar 9. Aplikasi diinstall di emulator.

Proses instalasi aplikasi android sistem pengendali yang dibuat di IDE *eclipse* yaitu melalui instalasi langsung terdiri dari teknik *package file* dan *debugging* (gambar 6) langsung ke perangkat *smartphone* berbasis *android* kemudian melalui *emulator* atau *android virtual device* (AVD) (gambar 7).

Instalasi langsung ke perangkat *android* terlebih dahulu melakukan *package file* dengan menyertakan *certificate* tujuannya yaitu sebagai legalitas dan bukti bahwa aplikasi dibangun/develop sendiri. *Certificate* disini adalah kunci yang didapatkan dari google. file yang dihasilkan berekstensi \*.apk (*android package kit*). File \*.apk itulah yang disebut dengan aplikasi *android*, dan nantinya dapat diinstall di perangkat *smartphone android*.

Teknik *debugging* dilakukan setelah USB *debugging* pada perangkat *android* sudah aktif dengan cara menghubungkan perangkat melalui kabel USB dengan Komputer kemudian pada IDE *eclipse* aplikasi *project* dikompilasi dan dijalankan, proses ini tanpa memerlukan AVD lagi.

Instalasi melalui *emulator android* bertujuan pengembang aplikasi dapat melihat hasil dari aplikasi yang dibuat tanpa perlu adanya perangkat *smartphone android*.

Gambar 8 dan gambar 9 menunjukkan bagian proses instalasi program aplikasi pada *emulator android*. Seluruh proses instalasi ke dalam sistem operasi *android* terlihat jelas pada tampilan *console* yang mendeskripsikan apa-apa yang terjadi sehingga aplikasi dapat berjalan di *smartphone* dari tahapan pengaktifan *adb* (*android debugging bridge*) sampai pada *uploading* dan *installing* aplikasi pada perangkat sistem *android*.

### C. Tampilan Program.

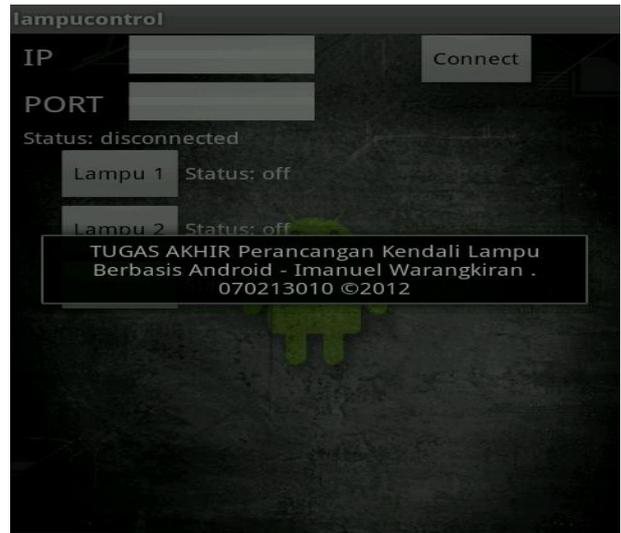
Tampilan utama (gambar 10) aplikasi ini terdiri dari *editText* untuk mengisi *IP address* dan *PORT*, perintah *ImageButton* untuk menyalakan dan mematikan lampu dan tombol status untuk memulai koneksi. Dalam pengisian alamat *IP user* harus mengisi alamat IP 192.168.1.1 dan *PORT* 3000 (gambar 11) karena Aplikasi ini hanya terkoneksi dengan Server yang mempunyai alamat IP 192.168.1.1 dan *PORT* 3000. Setelah diisi dengan benar kemudian tekan tombol *connect* maka aplikasi siap untuk mengirim *Stream command* ke *socket server*. gambar 12 dan Gambar 13 merupakan antarmuka menu dari aplikasi lampu control. ada 2 pilihan menu yaitu (Tentang) dan (Keluar). Fungsi dari menu (Tentang) adalah menampilkan nama dan identitas aplikasi dan pembuat sedangkan dari menu (Keluar) adalah menutup/mengakhiri Aplikasi.

### D. Pengujian Sistem.

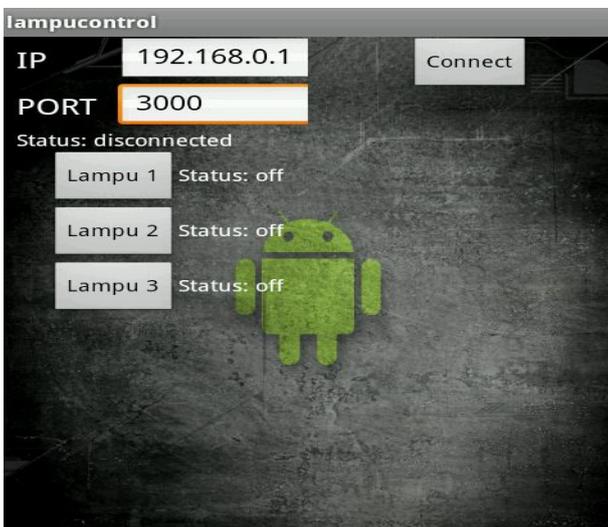
Pada awalnya *server* harus diaktifkan beserta dengan *device* kontroler yang telah terhubung ke *server*, dan terhubung juga ke sumber tegangan. *Server* tersebut sudah di setting sebagai *access point* sehingga dapat di akses oleh *user* lewat jaringan *local wireless*. Selanjutnya *user* dapat mengontrol / menyalakan dan mematikan lampu lewat program lampu control yang telah dibuat, dan yang telah terinstall di *laptop* (gambar 14) atau di perangkat *smartphone android* (gambar 15). Setelah *user* masuk dalam program lampu control yang ada di *laptop* atau *smartphone android*. *user* akan diperhadapkan langsung dengan halaman utama program aplikasi lampu control, dimana pada halaman ini *user* mengisi alamat *IP* dan *PORT* setelah itu mengkoneksikan lewat *wireless*.



Gambar 10. Tampilan Utama Aplikasi.



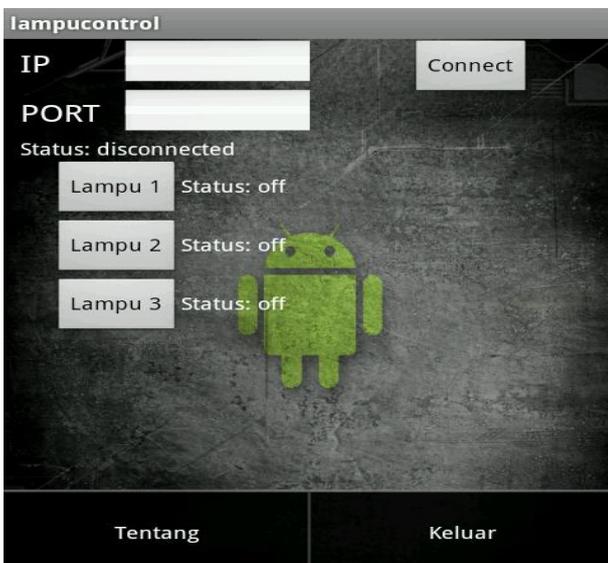
Gambar 13. Tampilan Tentang.



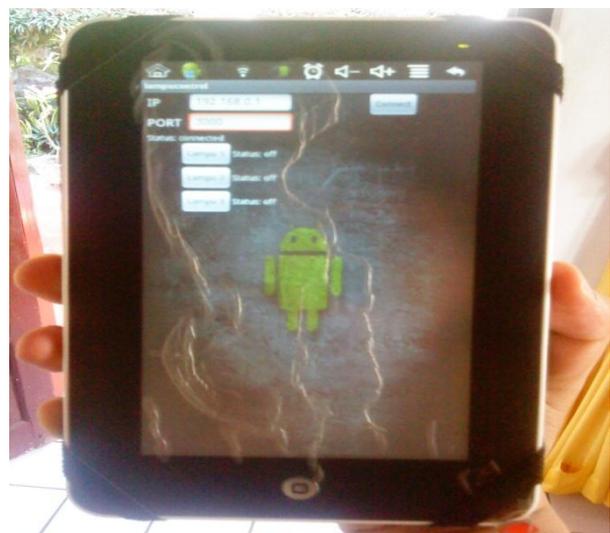
Gambar 11. Tampilan Masukan IP Dan Port.



Gambar 14. Tampilan Program dari Laptop.



Gambar 12. Tampilan Menu.



Gambar 15. Tampilan Program Dari Smartphone Android.

### E. Analisa Hasil

Kemampuan sistem kendali lampu *control* menggunakan *smartphone android* dengan komunikasi data menggunakan jaringan *wireless* dalam penelitian ini berjalan dengan baik, aplikasi dapat digunakan di *platform smartphone* yang mempunyai sistem operasi *android*. Dalam pengontrolan dapat menggunakan / menekan tombol navigasi pada aplikasi lampu *control*.

Aplikasi hanya dapat berfungsi apabila dari sisi perangkat *driver* lampu *wifi* sudah terinstal *socket server* dan pada perangkat *smartphone android* terinstal *client socket* aplikasi pengendali lampu (*lampucontrol*). Dari hasil pengujian jarak, kondisi tempat, dan spesifikasi perangkat *wireless* yang digunakan sangat berpengaruh untuk terjadinya proses komunikasi data dengan baik . Jarak jangkauan kendali maksimum perangkat *driver* lampu *wifi* pada kondisi didalam ruangan hanya mencapai 30 meter sedangkan pada kondisi diluar ruangan mencapai 50 meter.

Dalam pembuatan sistem aplikasi pengendali ini ditemukan banyak hal yang dapat dikembangkan. Sinkronisasi komunikasi data merupakan persoalan tersendiri dari pemrograman sistem ini serta kemampuan membaca data dari komunikasi *serial port* perangkat *wireless* dengan mikrokontroler dan perangkat *driver*.

### V. KESIMPULAN

Dengan memanfaatkan *smartphone android* sistem ini berhasil mengendalikan perangkat *driver* lampu melalui koneksi jaringan *wireless*. Kondisi sinyal pada *wifi* mempengaruhi komunikasi *transfer* data pada aplikasi pengendali yang dibuat, Jarak jangkauan sinyal *wifi* tergantung pada spesifikasi dari perangkat *wifi* dari pengirim atau penerima data. Dengan memanfaatkan teknologi pada perangkat *wireless wifi 210* dan mikrokontroler *arduino uno*, hasil penelitian ini didapat bahwa perangkat ini dapat digunakan untuk mengontrol, mengaktifkan dan menonaktifkan perangkat lampu.

### DAFTAR PUSTAKA

- [1] A. Murat. "New Features For Application Development", Packt Publishing Ltd, 2012.
- [2] A. Zaenal. "Mengenal Wireless LAN", Penerbit Andi, Yogyakarta, 2005.
- [3] B. Heriyanto "Esensi – esensi Bahasa Pemrograman Java", Penerbit Informatika, Bandung, 2011.
- [4] H. Wowiling. "Aplikasi Pengendali Robot Menggunakan Jaringan Wireless Pada Ponsel Pintar Android", Skripsi, Universitas Sam Ratulangi Fakultas Teknik, Jurusan Teknik Elektro, Manado, 2012.
- [5] I. Siregar, ST, MT. "Membongkar Source Code Berbagai Aplikasi Android", Penerbit Gava Media, Yogyakarta, 2012.
- [6] N. Safaat. "Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android", Penerbit Informatika, Bandung, 2011.