



Design of Wheeled Football Robot Coordination System at Base Station Using TCP / IP

Perancangan Sistem Koordinasi Robot Sepak Bola Beroda Pada Base Station Menggunakan TCP/IP

Diah Arie Widhining Kusumastutie¹⁾, Farrady Alif Fiolana²⁾

^{1,2)} Electrical Engineering Study Program Faculty Engineering, Kadiri Islamic University Kediri, Kediri, Indonesia

¹⁾diahariewk@uniska-kediri.ac.id

²⁾farradyalif@uniska-kediri.ac.id

Abstract. The Indonesian Robot Contest on the Wheeled Football division is an annual competition held by the ministry of education. One team in this division consists of 3 robots connected in wireless communication coordinated by a base station computer. In addition, the base station also connects the three robots with the referee's computer or called the RefBox (Referee Box). RefBox computer as a server for the base station and base station as a server for robots. The problem arises at the base station which has to play a dual role, as a server and client. In the test, two communication results are obtained from the referee box to the basestation and from the basestation to the client, so that the data from the referee box can be accepted by the client with a success rate of 93%.

Keywords: Indonesian Robot Contest; Soccer Robot; Base Station; Referee Box.

Abstrak. Kontes Robot Indonesia divisi Sepakbola Beroda merupakan lomba tahunan yang diadakan oleh kementerian pendidikan. Satu tim dalam divisi ini terdiri dari 3 robot yang terhubung dalam komunikasi nirkabel yang dikoordinir oleh komputer *base station*. Selain itu, *base station* juga menghubungkan ketiga robot tersebut dengan komputer wasit atau disebut dengan *RefBox* (*Referee Box*). Komputer *RefBox* sebagai server bagi *base station* dan *base station* sebagai server bagi robot. Masalah muncul pada *base station* yang harus berperan ganda, sebagai *server* dan *client*. Dalam pengujian didapat dua hasil komunikasi dari referee box ke basestation dan dari basestation ke client, sehingga data dari referee box dapat diteima oleh client dengan tingkat keberhasilan 93%.

Kata Kunci: Kontes Robot Indonesia; Robot Sepak Bola; Stasiun Pangkalan; Kotak Wasit.

PENDAHULUAN

Kontes Robot Sepakbola Indonesia Beroda (KRSBI-Beroda) diselenggarakan berdasarkan aturan yang berlaku di RoboCup Middle Size League (MSL), dengan menyesuaikan kondisi di Indonesia, misalnya pada ukuran lapangan dan lainnya. KRSBI Beroda ini merupakan salah satu kegiatan yang merupakan bagian dari Kontes Robot Indonesia (KRI) sebagai ajang kompetisi rancang bangun dan rekayasa dalam bidang robotika. KRSBI-Beroda diadakan untuk meningkatkan keilmuan dan kreatifitas mahasiswa di bidang robotika. Di dalam kontes ini, mahasiswa dituntut untuk bisa mengembangkan kemampuan dalam bidang mekanika, manufaktur, elektronik, pemrograman, dan strategi. Pemrograman meliputi *artificial intelligent, image processing*, komunikasi digital. Selain itu juga meningkatkan kemampuan mahasiswa dalam meneliti dan menulis artikel, sekaligus diperlukan pengembangan ke arah disiplin, toleransi, sportifitas, kerjasama, saling menghargai, kontrol emosi dan kemampuan *softskill* lainnya [1].

RefBox (Referee Box) adalah media yang digunakan dalam mengatur jalannya pertandingan dalam KRSBI Beroda. *RefBox* yang digunakan ini menganut peraturan RoboCup MSL. Fungsi *RefBox* yakni untuk membantu wasit, terutama dalam menyampaikan keputusan wasit kepada robot yang sedang bermain, sebagai penampil waktu pertandingan, dan juga untuk mencatat skor pertandingan. Secara khusus, setiap kali juri memerintahkan wasit untuk memberi sebuah isyarat, protokol komunikasi *RefBox* mengirimkan satu atau beberapa pesan ke komputer *base station* kedua tim. *RefBox* merupakan aplikasi *game controller* yang pengatur jalannya pertandingan yang memberikan perintah kepada robot dari awal dan akhir pertandingan, pelanggaran, tendangan bebas dan lain-lain seperti layaknya wasit pada permainan sepak bola yang sesungguhnya. *Base station* merupakan sebuah perangkat komputer yang berisi suatu program yang didesain sebagai koordinator antar robot, dan informasi bagi robot atas segala perintah dari wasit [2].

LANDASAN TEORI

Raspberry pi

Sebuah mini komputer dengan nama dagang Raspberry Pi 3+. Raspberry Pi 3+ merupakan *System on Modules* (SoMs) yang menggunakan chipset BCM2837B0 dengan CPU berupa 64-bit quadcore ARM Cortex-A53, eMMC Flash (pada varian non-Lite), GPU Broadcom VideoCore IV, dan memori 1GB LPDDR2 (berbagi antara CPU dan GPU). Raspberry Pi 3+ merupakan generasi ketiga dengan tambahan memori dari generasi sebelumnya. Antarmuka pada Raspberry Pi 3+ berupa 4 slot USB versi 2.0, Ethernet, audio I/O, HDMI, RasiCam (kamera khusus untuk Raspberry Pi), dan DSI (*Display Serial Interface*). Selain itu, Raspberry Pi 3+ telah dilengkapi dengan

modul WiFi b/g/n/ac dual band 2.4/5 GHz dan *Bluetooth* versi 5.0 yang telah terintegrasi [3].

Beberapa sistem operasi dirancang khusus untuk mendukung arsitektur Raspberry Pi, salah satunya adalah Raspbian. Raspbian merupakan varian dari sistem operasi linux dengan repo Debian. Raspbian mendukung beberapa bahasa pemrograman khas linux, salah satunya adalah python, C++, dan lain-lain. Raspberry Pi dapat beroperasi sebagai mana komputer pada umumnya, tetapi performanya masih lebih rendah karena dukungan perangkat keras yang minimalis. Dalam penelitian ini, raspberry cukup mampu untuk berperan sebagai *base station*. Bentuk dari Raspberry dapat dilihat pada Figure 1.

[Figure 1 about here.]

TCP/IP

TCP/IP (*Tranmission Control Protocol/Internet Protocol*) memungkinkan hubungan virtual antar komputer, dimana dua komputer atau lebih dapat saling berhubungan untuk pertukaran data serta layanan aplikasi jaringan lainnya. Tujuan daripada desain TCP/IP adalah sebagai berikut :

1. Standart protokol yang open. Artinya spesifikasi dapat diperoleh dengan bebas dan dikembangkan sesuai dengan hardware yang dimiliki. Dengan demikian TCP/IP dapat diimplementasikan pada platform hardware yang beragam.
2. Tidak tergantung pada jaringan fisik hardware. TCP/IP dapat diintegrasikan pada jaringan fisik yang bermacam-macam melalui ethernet, token ring, dial up (telepon) RS232 dan media transmisi lainnya.
3. Skema alamat yang luas. Skema alamat internet memungkinkan komputer mempunyai identitas tunggal (IP Address), sehingga walaupun mempunyai jangkauan internasional (wordwide), komputer manapun dapat dicapai dengan mudah karena mempunyai identitas yang jelas [4].

Referee Box

Refree box (disingkat *Refbox*) adalah sebuah perangkat komputer yang berisi aplikasi *game controller* yang digunakan dalam kontes robot Indonesia yang berfungsi sebagai pengatur jalannya pertandingan robot sepak bola beroda. *Refbox* memberikan perintah berupa dimulainya pertandingan, akhir pertandingan, pelanggaran dan tendangan bebas, dan lain-lain. Antarmuka pengguna tekstual dan grafis memungkinkan untuk instruksi dan pengawasan oleh manusia. Hal ini diperlukan secara khusus dalam situasi yang tidak terduga (misal Manusia harus menghentikan permainan karena sesuatu hal). *Refbox* juga mencatat setiap pesan yang dikirim atau yang diterima.

Refbox akan mengirimkan instruksi ke *base station* berupa bit data yang kemudian akan di terjemahkan dan dilanjutkan ke robot. *Refbox* menggunakan TCP sebagai protocol komunikasi dalam pengiriman perintahnya ke *base station* [5]. Bentuk dari Referee Box dapat dilihat pada Figure 2

[Figure 2 about here.]

Qt

Qt adalah framework pengembangan aplikasi *cross platform* (dapat beroperasi di berbagai system operasi) yang komperhensif dengan bahasa C++ dan menawarkan solusi pemrograman dengan konsep pemrograman berorientasi objek. Qt memiliki *tools* yang digunakan untuk menyederhanakan proses pengembangan program dan antar muka grafis pada *desktop*, *embedded system*, dan perangkat *mobile*; antara lain: Qt-Framework terdiri dari seperangkat API intuitif dalam bahasa C++, Qt *Quick* dengan sintaks yang mirip dengan JavaScript/CSS sebagai panduan bagi pengguna, dan Qt *Creator* IDE merupakan lingkungan pengembangan terpadu (*Integrated Development Environment*) yang *cross-platform*.

Selain mendukung C++, Qt juga memiliki *language binding* antara lain, PyQt untuk Python, Qyoto untuk C#, Jambix untuk Java, dan *language binding* lainnya. Solusi ini menungkinkan pengembangan program dalam berbagai bahasa pemrograman berbeda sehingga memberikan fleksibilitas bagi pemrogram. Untuk desain *form*, Qt dapat menggunakan antar muka pengguna yang menggunakan layout XML atau QML (Qt *Markup Language*) yang mirip dengan JavaScript dengan class-class *widget* yang lebih konsisten namun kurang praktis. Hal ini memungkinkan desain GUI (*Graphical User Interface*) yang lebih produktif, interaktif, dan menarik [6].

Python

Python memiliki sistem tipe dinamis dan manajemen memori yang otomatis. Ini mendukung beberapa paradigma pemrograman, termasuk berorientasi objek, imperatif, fungsional, prosedural, dan memiliki *library* standar yang besar dan komprehensif. *Compiler* python tersedia untuk banyak sistem operasi. Python adalah perangkat lunak open source dan memiliki model pengembangan berbasis komunitas, Python juga bersifat *cross platform* (dapat beroperasi di berbagai system operasi) [7]

Wireshark

Wireshark merupakan salah satu aplikasi untuk *network analysis tool*, atau *protocol analysis tool* atau *packet snifer*. Wireshark dapat digunakan sebagai *troubleshooting* jaringan. Wireshark memungkinkan pengguna mengamati data dari jar-

ingn yang sedang beroperasi dan langsung melihat dan mensortir data yang tertangkap. Informasi singkat dan detail bagi masing-masing paket data, termasuk *full header* dan posisi data. Wireshark mempunyai beberapa fitur termasuk display filter bahasa yang kaya dan kemampuan untuk merekonstruksi kembali sebuah aliran pada sesi TCP [8]. Bentuk dari Wireshark dapat dilihat pada Figure 3.

[Figure 3 about here.]

PERANCANGAN

Metode penelitian yang digunakan dalam penelitian ini adalah penelitian percobaan (*experimental research*), penelitian eksperimen merupakan bentuk khusus investigasi yang digunakan untuk menentukan variabel-variabel apa saja dan bagaimana bentuk hubungan antara satu dengan yang lainnya. Menurut konsep klasik, eksperimen merupakan penelitian untuk menentukan pengaruh variabel perlakuan (*independent variable*) terhadap variabel dampak (*dependent variable*). Definisi lain menyatakan bahwa penelitian eksperimen adalah penelitian yang dilakukan terhadap variabel yang data-datanya belum ada sehingga perlu dilakukan proses manipulasi melalui pemberian treatment/perlakuan tertentu terhadap subjek penelitian yang kemudian diamati/diukur dampaknya (data yang akan datang). Penelitian eksperimen juga merupakan penelitian yang dilakukan secara sengaja oleh peneliti dengan cara memberikan treatment/perlakuan tertentu terhadap subjek penelitian guna membangkitkan sesuatu kejadian/keadaan yang akan diteliti bagaimana akibatnya. Dimana penelitian ini bertujuan untuk menguji apakah perintah yang di kirim dari *RefBox* dapat di terima oleh *base station* dan di teruskan kepada setiap robot [9].

Jaringan Komunikasi

Jaringan komunikasi di susun berdasarkan peraturan mengenai komunikasi robot pada pedoman robot sepak bola beroda. Sistem komunikasi ini menggunakan protocol TCP sebagai penghubung *base station* ke *RefBox*. Dalam proses ini komunikasi menggunakan kabel LAN sebagai jalur penghubung. Jaringan komunikasi sepak bola beroda dapat dilihat pada Figure 4 berikut:

[Figure 4 about here.]

Dalam pertandingan, robot tidak boleh langsung berkomunikasi langsung dengan robot lain, semua komunikasi data yang terjadi harus melalui *base station* sebagai koordinator. Begitu pula dengan perintah dari *RefBox* harus di kirim melalui *base station* yang kemudian akan teruskan ke robot. *RefBox* memiliki port pada jaringan default 28097 sebagai penghubung dengan

base station. Pada sistem komunikasi *base station* menggunakan TCP/IP sebagai client agar dapat menerima data dari *RefBox* yang berupa bytes data yang sudah di terjemahkan menjadi *string* sebagai perintah untuk mengatur jalannya pertandingan seperti: START, STOP, KICK OFF, CORNER, GOAL dan perintah lainnya layaknya pertandingan sepak bola. Setiap *base station* mengetahui perintah yang diberikan oleh *RefBox* ke *base station* yang lain [10].

Figure 5 menunjukkan *flowchart* dari *base station*, yang akan mendistribusikan data dari *RefBox* yang kemudian akan di eksekusi oleh robot. Jika *base station* sudah terhubung dalam satu jaringan dengan *RefBox* dan IP *base station* sudah teregistrasi di *RefBox*, jika terjadi kesalahan maka secara otomatis akan dilakukan pengkoneksian ulang. Jika *base station* telah terkoneksi dengan *RefBox* akan ada pilihan untuk menjadi tim cyan atau magenta. setelah sukses terkoneksi, *base station* akan mendapat data karakter berupa huruf 'W' sebagai tanda bahwa *base station* telah berhasil terkoneksi, setelah itu *base station* akan menunggu perintah dari *RefBox* untuk melakukan *kick off* yang berupa data karakter 'k' untuk magenta dan data karakter 'K' untuk cyan. Perbedaan perintah dari *RefBox* antara cyan dan magenta adalah pada karakter data yang di terima, apabila berada dalam tim cyan maka perintah berupa data char huruf capital, sedangkan untuk tim magenta berupa data char huruf kecil.

[Figure 5 about here.]

Dalam sistem komunikasi, menggunakan LAN (local area network) dengan TCP yang memiliki keunggulan dalam konektivitas yang terjamin. Hubungan komunikasi yang diterapkan antara *RefBox* ke *base station*. Jaringan komputer dalam pertandingan KRSBI-Beroda ini menggunakan LAN melalui sebuah *switch*. Selama pertandingan berlangsung *RefBox* akan berperan sebagai *server* yang akan menunggu *base station* sebagai *client* untuk terhubung dengan *RefBox*. Proses penyambungan *base station* ke *RefBox* menggunakan IP dan *Port* yang telah ditentukan dalam pertandingan.

Algoritma *base station* sebagai client *RefBox*:

- Inisialisasi IP dan Port *RefBox*
- Inisialisasi socket
- Create socket TCP
- Base station
- Base station

Sebelum *base station* login ke *RefBox* perlu adanya konfigurasi dengan menyimpan IP *base station* yang akan digunakan selanjutnya oleh *RefBox*. Figure 6, baris 61 menunjukkan kedua

tim "DAHANA" dan "dahana" telah terhubung dengan *RefBox*. Dalam konfigurasi ini memerlukan *UniCastAddress*, *MultiCastAddress*, *NamaTeam*, *LongName*, *ShortName*, yang harus diisi. Ini bertujuan agar *RefBox* dapat membedakan antara tim cyan dan magenta.

Setiap pergerakan yang dilakukan oleh setiap robot berasal dari perintah yang dikirimkan *RefBox* melalui *base station*. Dalam hal ini, robot berperan sebagai *client* yang hanya mengirim dan menerima data dari *base station*, sedangkan *base station* sebagai *server*. *Base station* mempunyai peran ganda atau harus mampu melakukan *multiple processing* yang juga bertugas sebagai *client* bagi *RefBox* [11].

[Figure 6 about here.]

Ketika *base station* menerima instruksi *start* dari *RefBox*, *base station* akan mengirimkan data bytes berupa string agar robot "A" mendeteksi posisi robot lawan kemudian mengirimkan data posisi robot lawan ke *base station*. Sehingga *base station* dapat menentukan pergerakannya selanjutnya sesuai dengan yang difungsikan. Jika *RefBox* mengirimkan instruksi *stop* ke *base station*, maka *base station* akan menghentikan seluruh pergerakan dari seluruh robot.

Algoritma robot sebagai client bagi *base station*.

- Inisialisasi socket
- Create socket TCP
- Client terhubung ke *base station*
- Client menerima atau mengirim data dari atau ke *base station*

[Figure 7 about here.]

Dalam komunikasi antar robot harus melalui *base station* sebagai koordinasi gerakan antar robot untuk mengantisipasi sebuah kondisi. Misalkan ketika robot membawa bola atau tidak yang dapat di gambarkan sebagai *flowchart* pada Figure 7.

Pengujian Program Komunikasi

Pengujian program komunikasi pada Raspberry Pi sebagai perangkat koordinasi pada setiap robot. *RefBox* akan mengirimkan data bytes berupa karakter kepada *base station* yang kemudian akan di teruskan ke robot. Robot juga akan mengirimkan kode kepada *base station* yang kemudian akan dikembalikan lagi ke robot sebagai perintah. Analisa hasil pengamatan dan data yang diperoleh akan di amati, apakah tersampaikan dengan benar dan akurat atau tidak. Hasil pengiriman data dapat diamati menggunakan aplikasi Whirehark.

HASIL DAN PEMBAHASAN

Pengujian dilakukan khusus pada base station. Sebab terjadi 2 komunikasi pada base station, yaitu antara robot dan base station dan base station dan RefBox.

Pengujian Komunikasi antara Robot dan Base Station

Pada penelitian ini menggunakan bahasa pemrograman python 3 dan qt4 sebagai GUI pendukung aplikasi dan menggunakan system operasi Xubuntu 14.0. Sedangkan pemrograman socket yang digunakan adalah TCP sebagai penghubung base station ke RefBox.

[Figure 8 about here.]

Sock stream merupakan *sintaks unix* dalam pemrograman menggunakan protocol TCP, sedangkan *Sintaks connect()* dalam program di atas berfungsi agar base station dapat terkoneksi ke RefBox. Pada Figure 8 menunjukkan tampilan pada layar base station. Sedangkan pada Figure 9 merupakan code pada base station. Pada Figure 10 adalah proses komunikasi yang terjadi antarabase station dengan RefBox yang diamati oleh Wireshark.

[Figure 9 about here.]

[Figure 10 about here.]

Pada Figure 10 adalah salah satu pengambilan gambar tampilan pada Wireshark. Maka dapat dianalisa waktu terjadi pengiriman dan penerimaan data, IP asal, IP tujuan, Protocol yang digunakan, panjang data dalam byte yang dikirimkan, serta Port yang digunakan. Pada protocol TCP terdapat beberapa jenis *flag* untuk menunjukkan status dari lalu lintas data pada bagian info. Seperti flag SYN berfungsi untuk menunjukkan bahwa segmen TCP yang bersangkutan mengandung *Initial Sequence Number (ISN)*. *Flag ACK* berfungsi untuk mengetahui apakah data yang dikirimkan sudah diterima atau belum di komputer *client*. *Flag PSH* berfungsi untuk mengindikasikan isi dari TCP yang diterima di komputer *client*. *Flag FIN* berfungsi untuk menandakan bahwa pengirim segmen TCP telah selesai dalam mengirimkan data dalam sebuah koneksi TCP.

Pengujian Komunikasi antara Base Station dan RefBox .

Dalam pengujian ini base station akan dikoneksikan ke RefBox menggunakan LAN dengan protocol TCP/IP. Dalam hal ini penyambungan antara base station dan RefBox menjadikan base station sebagai *client*, sebab RefBox memiliki status sebagai *server*. Dalam proses pengujian ini RefBox menggunakan alamat IP dari komputer RefBox dan port default dari RefBox yaitu 28097. RefBox akan menampilkan form dialog untuk login tim seperti yang ditunjukkan pada Figure 11. Jika base

station telah terhubung dengan benar, maka akan ditampilkan seperti pada Figure 12 (dengan nama "DAHANA" pada tim cyan).

[Figure 11 about here.]

[Figure 12 about here.]

Setelah login dengan base station pada RefBox berhasil, pengujian perintah dari komputer RefBox dengan mengirimkan bytes data ke base station yang berupa karakter yang menjadi perintah bagi robot, hasil dari pengiriman RefBox ke base station dapat dilihat pada tabel 1 (catatan: huruf capital ditujukan untuk tim cyan, sedangkan huruf kecil ditujukan untuk tim magenta, kecuali perintah START dengan karakter 's' dan STOP dengan karakter 'S' berlaku untuk kedua tim). Dalam pengujian komunikasi ini akan dilakukan sebanyak 15 kali pengiriman perintah secara berturut-turut, sehingga di dapat hasil pada Table I bawah ini.

[Table 1 about here.]

Dari pengujian di atas data dapat dilihat menggunakan aplikasi wireshark, terdapat lalu lintas data dari ip yang di gunakan sebagai referee box.

[Figure 13 about here.]

Pada Figure 13 pengujian komunikasi ini dilakukan tanpa ada jeda sebanyak 15 data, Saat dilakukan pengiriman data tanpa pergerakan maka sama dengan melakukan pengiriman tanpa delay. Jika pengiriman dilakukan tanpa delay atau terus menerus maka akan terjadi penumpukan data. Data yang dikirim memerlukan waktu untuk diterima dan dilanjutkan oleh robot. Maka saat data dikirim terus menerus tanpa delay atau jeda maka data akan bertumpuk dan menyebabkan pengiriman atau komunikasi berhenti sehingga terjadi kegagalan pengiriman 1 data pada pengiriman ke 12.

Dalam 15 kali pengiriman data yang dilakukan, terdapat satu kali gagal dalam pengujian.

Dari hasil ini maka dapat diketahui tingkat keberhasilan sistem yaitu $\frac{15-1}{15} \times 100\% = 93, 33\%$

Maka persentase keberhasilan sistem komunikasi ini mencapai 93%.

Pengujian Antara RefBox dan Robot

Dalam hal ini, base station berperan ganda yaitu sebagai *client* dari RefBox dan juga sebagai *server* untuk robot, hal ini base station harus selalu *standby* menghubungkan informasi RefBox dan robot. Jika apabila ada salah satu robot mengalami masalah maka robot harus melakukan login ulang ke base station.

Agar fungsi dari *base station*, berjalan dengan baik sebagai *client* dan *server* fungsi *thread* perlu di masukan ke dalam program agar *base station* dapat menjalankan fungsi ganda tersebut. Fungsi *thread* ini akan menjalankan dua fungsi inti dari *base station* secara bersamaan. Hasil data *client* dapat dilihat pada.

Dalam pengujian ini robot akan disambungkan ke *base station* melalui jalur nirkabel WiFi, dengan alamat IP yang digunakan oleh *base station* yaitu 192.168.0.10 dan *Port* 5560. Hasil yang di harapkan dari pengujian ini adalah data bytes yang dikirim dari *RefBox* dapat diterima dan diteruskan ke robot melalui *base station*. Dari Figure 14 menunjukkan data yang dikirim oleh *RefBox* dan telah diterima oleh *base station*. Sedangkan pada Figure 15 menunjukkan data yang dikirimkan oleh *base station* ke robot.

Robot yang telah terkoneksi oleh *base station* dengan benar dapat mengirimkan kondisi yang ada, dalam pengujian ini robot sebagai *client* mengirimkan dua kondisi yaitu “ada” dan “tidak”, yang dimaksudkan “ada” artinya kondisi dimana robot mendapatkan bola sedangkan “tidak” merupakan sebaliknya dimana robot tidak mendapatkan bola. Dari gambar 15 robot mendapatkan perintah balik dari *base station* yang berupa perintah “maju” dan “mencari bola”. Dalam hal ini, perintah “maju” dimaksudkan robot yang telah membawa bola harus menggiring bola ke arah lawan, sedangkan “mencari bola” di maksudkan robot harus mencari bola.

[Figure 14 about here.]

[Figure 15 about here.]

KESIMPULAN

Berdasarkan hasil penelitian dengan menggunakan bahasa pemrograman python berbasis system operasi raspbian yang harus berperan ganda pada menggunakan protokol TCP, dapat disimpulkan bahwa :

1. Data dari *RefBox* yang berupa data bytes dalam bentuk karakter dapat dikirimkan menggunakan *socket* pada python dalam bentuk *utf 8*. Komunikasi yang terjadi antara *RefBox* dan *base station* dapat berjalan dengan baik menggunakan jalur kabel LAN dengan protocol TCP.

2. Data informasi yang kirim oleh robot yang menggunakan melalui jalur WiFi dapat di terima dengan baik oleh *base station* dan dapat dikirim kembali ke robot berupa data perintah.

3. Multi tasking yang dilakukan oleh *base station* sebagai *server* bagi robot dan *client* bagi *RefBox* dapat berjalan dengan baik sebagai sebagai koordinator dan pengambil keputusan. Hal ini dapat berjalan dengan baik jika jalur komunikasi antara robot dan *base station*, dan antara *base station* dan *RefBox* berbeda.

REFERENSI

- [1] Ristekdikti, Buku, Kontes, Sepakbola, B. (krsbi, & Beroda, “BUKU PANDUAN KONTES ROBOT SEPAKBOLA INDONESIA BERODA (KRSBI Beroda),” Direktorat Kemahasiswaan Direktorat Jenderal Pembelajaran dan Kemahasiswaan Kementerian. Riset, Teknol. dan Pen- didik. Tinggi Republik Indonesia,” D. K. D. J. P. dan Kemahasiswaan Kementerian. Riset & T. dan Pendidik. Tinggi Republik Indones., Eds., 2017, pp. 15–46.
- [2] T. Ardhiansyah, I. Syarifuddin, M. R. Naufal, Y. Pramono, & O. T. Hartatik, “Pergerakan Otomatis Robot Sepak Bola Beroda Melalui Komunikasi dengan Referee Box Menggunakan Base Station,” 5th Indones,” *Syposium Robot. Syst. Control*, pp. 82–86, 2017.
- [3] R. Pi, ““DATASHEET Raspberry Pi Compute Module 3+ Raspberry Pi Compute Module 3+ Lite,”,” 2019.
- [4] H. Listiyono, “Program Socket Untuk Mengirim File Dengan Visual Basic Pada Sistem Operasi Windows,” *J. Din. Inform*, vol. 1, no. 1, pp. 1–5, 2009.
- [5] T. Niemueller, “Referee Box for the RoboCup Logistics League Integration Manual,” pp. 1–22, 2014.
- [6] N. M. Figueiredo, A. J. R. Neves, N. Lau, A. Pereira, & G. Corrente, “Control and monitoring of a robotic soccer team: The base station application,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5816, pp. 299–309, 2009.
- [7] K. R. R, A. Rahmansyah, W. Darwin, & A. R. Box, “Penggunaan Bahasa Pemrograman Python Sebagai Pusat Kendali Pada Robot 10-D,” pp. 23–26, 2017.
- [8] A. M. A. El-Sayed & M. M. A. Al-Fadel, “Weak Solutions of a Coupled System of Urysohn Stieltjes Functional (Delayed) Integral Equations,” pp. 1–6, 2018. [Online]. Available: [10.1155/2018/8714694](https://doi.org/10.1155/2018/8714694);<https://dx.doi.org/10.1155/2018/8714694>
- [9] A. Jaedun, “ Metodol. Penelit. Eksperimen,” pp. 0–12, 2011.
- [10] Y. D. Santoso, S. Nugroho, H. K. Wardana, & M. Fielder, “Communication Between Wheeled Soccer Robot,” pp. 287–299, 2017.
- [11] W. Dai, Q. Yu, J. Xiao, & Z. Zheng, “Communication - less Cooperation between Soccer Robots,” 2016.

Conflict of Interest Statement: The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Author [s]. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Received: 2020-03-23

Accepted: 2020-04-02

Published: 2020-04-25

DAFTAR TABEL

I Hasil Pengujian Data dari Referee 9

TABLE I. HASIL PENGUJIAN DATA DARI REFEREE

| Referee box | Base station | Keterangan |
|-------------|--------------|------------|
| KICK OFF | "K" | Berhasil |
| START | "s" | Berhasil |
| STOP | "S" | Berhasil |
| FREE KICK | "F" | Berhasil |
| START | "s" | Berhasil |
| STOP | "S" | Berhasil |
| GOAL KICK | "G" | Berhasil |
| THROW IN | "T" | Berhasil |
| CORNER | "C" | Berhasil |
| START | "s" | Berhasil |
| STOP | "S" | Berhasil |
| FREE KICK | "F" | Gagal |
| START | "s" | Berhasil |
| STOP | "S" | Berhasil |
| GOAL KICK | "G" | Berhasil |

DAFTAR GAMBAR

| | | |
|----|---|----|
| 1 | Raspberry Pi 3+ | 11 |
| 2 | Referee Box | 11 |
| 3 | Aplikasi Wireshark | 11 |
| 4 | Jaringan komunikasi sepak bola beroda | 12 |
| 5 | Flowchart base station | 12 |
| 6 | Konfigurasi Referee Box | 12 |
| 7 | Flowchart Pengiriman Data | 13 |
| 8 | Interface Base station | 13 |
| 9 | Code pada <i>base station</i> (atas) sebagai <i>server</i> bagi robot, (bawah) sebagai <i>client</i> bagi <i>RefBox</i> | 14 |
| 10 | Capturing Wireshark | 14 |
| 11 | Login Team | 15 |
| 12 | Login team berhasil | 15 |
| 13 | Transfer Data | 16 |
| 14 | Hasil Data Client | 16 |
| 15 | Hasil Pengiriman Dari Client | 17 |



Figure 1. Raspberry Pi 3+

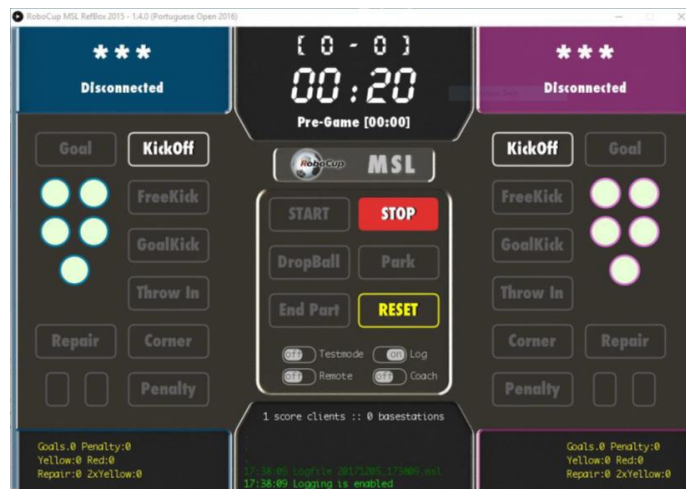


Figure 2. Referee Box

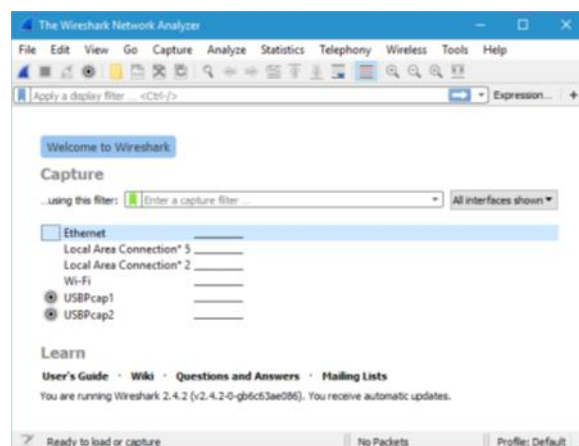


Figure 3. Aplikasi Wireshark

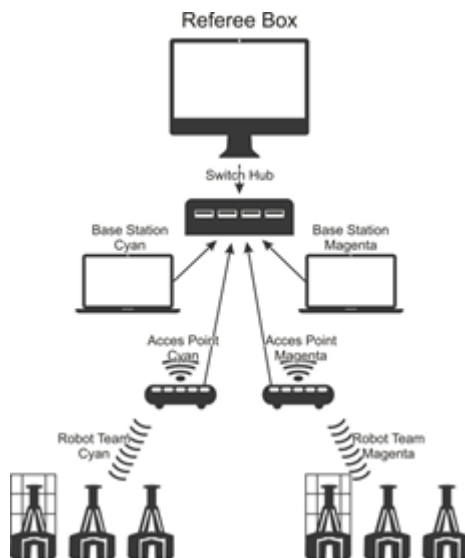


Figure 4. Jaringan komunikasi sepak bola beroda

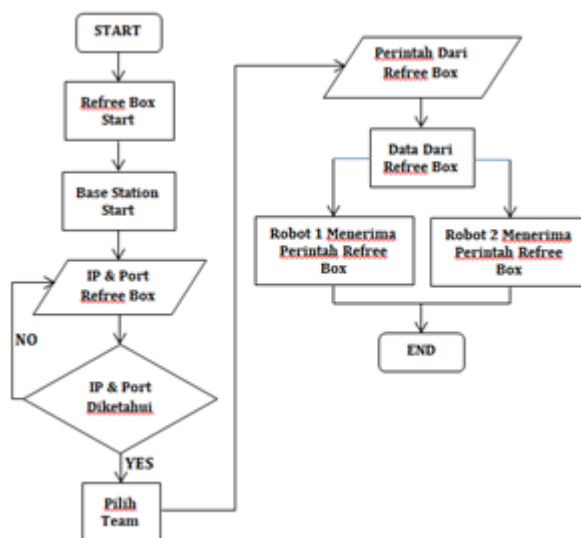


Figure 5. Flowchart base station

```

53 172.16.83.*,224.16.32.83,n.a.,n.a.,n.a.
54 172.16.84.*,224.16.32.84,n.a.,n.a.,n.a.
55 172.16.85.*,224.16.32.85,n.a.,n.a.,n.a.
56 172.16.86.*,224.16.32.86,n.a.,n.a.,n.a.
57 172.16.87.*,224.16.32.87,n.a.,n.a.,n.a.
58 172.16.88.*,224.16.32.88,n.a.,n.a.,n.a.
59 172.16.100.*,224.16.32.100,Mixed teams,Mixed teams,Mixed te
60 127.0.0.*,224.16.32.127,localhost,LOCALHOST,local
61 192.168.1.*,224.16.32.1,dahana,DAHANA,dahana
62
    
```

Figure 6. Konfigurasi Referee Box

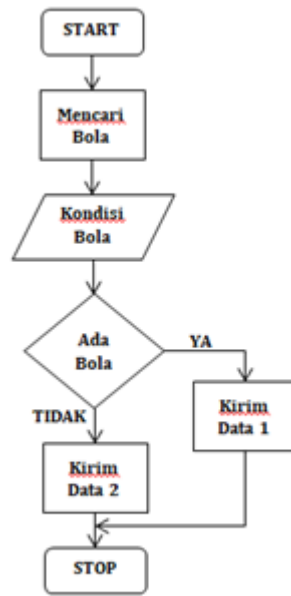


Figure 7. Flowchart Pengiriman Data



Figure 8. Interface Base station

```

1 import sys
2 import socket
3 import Server
4 import Client
5 import threading
6
7 class Base02(object):
8     def __init__(self, parent = None):
9         self.parent = parent
10        self.sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
11        self.sock.bind(('0.0.0.0', 8080))
12        self.sock.listen(5)
13        self.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
14        self.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEPORT, 1)
15        self.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEPORT, 1)
16        self.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEPORT, 1)
17        self.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEPORT, 1)
18        self.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEPORT, 1)
19
20    def connectServer():
21        host = '192.168.0.10'
22        port = 8080
23        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
24        s.connect((host, port))
25        s.send('GET / HTTP/1.1\r\n\r\n')
26        data = s.recv(1024)
27        print(data)
28
29    def startBase02():
30        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
31        s.bind(('0.0.0.0', 8080))
32        s.listen(5)
33        while True:
34            client, addr = s.accept()
35            t = threading.Thread(target=connectServer)
36            t.start()
37
38    def main():
39        startBase02()
40
41    if __name__ == '__main__':
42        main()
43
44    def connectServer():
45        host = '192.168.0.10'
46        port = 8080
47        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
48        s.connect((host, port))
49        s.send('GET / HTTP/1.1\r\n\r\n')
50        data = s.recv(1024)
51        print(data)
52
53    def startBase02():
54        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
55        s.bind(('0.0.0.0', 8080))
56        s.listen(5)
57        while True:
58            client, addr = s.accept()
59            t = threading.Thread(target=connectServer)
60            t.start()
61
62    def main():
63        startBase02()
64
65    if __name__ == '__main__':
66        main()
    
```

Figure 9. Code pada *base station* (atas) sebagai *server* bagi robot, (bawah) sebagai *client* bagi *RefBox*.

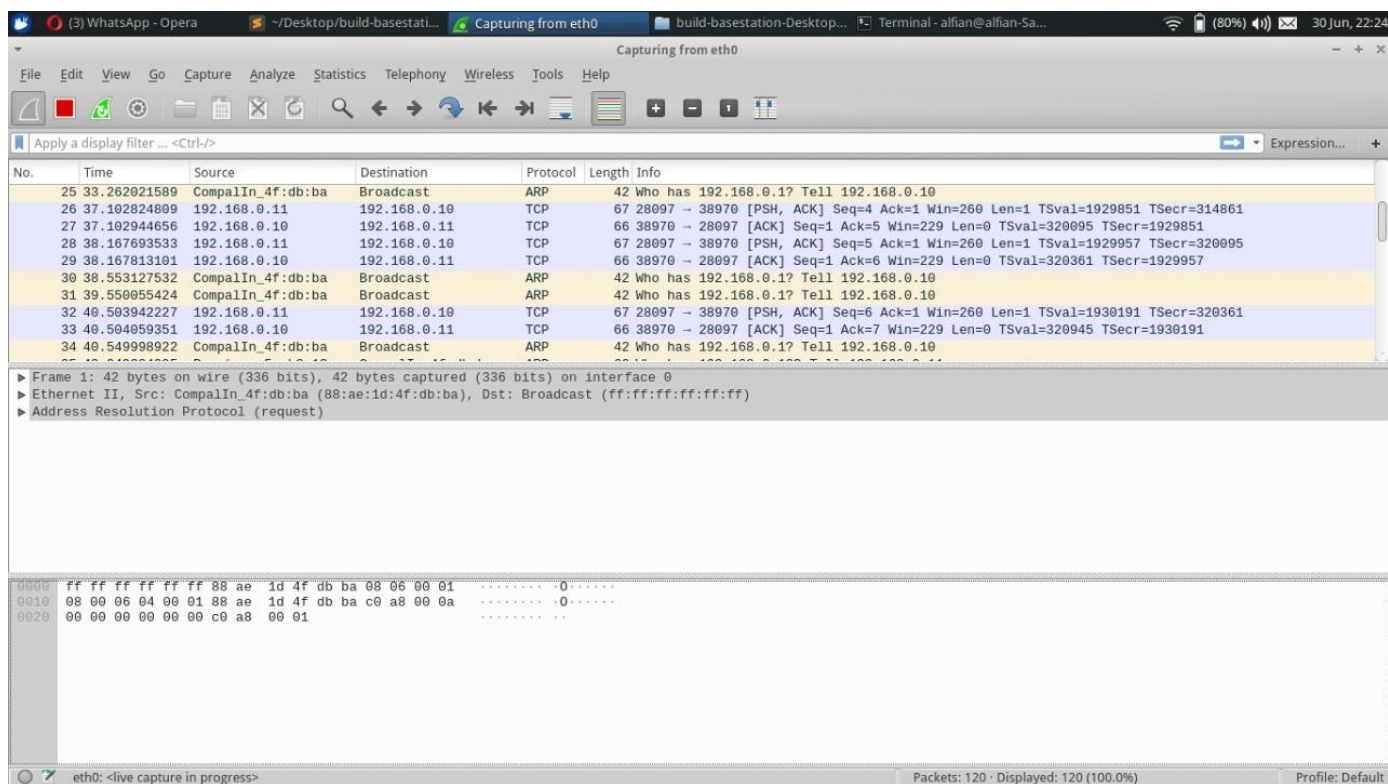


Figure 10. Capturing Wireshark



Figure 11. Login Team

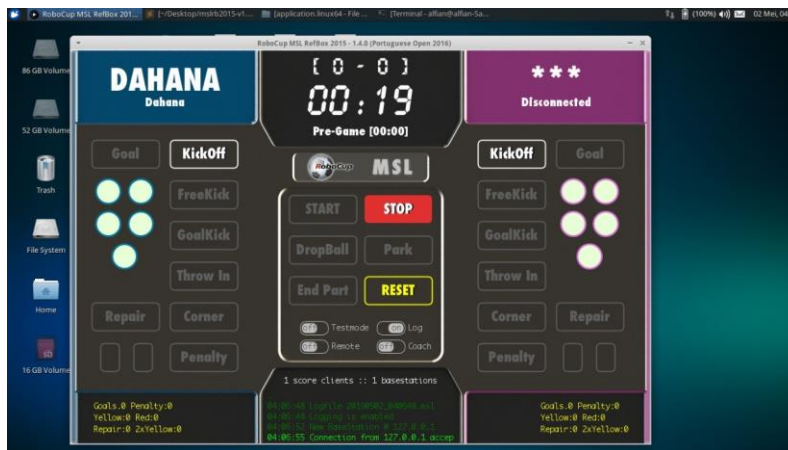


Figure 12. Login team berhasil

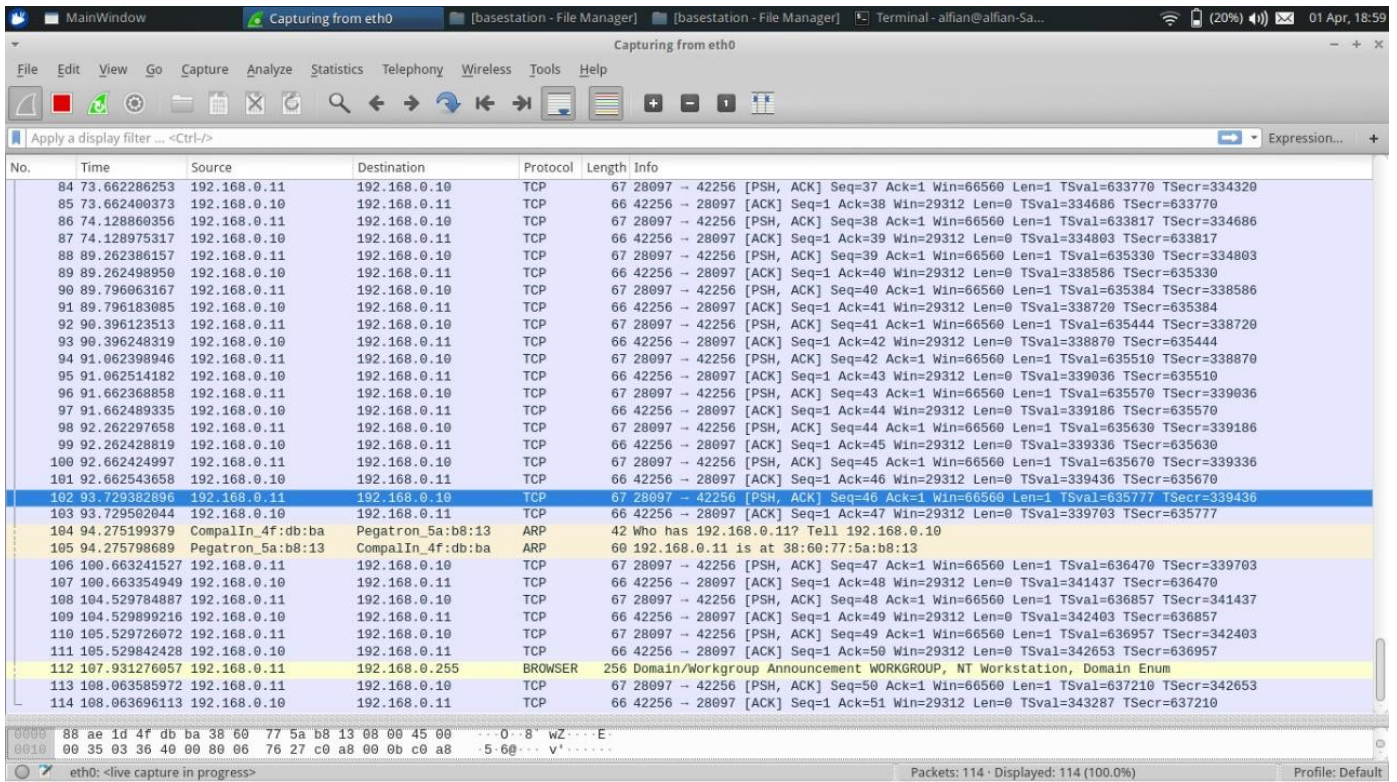


Figure 13. Tranfer Data

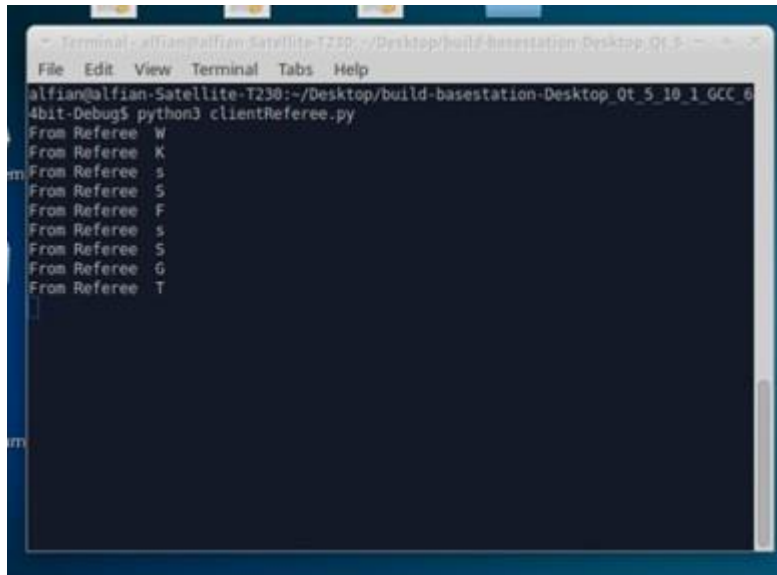
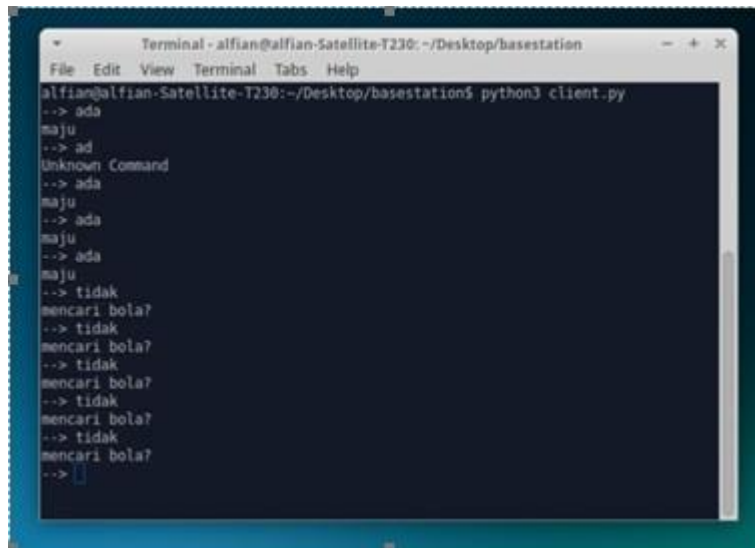


Figure 14. Hasil Data Client



```
Terminal - alfian@alfian-Satellite-T230: ~/Desktop/basestation
File Edit View Terminal Tabs Help
alfian@alfian-Satellite-T230:~/Desktop/basestation$ python3 client.py
--> ada
maju
--> ad
Unknown Command
--> ada
maju
--> ada
maju
--> ada
maju
--> tidak
mencari bola?
--> tidak
mencari bola?
--> tidak
mencari bola?
--> tidak
mencari bola?
--> tidak
mencari bola?
--> [ ]
```

Figure 15. Hasil Pengiriman Dari Client