

PEMBUATAN ANTARMUKA APLIKASI PENGATUR MASSA BERAS DAN VOLUME AIR UNTUK PENANAK NASI BERBASIS IOT

INTERFACE MAKING OF RICE MASS AND WATER VOLUME SETTER APPLICATION FOR IOT-BASED RICE COOKER

Josua Ronaldo S. ¹, Ir. Porman Pangaribuan, M.T. ², Dr.Eng Willy A. Cahyadi, S.T., M.T. ³ ^{1,2,3} Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom jorosh@student.telkomuniversity.ac.id, 2porman@telkomuniversity.ac.id, 3wacze@telkomuniversity.ac.id

Abstrak — Menanak nasi sudah menjadi aktivitas sehari-hari yang dilakukan sebagian besar masyarakat Indonesia. Nasi menjadi makanan pokok yang dikonsumsi sehari-hari hampir di sebagian wilayah Indonesia bahkan di wilayah Asia sekalipun. Hal itu disebabkan nasi menjadi sumber karbohidrat yang sangat baik bagi tubuh. Adanya *rice cooker* sudah menjadi alat yang selalu digunakan untuk menanak nasi di masa sekarang ini. Akan tetapi, terlepas dari kemudahan dalam menggunakan *rice cooker* muncul kekurangan dari alat tersebut. Pengguna harus menakar beras dan air secara langsung dan tepat takaran agar menghasilkan nasi yang baik. Keterbatasan tersebut membutuhkan kesediaan waktu dan tempat bagi pengguna *rice cooker*.

Berdasarkan permasalahan tersebut dibutuhkan antarmuka aplikasi yang dapat mengontrol dan memantau suatu sistem penanak nasi yang dapat menakar beras dan air secara otomatis. Pengontrolan sistem menggunakan aplikasi android berbasis IoT. Aplikasi *android* akan dapat memberi perintah untuk memulai proses menanak nasi, memeriksa ketersediaan air dan beras, melihat riwayat memasak, serta mematikan atau menghidupkan *rice cooker*. Pemanfaatan konsep IoT dengan menyimpan data di *web server* menggunakan layanan *web hosting* memungkinkan aplikasi *android* dan sistem dapat mengirim dan menerima data melalui koneksi internet sehingga aplikasi *android* dapat memantau dan mengontrol sistem dari jarak jauh dan kapan selama sistem dan aplikasi terkoneksi ke internet.

Adapun hasil dari tugas akhir ini adalah pembuatan antarmuka aplikasi penanak nasi yang dapat mengirimkan data banyak beras 2-9 cup dengan delay rata-rata keseluruhan adalah 235 ms dan *data rate* keseluruhan adalah 1,5 Kbps. Lalu rata-rata energi yang dibutuhkan saat sistem penanak nasi bekerja pada satu jam pertama adalah 0.08875 KWh dan pada dua jam pertama adalah 0.005375 KWh.

Kata Kunci : *Internet of Things, Android Studio, phpMyAdmin, Mikrokontroler*

Abstract — Cooking rice has become a daily activity carried out by most Indonesian people. Rice is the most common staple food in Indonesia even in Asia. It becomes rice for a very good source for the body. The existence of a rice cooker has become a tool that is always used to cook rice today. However, in spite of the ease of using the rice cooker a shortcoming of the tool appeared. The user must conquer rice and water directly and precisely to make good rice. This limitation requires setting the time and place for rice cooker users.

Based on this problem an interface application is needed that can control and use a rice cooker system that can measure rice and air automatically. The control system uses an IoT-based android application. The android application will be able to give commands to start the process of cooking rice, check air and rice, see cooking participation, and activate or start rice cookers. Using the IoT concept by storing data in a web server using a web hosting service that supports Android applications and systems can send and receive data through an internet connection so that the Android application can connect and control the system remotely and anytime using systems and applications connected to the internet.

The final result of this final project is the creation of a rice cooker application interface that can send data of 2-9 cups of rice with an overall average delay of 235 ms and an overall data rate of 1.5 Kbps. Then the average energy needed when the rice cooker system works in the first hour is 0.08875 KWh and in the first two hours is 0.005375 KWh.

Keywords: *Internet of Things, Android Studio, phpMyAdmin, Microcontroller*

1. Pendahuluan

Nasi merupakan salah satu bahan pangan utama yang dikonsumsi oleh sebagian besar masyarakat Indonesia sebagai sumber karbohidrat yang baik untuk kebutuhan energi tubuh. Menurut Departemen Kesehatan Republik Indonesia, nasi merupakan bahan makanan favorit yang paling banyak dikonsumsi masyarakat di negara Asean seperti Indonesia, Filipina, Thailand, dan Singapura [1]. Seiring dengan perkembangan teknologi, masyarakat saat ini menggunakan penanak nasi (*rice cooker*) sebagai alat untuk memasak nasi.

Pada kondisi saat ini terdapat keterbatasan jarak dan waktu di mana pengguna *rice cooker* harus melakukan proses dari menakar beras dan air hingga mengalihkan ke mode memasak (*cooking*) secara langsung. Pengguna *rice cooker* seperti ibu-ibu rumah tangga maupun pekerja pada restoran/rumah makan harus melakukan proses tersebut yang sering sekali menjadi kendala di mana takaran beras dengan takaran air yang kurang sesuai membuat nasi menjadi terlalu kering atau menjadi terlalu lembek [2]. Hal tersebut menjadi masalah yang dapat terjadi saat pengguna *rice cooker* ingin menanak nasi akan tetapi memiliki pekerjaan lain yang harus dikerjakan atau pengguna tidak sedang berada di dekat alat penanak nasi.

Pada perkembangan penggunaan *smartphone* dan internet yang sudah sangat luas, diperkirakan 93,2% pengguna *smartphone android* dan 338,2 juta koneksi seluler di Indonesia dari Januari 2019 – Januari 2020 [3]. Dari data seperti itu, hampir setiap orang menggunakan aplikasi android dan menggunakan koneksi internet. Dengan kebutuhan itu, dapat diimplementasikan konsep *Internet of Things* (IoT) pada sistem penanak nasi. Oleh karena itu, dalam menutupi keterbatasan jarak dan waktu serta dapat mengatur takaran banyak beras diperlukan antarmuka aplikasi yang dapat digunakan pada *smartphone* pengguna *rice cooker* untuk mengontrol dan memantau sistem alat penanak nasi. Aplikasi akan terintegrasi dengan sistem dengan pengimplementasian IoT.

Berdasarkan latar belakang masalah, didapatkan rumusan masalah sebagai berikut.

1. Bagaimana desain antarmuka aplikasi penanak nasi berbasis IoT?
2. Bagaimana pengiriman perintah mulai menanak nasi melalui aplikasi penanak nasi?
3. Bagaimana penerapan konsep IoT pada aplikasi dan sistem penanak nasi?

2. Landasan Teori

Penanak nasi merupakan alat elektronik yang sudah biasa digunakan untuk menanak nasi hampir pada semua kalangan penduduk Indonesia. Seiring implementasi konsep IoT sudah dikembangkan pada beberapa alat elektronik, memungkinkan untuk mengintegrasikan alat penanak nasi otomatis dengan aplikasi penanak nasi. Dengan implementasi IoT dapat mengirimkan perintah menanak nasi dari jarak jauh dan kapan saja melalui aplikasi pada *smartphone* dengan koneksi internet.

Pengiriman perintah menanak nasi berupa data banyak cup beras yang dipilih pengguna melalui aplikasi. Data yang dikirimkan disimpan ke *database web server*. Pengiriman data menggunakan protokol http sebagai komunikasi antara aplikasi dan alat penanak nasi otomatis terhadap *database*. Protokol http bersifat *request* (permintaan) dan *response* (respon). Pada alat penanak nasi otomatis terdapat sistem yang dikontrol oleh dua mikrokontroler. Mikrokontroler tersebut terhubung melalui komunikasi serial.

2.1 Latensi (Delay)

Latensi merupakan jeda waktu yang dibutuhkan dalam pengantaran paket data dari pengirim ke penerima. Semakin tinggi jeda waktu atau latensi tersebut maka akan semakin tinggi resiko kegagalan akses [4].

2.2 Packet Loss

Packet loss adalah suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang. Merupakan perbandingan seluruh paket data yang hilang dengan total paket data dikirimkan antara sumber dan tujuan pengirim [4].

2.3 MySQL

MySQL merupakan salah satu aplikasi yang digunakan untuk menyimpan data dalam sebuah aplikasi. Pada aplikasi dan sistem alat akan menyimpan data secara *online* pada MySQL sehingga dapat mengambil dan mengirim data dari aplikasi dan sistem alat.

2.4 Mikrokontroler

Mikrokontroler merupakan sebuah *chip* yang bertugas sebagai pengatur rangkaian elektronika, melakukan perintah-perintah yang diberikan, dan menjadi bagian utama dari suatu program terkomputerisasi. Penggunaan mikrokontroler pada sistem alat ialah memakai mikrokontroler Arduino Uno dan NodeMCU. Pemilihan kedua mikrokontroler tersebut berkaitan dengan konsep IoT di mana Arduino uno berfungsi sebagai mikrokontroler yang mengeksekusi program menanak nasi sedangkan nodemcu berfungsi sebagai mikrokontroler yang mengkomunikasikan Arduino uno dengan aplikasi secara online menggunakan koneksi internet agar dapat terhubung dengan penyimpanan web server. Program yang akan ditanamkan dalam mikrokontroler akan menjadi perintah untuk mengerjakan fungsi-fungsi tertentu dalam menjalankan tugas yang kompleks.

2.5 Android SDK (Software Development Kit)

Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk memulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman *Java*. Fitur pada android seperti *framework* aplikasi yang mendukung penggantian komponen dan *reusable*, dukungan untuk audio, video, gambar, dan wifi, serta lingkungan *development* yang lengkap dan kaya termasuk perangkat emulator, *tools* untuk *debugging*, dan kerja memori menjadi faktor penting untuk mendukung pengembangan aplikasi android [5].

2.6 Sistem Operasi Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan tablet. *Android* menyediakan platform terbuka untuk mengembangkan aplikasi buatan *developer*. Dengan sistem distribusi *open source* yang digunakan memungkinkan para developer untuk menciptakan beragam aplikasi menarik yang dapat dinikmati oleh para penggunanya, seperti *game*, *chatting* dan lain-lain. hal ini menjadikan *smartphone* berbasis *Android* ini lebih murah dibanding gadget dengan sistem operasi lainnya [6].

2.7 Wireshark

Wireshark adalah *tool* yang digunakan untuk menganalisis paket data dalam sebuah kinerja jaringan. Wireshark dapat menangkap paket data atau informasi yang berada di dalam jaringan. Wireshark memiliki pengawasan paket data secara *real time*. Aplikasi wireshark dapat diakses secara gratis dan dapat dijalankan di beberapa platform seperti linux, mac dan windows [7].

2.8 Web Server

Web server merupakan software yang memberikan layanan data yang berfungsi yang menerima permintaan http ataupun https (*hypertext transfer protocol security*) dari client melalui web browser dan mengirimkan kembali hasilnya dalam bentuk halaman *web* yang umumnya berbentuk dokumen dalam format HTML [8].

2.9 HTTP (Hypertext Transfer Protocol)

HTTP Singkatan dari *Hypertext Transfer Protocol*, yang mana adalah suatu protokol yang digunakan oleh World Wide Web. HTTP mendefinisikan bagaimana suatu pesan bisa diformat dan dikirimkan dari server ke *client*. HTTP juga mengatur aksi-aksi apa saja yang harus dilakukan oleh web server dan juga web browser sebagai respon atas perintah-perintah yang ada pada protokol HTTP ini. Sebagai contoh, ketika Anda mengetikkan suatu alamat atau URL pada internet browser Anda, maka sebenarnya web browser akan mengirimkan perintah HTTP ke web server. Protokol HTTP bersifat *request-response*, yaitu dalam protokol ini client menyampaikan pesan *request* ke server dan server kemudian memberikan *response* yang sesuai dengan *request* [8].

2.10 PHP (HyperText Preprocessor)

PHP adalah singkatan dari "PHP: *Hypertext Preprocessor* ", yang merupakan sebuah bahasa *scripting* yang terpasang pada *HyperText Markup Language* (HTML). Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang *web* menulis halaman *web* dinamik dengan cepat [9].

2.11 Internet of Things (IoT)

Internet of things dalam pengertian secara luas membuat semua yang ada di dunia terkoneksi ke dalam internet yang tersambung secara terus menerus. *Internet of things* bisa mengontrol, mengirim data, dan sebagainya yang memanfaatkan internet sehingga bisa dilakukan dengan jarak jauh tanpa mengenal jarak. Konsep dasar dari *internet of things* adalah dengan menggabungkan obyek, sensor, kontroler, dan internet yang bisa menyebarkan informasi kepada pengguna. Obyek akan dideteksi oleh sensor yang akan diproses oleh kontroler dan dilanjutkan untuk mengirim data yang sudah diolah sehingga menjadi sebuah informasi yang berguna dan secara *real-time* kepada pengguna [10].

2.12 Data Rate

Data rate adalah besarnya jumlah bit yang ditransmisikan setiap detik. Dalam sistem komunikasi digital, besar nilai bandwidth sangat mempengaruhi laju bit yang dapat dikirimkan [11]. *Data rate* merupakan kecepatan transfer data efektif yang dibutuhkan agar dapat mengirimkan atau menerima data. Dapat dihitung dengan membagi besar data dengan durasi pengiriman data.

3. Pembahasan

3.1 Komponen yang Digunakan

1. Arduino Mega 2560

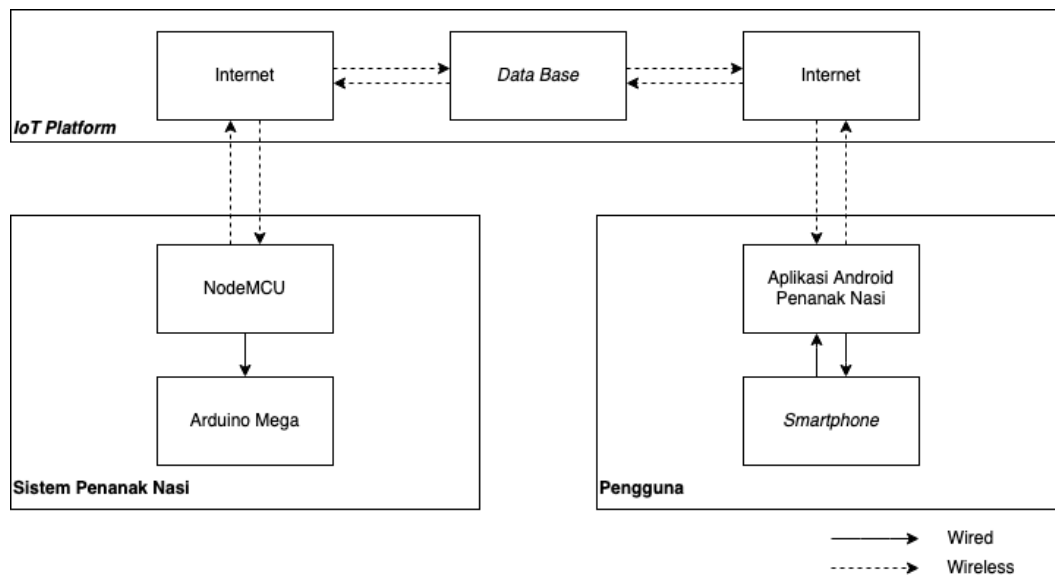
Merupakan *board* mikrokontroler berbasis ATmega 2560. Modul ini memiliki 54 digital input/output di mana pin 14 digunakan untuk PWM output dan pin 16 digunakan sebagai analog input, pin 4 untuk UART, 16 MHz osilator kristal, koneksi USB, power jack, ICSP Header, dan tombol reset. Arduino Mega 2560 memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega2560 ini menyediakan empat UART *hardware* untuk komunikasi serial.

2. NodeMCU ESP8266

Merupakan sebuah modul yang terdiri dari NodeMCU dan mikrokontroler ESP 8266. Dalam board ini NodeMCU dan ESP 8266 langsung di letakkan dalam satu tempat sehingga kita tidak perlu membelinya terpisah ataupun merangkainya lagi, ESP8266 dirancang agar WiFi terintegrasi secara langsung, sehingga ESP8266 tidak memerlukan modul WiFi dan memungkinkan untuk menjadi host ataupun sebagai WiFi *client* [12]. ESP8266 memiliki kemampuan pengolahan dan penyimpanan *on-board* yang kuat, yang memungkinkannya untuk diintegrasikan dengan sensor dan aplikasi perangkat khusus lain melalui GPIOs dengan pengembangan yang mudah serta waktu *loading* yang minimal.

3.2 Desain Sistem

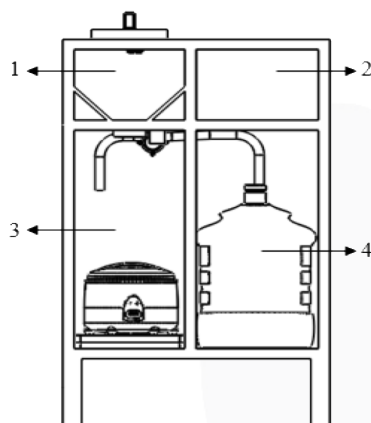
Sistem secara keseluruhan dapat direpresentasikan dengan diagram blok seperti yang ditunjukkan oleh gambar di bawah. Pengguna dapat menjalankan sistem penanak nasi melalui aplikasi *android* penanak nasi. Melalui aplikasi dapat mengirimkan data ke *database* menggunakan koneksi internet. Begitu halnya juga pada sistem penanak nasi terdapat mikrokontroler yang bertugas untuk berkomunikasi dengan *database* melalui jaringan internet sehingga sistem dapat menanak nasi sesuai data yang diterima.



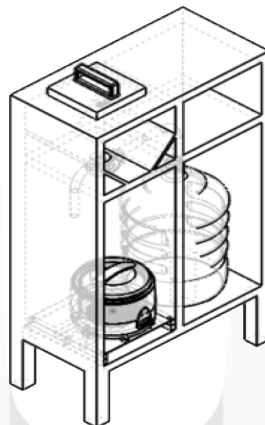
Gambar 1 Diagram Blok Sistem

3.3 Desain Perangkat Keras

Pada alat penanak nasi otomatis diimplementasikan konsep IoT menggunakan dua mikrokontroler yang akan bertugas mengirim dan menerima data melalui jaringan internet dan bertugas untuk mengontrol sistem kerja alat.



Gambar 2 Desain Rangka Alat Penanak Nasi

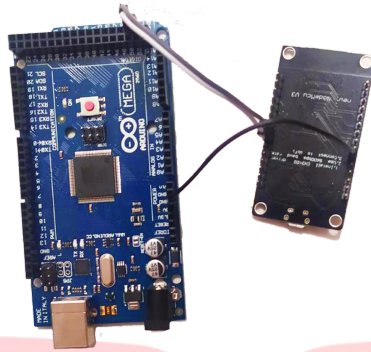


Gambar 3 Alat Penanak Nasi Otomatis

1. Tempat penyimpanan beras yang mampu menampung $\pm 13 \text{ kg}$ beras dan sensor ultrasonik yang berada di atas bagasi beras.
2. Tempat penyimpanan beberapa komponen yang berkaitan dengan elektronika sistem.
3. Tempat penanak nasi dengan kapasitas 2 liter, aktuatur linier dan sensor *load cell*.
4. Tempat galon air dengan volume tampung 19 liter.

Dari gambar rangka alat penanak nasi otomatis bagian nomor 2 menjadi tempat Arduino Mega dan NodeMCU sebagai mikrokontroler yang menjadi bagian penting agar dapat menghubungkan sistem penanak nasi dengan aplikasi penanak nasi dan memproses program menanak nasi.

Pada perancangan elektronika sistem, tegangan AC 220 V dikonversi menjadi tegangan DC 5V menggunakan adaptor. Tegangan 5 V DC sebagai catu daya PCB pada sistem penanak nasi yang telah. Dalam rancangan PCB, terdiri dari dua mikrokontroler yaitu Arduino Mega 2560 dan NodeMCU ESP8266. Kedua mikrokontroler tersebut terhubung melalui komunikasi serial RX-TX. Komunikasi serial menggunakan kabel yang terhubung pada pin kedua mikrokontroler dan diatur di dalam program.

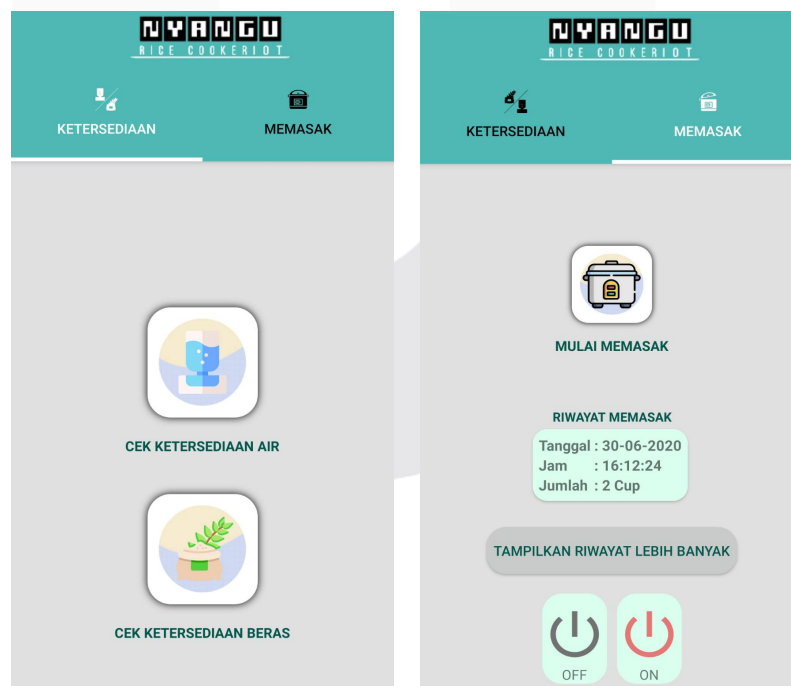


Gambar 4 Komunikasi Serial Arduino & NodeMCU

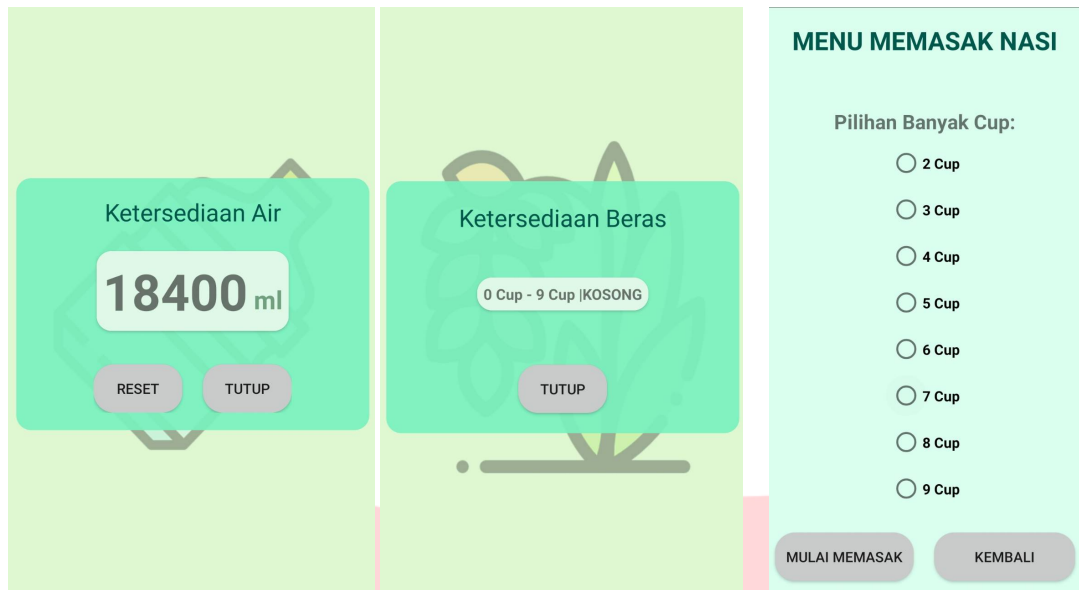
3.4 Desain Perangkat Lunak

Aplikasi penanak nasi dikhususkan pada pengguna *smartphone android* karena tingkat pengguna *smartphone android* lebih dominan jumlahnya dibandingkan sistem operasi *smartphone* lainnya. Pembuatan aplikasi menggunakan *Android Studio*. *Android Studio* sangat umum digunakan para *developer* aplikasi *android* karena memiliki banyak fitur dan fasilitas yang sangat baik untuk membuat aplikasi *android* yang lebih kompleks. Desain aplikasi dapat dilihat pada gambar di bawah.

Tampilan utama aplikasi akan memperlihatkan kepada pengguna halaman utama yang bersifat *swipe*. Halaman utama berisi menu “KETERSEDIAAN” dan menu “MEMASAK”. Pengguna ditujukan ke menu ketersediaan yang memiliki tombol untuk menuju ke tampilan pemantauan ketersediaan air dan beras sebelum ingin memasak nasi. Pada menu memasak terdapat tombol untuk mulai memasak nasi, tombol menampilkan riwayat memasak lebih banyak, tombol untuk mematikan/menghidupkan *rice cooker*, dan tampilan riwayat memasak.



Gambar 5 Tampilan Utama Aplikasi

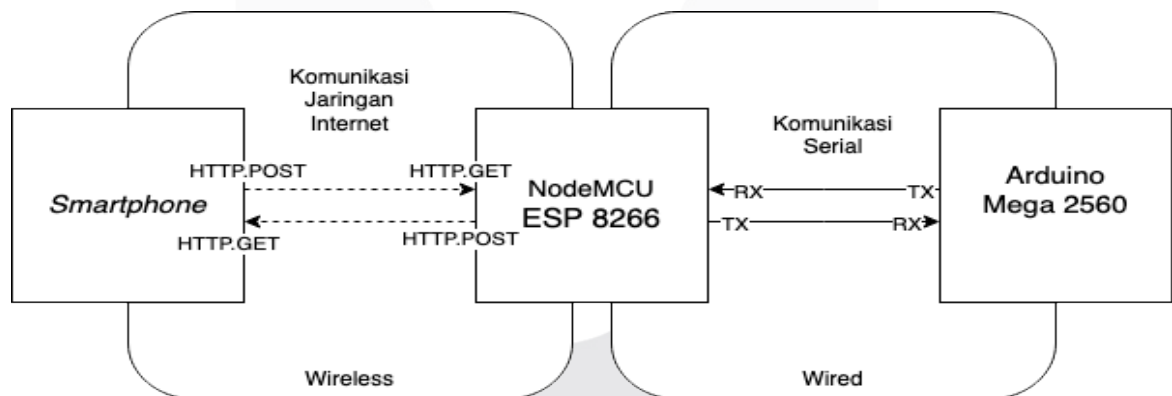


Gambar 6 Tampilan Ketersediaan Air & Beras

Gambar 7 Tampilan Menu Memasak Nasi

Fitur utama dari aplikasi ini merupakan dapat mengirimkan data banyak beras yang ingin dimasak oleh pengguna. Tombol mulai memasak berisi tampilan menu mulai memasak nasi. Terdapat pilihan banyak beras dari 2-9 cup dengan jumlah 1 cup sama dengan 300 ml. Pada tampilan Ketersediaan air, terdapat tombol untuk mereset nilai dari ketersediaan air saat galon air diisi ulang. Karena pemantauan nilai ketersediaan air didapat dari riwayat aktivitas memasak sehingga pengguna perlu untuk mereset nilai saat pengisian ketersediaan air kembali.

3.5 Desain Sistem IoT



Gambar 8 Komunikasi antara Smartphone dan Mikrokontroler

Data banyak beras yang telah dipilih akan dikirimkan ke database server menggunakan protokol http sebagai jenis komunikasi jaringan internet yang berbasis *web*. *Web* tersebut merupakan layanan dari *web hosting* yang memungkinkan penggunaan *database server*. *Web hosting* dapat berisi *file-file* yang dapat diprogram menggunakan pemrograman bahasa PHP. Dengan begitu, *file-file* tersebut akan menjadi alamat *web* yang akan dituju dengan berisikan program PHP dengan kebutuhan tertentu.

4. Hasil dan Analisis

1. Pengujian Delay dan Data Rate

Pengujian terhadap *delay* bertujuan untuk mengetahui keterlambatan dalam waktu transmisi data yang dibutuhkan dari pengiriman data banyak beras melalui aplikasi hingga penerimaan data oleh sistem penanak nasi. Pengujian dilakukan dengan mengirimkan data banyak beras dari 2-9 cup dengan masing-masing banyak cup akan dilakukan sebanyak 30 kali pengiriman. Pada saat banyak cup dipilih melalui aplikasi, waktu tekan akan disimpan di *database web server*, sedangkan waktu *Arduino Mega* menerima data banyak cup akan dilihat melalui serial monitor timestamp aplikasi *Arduino IDE*. Selisih antara waktu penerimaan data *Arduino Mega* akan dikurangkan dengan waktu tekan dari aplikasi yang merupakan menjadi nilai *delay*. Pada saat pengujian pengiriman data melalui aplikasi menggunakan koneksi internet data seluler *smartphone*, sedangkan pada sistem penanak nasi menggunakan koneksi internet dari *hotspot smartphone* tersebut.

Tabel 1 Pengujian *Delay* dan *Data Rate* Rata-Rata 2-9 Cup

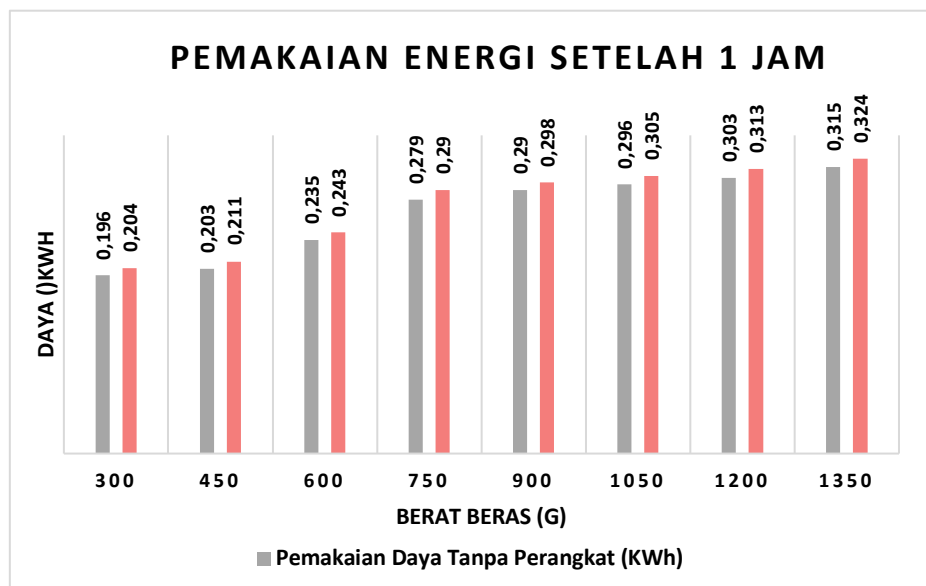
MENU	BESAR DATA (Bytes)	30 Kali Pengiriman Data			DATA RATE RATA-RATA	
		DELAY RATA-RATA (ms)	DURASI PENGUJIAN	SELANG WAKTU PENGIRIMAN DATA (s)	bit/s	Kbps
2 CUP	44	215	00:01:16.853	2.561	1637,209	1,63
3 CUP	44	252	00:01:10.673	2.355	1396,825	1,39
4 CUP	44	206	00:01:04.234	2.141	1708,737	1,7
5 CUP	44	258	00:01:11.639	2.387	1364,341	1,36
6 CUP	44	255	00:01:05.027	2.167	1380,392	1,38
7 CUP	44	232	00:01:01.417	2.047	1517,241	1,51
8 CUP	44	240	00:01:04.810	2.160	1466,666	1,46
9 CUP	44	221	00:01:06.299	2.209	1592,76	1,59
Rata-Rata Minimum Keseluruhan		206	00:01:07.619	2.047	1364,341	1,36
Rata-Rata Maksimum Keseluruhan		258	00:01:16.853	2.561	1708,737	1,7
Rata-Rata Keseluruhan		235	00:01:07.619	2.253	1508,021375	1,5025

2. Pengujian *Packet Loss*

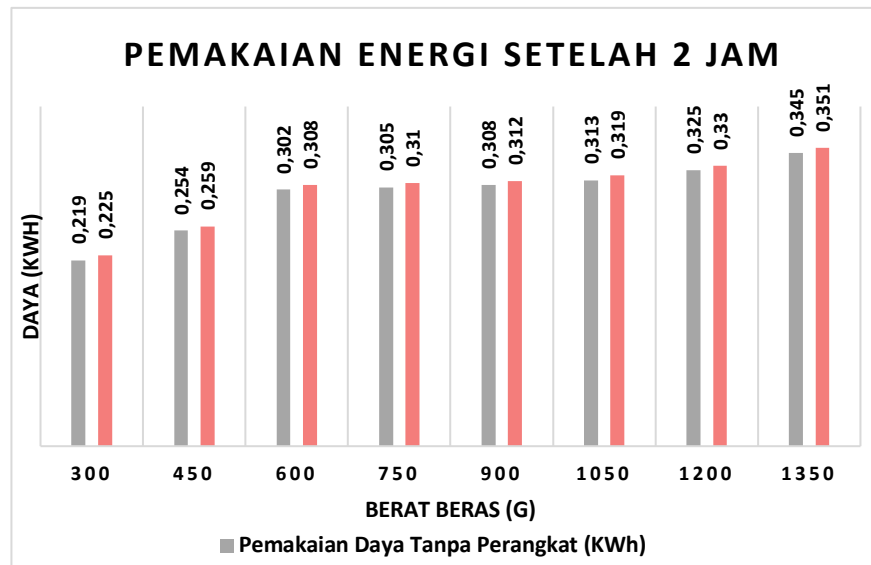
Pengujian terhadap *packet loss* dari pengiriman data banyak beras 2-9 cup tidak memiliki paket data yang hilang karena pengiriman data sebanyak 30 kali berhasil diterima semua oleh sistem dengan rentang waktu secepat mungkin yang dapat dilakukan melalui aplikasi. Sehingga *packet loss* dari setiap pengiriman data banyak beras 2-9 cup adalah 0%.

3. Pengujian Kebutuhan Energi

Pengujian kebutuhan energi bertujuan untuk mengetahui konsumsi energi yang dibutuhkan oleh sistem penanak nasi dengan menggunakan alat pengukur daya listrik. Pengujian konsumsi energi dilakukan saat alat melakukan proses memasak sejak satu jam hingga dua jam. Pengujian dilakukan dengan cara membandingkan penggunaan energi yang dibutuhkan oleh alat saat melakukan proses menanak nasi secara manual dan menanak nasi secara otomatis.



Gambar 9 Grafik Pengujian Kebutuhan Energi Saat 1 Jam



Gambar 10 Grafik Pengujian Kebutuhan Energi Saat 2 Jam

4. Kesimpulan

Berdasarkan hasil pengujian dan analisis, diperoleh kesimpulan pada Tugas Akhir ini sebagai berikut.

1. Pembuatan antarmuka aplikasi penanak nasi berbasis IoT telah berhasil dibuat dengan memiliki tampilan menu mengatur banyak beras 2-9 cup, tampilan memantau ketersediaan air dan beras, dan estimasi waktu memasak nasi.
2. Pengiriman data banyak beras 2-9 cup dari aplikasi menghasilkan *delay* rata-rata sebesar 235 ms sehingga masuk Kategori *Delay* baik Menurut TIPHON.
3. *Data rate* rata-rata dari pengiriman data banyak beras 2-9 cup adalah 1,5 Kbps.
4. Pengiriman data sebanyak 30 kali berhasil dilakukan sehingga tidak ada *packet loss* dari pengiriman data banyak cup beras 2-9 cup.
5. Kebutuhan energi rata-rata dari sistem setelah 1 jam pertama adalah 0,008875 kWh dan setelah 2 jam pertama adalah 0,005375 kWh.

Daftar Pustaka

- [1] Subekti, "Singapura Terbanyak Konsumsi Karbohidrat di ASEAN," Tempo.co, 22 6 2015. [Online]. Available: <https://nasional.tempo.co/read/677310/singapura-terbanyak-konsumsi-karbohidrat-di-asean/full&view=ok>. [Diakses 1 12 2019].
- [2] Syanne Susita, "Tips Bagaimana Menanak Nasi Hingga Pulen," CNN Indonesia, 26 5 2017. [Online]. Available: <https://www.cnnindonesia.com/gaya-hidup/20170526082943-267-217360/tips-bagaimana-menanak-nasi-hingga-pulen>. [Diakses 1 12 2019].
- [3] Simon Kemp, "Digital 2020: Indonesia," DATAREPORTAL, 18 February 2020. [Online]. Available: <https://datareportal.com/reports/digital-2020-indonesia>. [Diakses 26 6 2020].
- [4] S. Arifin, "ANALISA KUALITAS LAYANAN THROUGHPUT HANDPHONE DAN MODEM HIGH SPEED DOWNLINK PACKET ACCES (HSDPA)," *Jurnal Teknik Elektro Universitas Tanjungpura*, vol. 1, no. 1, p. 2, 2014.
- [5] H. Kusniyati dan N. S. P. Sitanggang, "APLIKASI EDUKASI BUDAYA TOBA SAMOSIR BERBASIS ANDROID," *JURNAL TEKNIK INFORMATIKA*, vol. 9, no. 1, p. 12, 2016.
- [6] A. Giyartono dan P. E. Kresnha, "APLIKASI ANDROID PENGENDALI LAMPU RUMAH BERBASIS MIKROKONTROLER ATMEGA328," *Jurnal Fakultas Teknik Universitas Muhammadiyah Jakarta*, p. 2, 2015.
- [7] F. A. Afrida dan S. Rahmatia, "Analisis Internet Group Management Protocol (IGMP) Menggunakan Software Wireshark dalam Layanan Live Streaming IPTV pada Multi Service Access Network (MSAN) di Area Darmo, Surabaya," *AL-AZHAR INDONESIA SERI SAINS DAN TEKNOLOGI*, vol. 4, no. 4, p. 178, 2018.

- [8] T. Evi dan Malabay, "ANALISIS PENGEMBANGAN APLIKASI WEB UNTUK PROFIL PERUSAHAAN," dalam *Seminar Nasional Informatika 2009*, Yogyakarta, 2009.
- [9] M. Jaenuri, "PERANCANGAN PENGOLAHAN DATA PERPUSTAKAAN MADRASAH ALIYAH NEGERI LASEM BERBASIS INTRANET," *Indonesian Journal on Networking and Security*, vol. 2, no. 3, p. 51, 2013.
- [10] A. R. Agusta, J. Andjarwirawan dan R. Lim, "Implementasi Internet of Things Untuk Menjaga Kelembaban Udara Pada Budidaya Jamur," *Jurnal Infra*, vol. 7, no. 2, p. 2, 2019.
- [11] B. S. K. Sakti, A. Fahmi dan V. S. W. Prabowo, "Analisis Performansi Alokasi Sumber Daya Radio Berbasis Algoritma Greedy pada Sistem Komunikasi D2d Underlying Analysis Performance Radio Resource Allocation with Greedy Algorithm In D2dD Underlying Communication," *Seminar Nasional Teknik Elektro UIN Sunan Gunung Djati Bandung*, Bandung, 2019.
- [12] I. Gunawan, T. Akbar dan M. Ilham, "Prototipe Penerapan Internet Of Things(Iot) Pada Monitoring Level Air Tandon Menggunakan Nodemcu Esp8266 Dan Blynk," *Jurnal Informatika dan Teknologi*, vol. 3, no. 1, p. 3, 2020.

