



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Automasi

Automasi adalah teknologi yang mampu menjalankan aktivitas secara otomatis dan berkelanjutan, sesuai dengan algoritma pemrograman yang sebelumnya telah diperintahkan oleh pengembang (Fridman et al., 2019)

Automasi merupakan penerapan teknologi yang memungkinkan sistem untuk menjalankan tugas-tugas tertentu secara otomatis, dengan sedikit atau tanpa intervensi manusia. Dalam konteks teknologi berbasis IoT automasi menjadi lebih canggih karena perangkat yang terhubung dapat berkomunikasi dan bertukar data satu sama lain melalui jaringan internet. IoT memungkinkan automasi untuk menjadi lebih efisien, responsif, dan adaptif terhadap kondisi lingkungan atau kebutuhan pengguna (Gubbi et al., 2019).

Automasi berbasis IoT menawarkan berbagai manfaat bagi layanan *laundry*. Pertama, sistem ini dapat mengurangi kesalahan manusia dan meningkatkan konsistensi dalam proses pencucian. Kedua, melalui pemantauan dan pengendalian jarak jauh, pengelola laundry dapat mengawasi operasi mesin dari mana saja dan kapan saja, yang berpotensi meningkatkan produktivitas. Ketiga, sistem IoT dapat memberikan notifikasi kepada pengguna mengenai status cucian, seperti kapan cucian selesai atau jika ada gangguan, sehingga meningkatkan kepuasan pelanggan (Makadam et al., 2019).

2.2 Internet of Things (IoT)

Internet of Things (IoT) adalah sistem *embedded* yang bertujuan memperluas pemanfaatan konektivitas internet yang tersambung secara terus-menerus, sehingga memungkinkan berbagi data, *remote control*, dan berbagai kemampuan lainnya pada benda-benda di dunia nyata, seperti bahan pangan, perangkat elektronik, dan peralatan yang dilengkapi dengan sensor serta terhubung ke jaringan (Susanto et al., 2022).

Internet of Things (IoT) merujuk pada jaringan perangkat fisik yang dilengkapi dengan teknologi komunikasi untuk saling berhubungan dan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

bertukar data melalui internet. Perangkat-perangkat ini dapat mencakup berbagai objek mulai dari alat rumah tangga hingga mesin industri, yang memiliki sensor dan kemampuan untuk mengumpulkan, mengirim, dan menerima data. IoT memungkinkan objek-objek tersebut untuk berinteraksi dengan lingkungan mereka dan melakukan tugas secara otomatis tanpa intervensi manusia, yang membuat sistem menjadi lebih efisien dan responsif (Khan et al., 2020).

Implementasi IoT dalam layanan *laundry* berbasis automasi memungkinkan integrasi antara perangkat dan sistem manajemen yang lebih baik. Sistem dapat diatur untuk melakukan tugas secara otomatis berdasarkan data yang dikumpulkan oleh sensor, seperti menentukan waktu yang optimal untuk pencucian dan pengeringan, serta mengelola penggunaan sumber daya seperti air dan listrik secara efisien. Selain itu, IoT memungkinkan pengguna untuk memantau proses *laundry* mereka melalui aplikasi *mobile*, memberikan kemudahan dan fleksibilitas yang lebih tinggi (Zhou et al., 2022).

2.3 Laundry

Laundry adalah layanan mencuci pakaian dan kain dengan menggunakan air, deterjen, pelembut, serta pewangi. Layanan ini tidak hanya berfungsi sebagai tempat mencuci pakaian, tetapi juga sebagai tempat untuk merawat pakaian agar lebih bersih dan tahan lama. Selain itu, *laundry* membantu mempermudah pekerjaan seseorang di tengah lingkungan yang sibuk dan waktu yang terbatas, karena aktivitas sehari-hari yang padat sering kali membuat mereka tidak memiliki waktu untuk mencuci pakaian dan sejenisnya (Sari et al., 2021).

Dalam beberapa tahun terakhir, layanan *laundry* telah beradaptasi dengan perkembangan teknologi untuk meningkatkan efisiensi dan kualitas layanan. Penggunaan mesin cuci otomatis dan pengering modern telah memungkinkan penyedia layanan *laundry* untuk mengolah pakaian dengan lebih cepat dan efisien. Lebih jauh lagi, integrasi teknologi seperti IoT ke dalam layanan *laundry* telah membuka peluang untuk pengelolaan operasi yang lebih cerdas dan terhubung, di mana proses pencucian dan pengeringan dapat dipantau dan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dikendalikan secara otomatis (Al-Qudah et al., 2019).

Automasi berbasis IoT dalam layanan laundry memberikan banyak manfaat, termasuk peningkatan efisiensi operasional, penghematan waktu, dan peningkatan kepuasan pelanggan. Sistem yang diotomatisasi dapat meminimalkan kesalahan manusia, mengoptimalkan penggunaan sumber daya, dan menyediakan layanan yang lebih cepat dan akurat. Selain itu, integrasi IoT memungkinkan layanan *laundry* untuk menawarkan fitur-fitur canggih seperti notifikasi status cucian, pembayaran otomatis, dan analisis data untuk memahami pola penggunaan pelanggan (Martinez et al., 2022).

2.4 LCD Touchscreen Nextion

LCD *Nextion* adalah *human machine interface* (HMI) yang dilengkapi dengan *capacitive touchscreen*. LCD *Nextion* berfungsi untuk menampilkan grafik display sampel (Ikbal et al., 2019).

LCD *Nextion* dapat digunakan sebagai antarmuka pengguna yang interaktif dan user-friendly. Dengan adanya layar sentuh kapasitif, pengguna dapat dengan mudah berinteraksi dengan sistem automasi laundry, seperti memilih jenis layanan, mengatur waktu, dan memonitor status cucian (Pratama et al., 2021). Gambar 2.1 merupakan jenis-jenis ukuran LCD *Nextion*.



Gambar 2.1 LCD *Nextion*

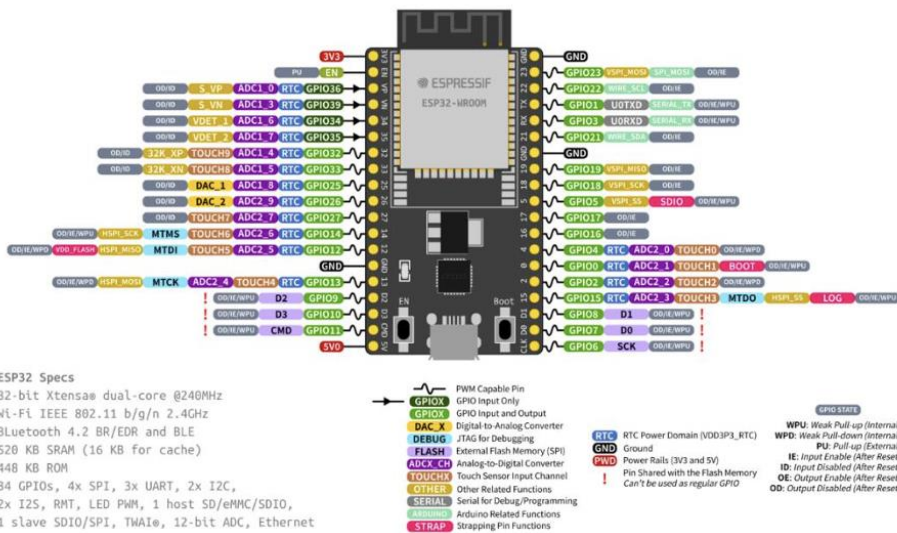
Sumber: (<https://nextion.tech/basic-series-introduction>)



2.5 Modul ESP32

Mikrokontroler ESP32 merupakan mikrokontroler SoC (*System on Chip*) terpadu dengan dilengkapi WiFi 802.11 b/g/n, Bluetooth versi 4.2, dan berbagai *peripheral*. ESP32 adalah chip yang cukup lengkap, terdapat prosesor, penyimpanan dan akses pada GPIO (*General Purpose Input Output*). ESP32 bisa digunakan untuk rangkaian pengganti pada Arduino, ESP32 memiliki kemampuan untuk mendukung terkoneksi ke WI-FI secara langsung (Wagyan, 2019). Gambar 2.1 merupakan kaki pin pada ESP32.

ESP32-DevKitC



Gambar 2.2 Pin Out ESP32-DevKitC

Sumber: (docs.espressif.com)

Deskripsi untuk ESP32-DevKit C dapat dilihat pada Tabel 2.1

Tabel 2.1 Deskripsi ESP32-DevKit C

Kategori	Komponen	Spesifikasi
Wi-Fi	Protocols signal	802.11 b/g/n (802.11n up to 150 Mbps) Agregasi A-MPDU dan A MSDU serta dukungan interval penjaga 0,4 μ s.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Kategori	Komponen	Spesifikasi
Hardware	Jangkauan frekuensi tengah dari saluran operasi	2412 ~ 2484 MHz
	Integrasi <i>SPI flash</i>	4 MB
	Integrasi <i>Crystal (clock speed)</i>	40 MHz
	<i>Operating power supply/voltage</i>	3.0 V ~ 3.6 V
	<i>Operating current</i>	Rata-rata: 80 mA
	Minimum <i>current</i> dari <i>power supply</i>	500 mA
	Rekomendasi operasi rata-rata jangkauan temperatur	-40 °C ~ +85 °C
	Dimensi	18 mm x 25,5 mm x 3.10 mm
	<i>Moisture Sensitivity Level (MSL)</i>	Level 3

2.6 Website

Website merupakan kumpulan halaman yang saling terkait, dengan di hubungkan melalui link agar menjadi suatu kesatuan yang menampilkan informasi berupa teks, gambar, animasi, audio, video, ataupun kombinasi semuanya. Pada saat mendesain *website* dikenal dua jenis yaitu UI (*User Interface*) yang fokus pada bagaimana tampilan sebuah *website* menarik dan UX (*User Experience*) yang fokus pada bagaimana sebuah *website* mudah di gunakan oleh pengguna (*user friendly*) (Susanto, 2023).

2.7 Power Supply 5-9V

Power supply dengan rentang tegangan 5V sampai 9V merupakan komponen penting dalam sistem automasi berbasis IoT untuk layanan *laundry*. *Power supply* ini berfungsi untuk menyediakan daya yang stabil dan konsisten bagi komponen-komponen elektronik dalam sistem, seperti mikrokontroler, sensor, dan *aktuator* (Pratama et al., 2022).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Dengan menggunakan *power supply* yang sesuai, sistem automasi dapat bekerja dengan optimal dan terhindar dari kerusakan akibat fluktuasi tegangan yang tidak stabil (Sari et al., 2021). Gambar 2.3 merupakan contoh adaptor.



Gambar 2.3 Adaptor *Universal AC to DC*

Sumber: (<https://www.alibaba.com/showroom/9-volt-adapter>)

2.8 Stepdown LM2596

LM2596 adalah regulator tegangan yang berfungsi untuk menurunkan tegangan dari sumber daya ke level yang sesuai dengan kebutuhan komponen-komponen elektronik dalam sistem. Modul regulator tegangan yang umumnya digunakan dalam proyek elektronika untuk menurunkan tegangan *input* menjadi tegangan *output* yang lebih rendah secara stabil. Ini berguna dalam situasi di mana sumber daya memiliki tegangan yang lebih tinggi dari yang dibutuhkan oleh komponen elektronik yang akan digunakan. (Pratama et al., 2023). Gambar 2.4 merupakan modul *stepdown* LM2596.



Gambar 2.4 Modul *Stepdown* DC LM2596

Sumber: (<https://ecadio.com/jual-modul-step-down-dc-lm2596>)

2.9 Firebase

Firebase merupakan layanan yang disediakan oleh Google untuk memberikan kemudahan kepada para pengembang aplikasi dalam



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mengembangkan aplikasinya. Dengan memanfaatkan *Firebase*, pengembang aplikasi dapat lebih fokus dalam mengembangkan aplikasi tanpa harus mengeluarkan usaha yang besar untuk menangani masalah *backend*, sehingga proses pengembangan aplikasi menjadi lebih efisien dan cepat (Dicoding Intern., 2020). Gambar 2.5 merupakan logo dari *firebase*.



Gambar 2.5 Logo *Firebase*

Sumber: (<https://firebase.google.com>)

2.10 Gerbang Logika Dasar dan Kombinasi

Sistem digital dimulai dengan gerbang logika dasar, yang juga dikenal sebagai gerbang logika *biner*, yang sering ditemukan dalam sirkuit digital yang dioperasikan secara elektronik dengan diode atau transistor (Helmi & Ikhsan, 2017).

Gerbang logika terdiri dari gerbang dasar :

1. Gerbang AND (Gerbang perkalian)
2. Gerbang OR (Gerbang penjumlahan)
3. Gerbang NOT (Gerbang Invers atau kebalikan)

Percabangan dalam bahasa C terdiri dari beberapa percabangan yaitu yang pertama Percabangan If yang merupakan sebuah percabangan yang digunakan saat terdapat satu pilihan keputusan. Lalu ada Percabangan If/Else yang digunakan ketika terdapat 2 pilihan keputusan. Dan yang terakhir ada Percabangan If/Elif/Else yang digunakan apabila terdapat lebih dari dua pilihan keputusan.

Perulangan dalam sebuah pemrograman berfungsi untuk menyuruh komputer melakukan suatu secara berulang – ulang. Struktur perulangan yang terdapat pada bahasa C yaitu perulangan for dan while (Wijaya et al, 2021).

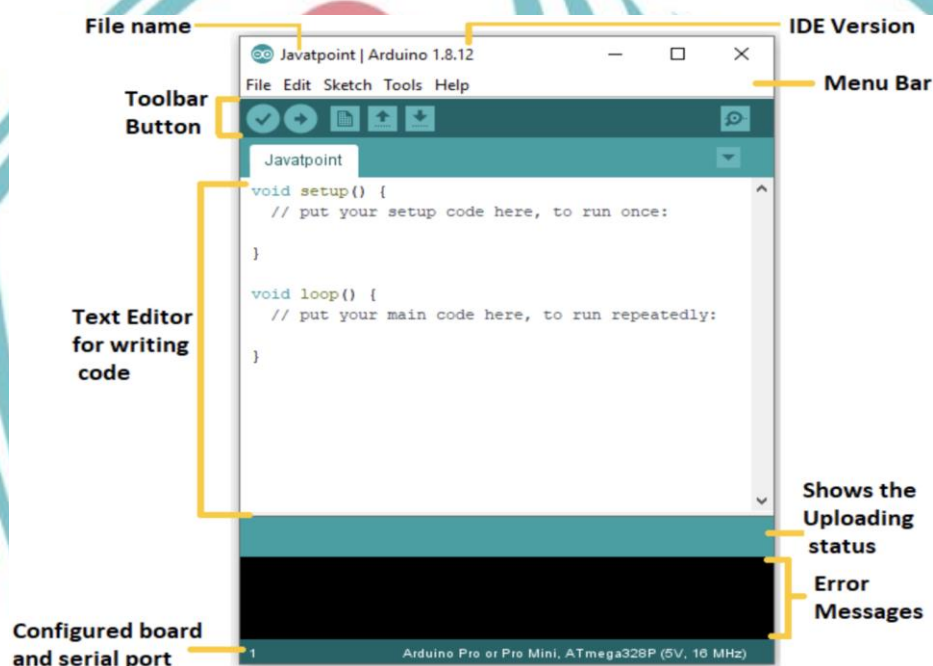


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2.11 Arduino IDE

Integrated Development Environment (IDE). IDE adalah perangkat lunak yang sangat penting untuk pemrograman, kompilasi biner, dan unduhan memori mikrokontroler. Arduino telah menjadi platform yang populer bagi banyak profesional karena memiliki banyak modul pendukung seperti sensor, monitor, pembaca, dll. Arduino sangat disukai karena *hardware* dan softwarena *open source*. Semua orang dapat mendapatkan skema Arduino secara gratis (Santoso & Wijayanto, 2022). Gambar 2.6 merupakan tampilan dari Arduino IDE.



Gambar 2.6 Arduino IDE

Sumber: (<https://www.javatpoint.com/arduino-ide>)



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB III

PERENCANAAN DAN REALISASI

3.1 Rancangan Alat

Dalam usaha *laundry*, penghitungan biaya berdasarkan berat cucian seringkali menjadi tantangan bagi pemilik usaha. Untuk mengatasi masalah tersebut, dirancanglah sebuah sistem perhitungan timbangan berbasis *Internet of Things* (IoT) dan *Website*. Alat ini dirancang dengan menggunakan dua buah sensor sebagai *input* utama. Pertama, sensor *Loadcell* yang berfungsi untuk mengukur kapasitas atau berat barang yang akan ditimbang. Kedua, sensor *LCD Touchscreen*. Sebagai otak dari sistem ini, digunakan modul mikrokontroler *ESP32* yang berfungsi sebagai modul *Wi-Fi*. Pembahasan lebih lanjut mengenai cara kerja alat yang digunakan sebagai berikut:

A. Deskripsi Alat

Alat yang dirancang adalah sebuah sistem perhitungan timbangan pada usaha *laundry* berbasis *Internet of Things* (IoT) dan *Website* yang ditampilkan data *ter-update* dari pelanggan yang di tampilkan pada halaman *website* dan *admin* juga dapat memverifikasi status pemesanan dan bisa mengirimkan struk digital ke *email* pelanggan. Alat ini menggunakan 2 buah sensor sebagai *input* yaitu sensor *Loadcell* yang akan mengukur kapasitas dari barang yang akan ditimbang dan lalu jika barang yang di timbang akan dikalkulasi dan menghasilkan jumlah yang harus dibayar lalu akan tertampil pada *LCD Touchscreen* yang akan membantu menampilkan jumlah yang harus dibayar, kapasitas dan memilih layanan yang akan dipilih selanjutnya Modul *ESP32* digunakan sebagai mikrokontroler dalam sistem sebagai modul *WI-FI* yang akan mengirimkan data dari timbangan IoT ke *website*, Lalu *input* data dari *hardware* akan dimasukan *input website* masuk kedalam *query javascript* lalu *query* akan dipaketkan melalui di *mysql*. Data dari *sql* ditarik melalui *firebase* menggunakan *javascript*. Kemudian, terdapat LED sebagai indikator apakah penggunaan sudah meng *input* status layanan, *ESP32* berfungsi sebagai *Bluetooth* yang berfungsi menghubungkan *hardware* dengan mesin *printer*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mini, lalu mesin *printer* mini akan mencetak setiap data yang masuk dari *hardware* melalui ESP32.

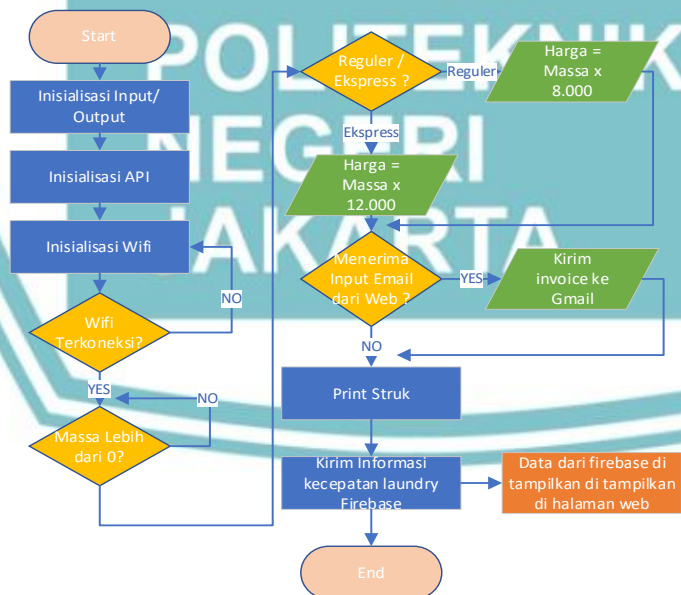
Berdasarkan Gambar 3.1 memperlihatkan ilustrasi alat pada penelitian berikut:



Gambar 3.1 (a) Tampak Depan Alat (b) Tampak Atas Alat

B. Cara Kerja Alat

Alat timbangan *laundry* otomatis berbasis *Internet of Things* (IoT) yang dirancang memiliki cara kerja yang cukup komprehensif. Proses dimulai ketika calon pelanggan menempatkan pakaian kotor di atas timbangan yang tersedia. Cara kerja alat dapat dilihat pada Gambar 3.2 dalam bentuk diagram alir berikut.



Gambar 3.2 Diagram Alir Alat

Berdasarkan Gambar 3.2 Sensor *Loadcell* pada timbangan akan mengukur berat dari pakaian kotor tersebut, dan hasil pengukuran akan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

ditampilkan pada layar LCD *touchscreen*. Selanjutnya, pada layar *touchscreen* akan muncul kotak *input* untuk memasukkan nama pelanggan. Setelah nama pelanggan diinput, akan muncul opsi untuk memilih layanan *laundry*, yaitu Reguler atau Ekspres. Pelanggan dapat memilih layanan yang diinginkan melalui tombol yang tersedia di layar *touchscreen*, lalu menekan tombol "OK" untuk melanjutkan proses.

Setelah pelanggan menekan tombol "OK", informasi mengenai berat pakaian, nama pelanggan, dan layanan yang dipilih akan dikirimkan ke *database*. Informasi ini juga akan dikirimkan ke modul *printer* mini yang terhubung dengan sistem, untuk mencetak struk transaksi. Struk transaksi akan berisi informasi seperti nama pelanggan, tanggal *laundry*, berat pakaian kotor, layanan yang dipilih, dan harga tagihan yang harus dibayar oleh pelanggan saat mengambil pakaian yang telah selesai dicuci.

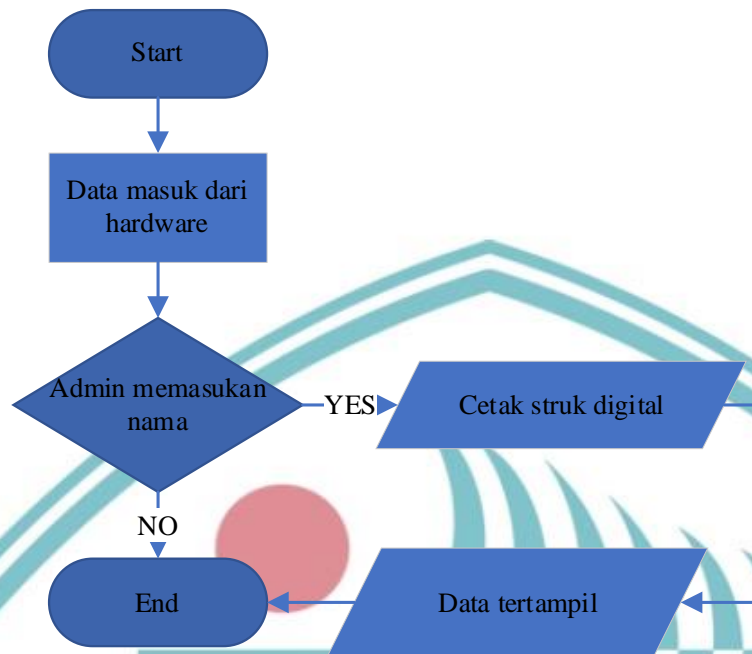
Selain itu, pemilik usaha *laundry* dapat mengakses *website* admin untuk melihat history transaksi *laundry*. *Website* admin menampilkan informasi dalam bentuk tabel, yang mencakup nama pelanggan, tanggal *laundry*, berat pakaian kotor, layanan yang dipilih, dan harga tagihan. Pada setiap baris tabel, terdapat tombol yang dapat digunakan oleh admin untuk menandai apakah pakaian pelanggan sudah selesai dicuci atau belum. Dengan alat timbangan *laundry* otomatis berbasis IoT ini, proses pengukuran, pemilihan layanan, pencatatan transaksi, dan monitoring status pencucian dapat dilakukan secara lebih efisien dan transparan, sehingga memberikan pengalaman yang lebih baik bagi pelanggan dan memudahkan pengelolaan usaha *laundry* bagi pemiliknya.

Proses penerimaan data dari alat dapat dilihat pada Gambar 3.2 yang berisikan proses yang terjadi pada *website*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 3.3 Diagram Alir Website

Berdasarkan Gambar 3.3 di atas menggambarkan sebuah proses otomatis yang dimulai dari penerimaan data masukan dari perangkat keras. Tahap awal proses ini adalah "Data masuk dari *hardware*", di mana sistem menerima data-data masukan yang berasal dari perangkat keras terkait.

Selanjutnya, proses dilanjutkan dengan meminta admin untuk memasukkan namanya melalui langkah "Admin memasukkan nama". Sistem akan memeriksa apakah *admin* telah memasukkan nama atau belum melalui tahap "YES/NO". Jika admin belum memasukkan nama, maka sistem akan mengarahkan kembali ke langkah sebelumnya agar admin dapat memasukkan namanya.

Setelah *admin* berhasil memasukkan namanya, sistem akan melanjutkan proses dengan mencetak struk digital yang berisi informasi transaksi melalui langkah "Cetak struk digital". Informasi-informasi yang tercantum dalam struk digital ini dapat berupa nama pelanggan, tanggal transaksi, jumlah barang, jenis layanan, dan total biaya yang harus dibayarkan.

Proses berikutnya adalah menampilkan data yang telah diterima sebelumnya melalui langkah "Data tertampil". Pada tahap ini, sistem akan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

menampilkan data-data yang sebelumnya telah diterima dan diproses, sehingga dapat dilihat dan dipantau oleh pengguna.

Setelah seluruh tahapan proses selesai dilakukan, diagram *alir* ini akan berakhir pada langkah "End". Dengan demikian, alur proses yang tergambar dalam diagram ini dimulai dari penerimaan data, verifikasi pengguna, pencetakan struk digital, hingga penampilan data, yang dijalankan secara otomatis oleh sistem.

C. Spesifikasi Alat

Berikut ini ditunjukkan pada Tabel 3.1 merupakan spesifikasi dari setiap komponen yang digunakan.

Tabel 3.1 Spesifikasi Komponen

Jenis	Komponen	Spesifikasi
Timbangan Digital	QM-5140	<i>Max Weight</i> : 40kg <i>Min Weight</i> : 200g G.W : 3.0 kg N.W : 2.5 kg Dimensi : 354 x 127 x 378 mm
LCD Touchscreen	Nextion NX4832F035	3.5" LCD-TFT HMI Display, STM32GO ARM® Cortex®-M0+ 64MHz Onboard MCU Processor, Fonts Creator with Versatile Character Encodings, Low Learning Curve Text-based Instruction Set, Support 15+ Amazing WYSIWYG Components, New Nextion Cost-effective HMI Solution



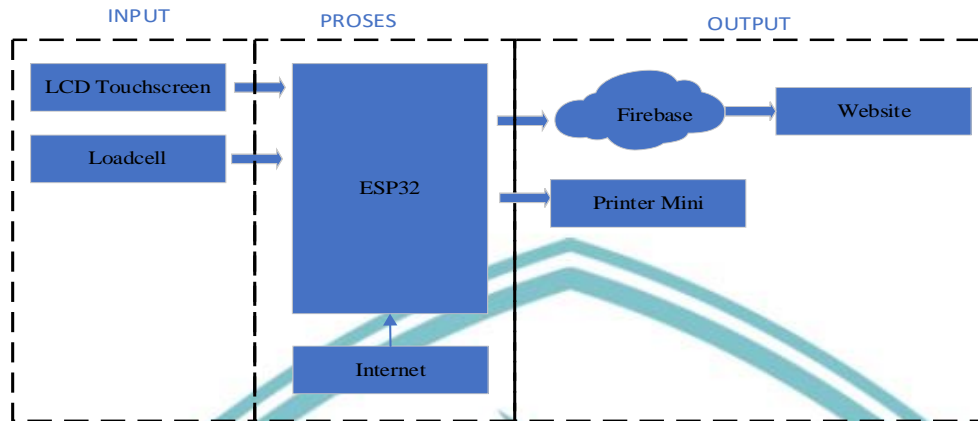
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Jenis	Komponen	Spesifikasi
Mikrokontroler	ESP 32 DivKit C V4	GPIO, 802.11 b/g/n (802.11n up to 150 Mbps) 3.3V – 5V, Output
Catu daya	Universal ac to dc adapter power supply	Input : 100-240V 50/60Hz 700mA Output : 24V 0.5A, 12V 1A, 5V 1A, 5V 0.5V, 9V 1A, 6V 1A.
Regulator tegangan	Stepdown LM2596	Input voltage : DC 3V-40V Output voltage : DC 1.5V- 35V Arus max : 3 A Size : 42 mm x 20 mm x 14 mm
Printer Thermal	Digit up DT-58D	Support android & iphone via bluetooth, Support windows via USB, Power using electricity (without battery), Has RJ11 port to connect to cash drawer, Paper size 58mm

D. Diagram Blok Sistem

Penelitian ini bertujuan untuk melihat bagaimana proses penghitungan biaya berdasarkan berat cucian. Diagram blok penelitian ini menyajikan rangkaian langkah-langkah yang dijalankan dalam penelitian. Diagram blok penelitian dapat dilihat pada Gambar 3.3.



Gambar 3.4 Diagram Blok Alat

Berdasarkan Gambar 3.4 Diagram blok di atas menggambarkan alur proses dari sebuah sistem yang terdiri dari tiga komponen utama, yaitu *input*, *proses*, dan *output*.

Pada bagian *input*, terdapat dua komponen:

1. *LCD*: Komponen ini berfungsi sebagai tampilan yang menampilkan informasi kepada pengguna.
2. *Loadcell*: Komponen ini berfungsi untuk mengukur berat atau massa dari suatu objek yang diletakkan di atasnya.
3. Pada bagian *proses*, terdapat satu komponen utama, yaitu ESP32. Komponen ini merupakan sistem mikroprosesor yang berfungsi sebagai otak dari sistem ini. ESP32 akan menerima data *input* dari *Loadcell* dan *LCD*, lalu memproses data tersebut sesuai dengan algoritma yang telah diprogram.

Pada bagian *output*, terdapat tiga komponen:

1. *Firebase*: Komponen ini berfungsi sebagai media penyimpanan data yang dihasilkan oleh sistem. Data yang disimpan di *Firebase* dapat diakses melalui Internet.
2. *Blynk Email*: Komponen ini berfungsi untuk mengirimkan *email* kepada pengguna terkait dengan aktivitas atau status dari sistem.
3. *Printer Mini*: Komponen ini berfungsi untuk mencetak struk atau bukti transaksi yang dilakukan oleh pengguna.

Secara keseluruhan, diagram blok ini menggambarkan sebuah sistem yang dapat menerima *input* dari sensor *Loadcell* dan *LCD*, memproses data



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

tersebut menggunakan ESP32, lalu menyimpan data di Firebase, mengirimkan *email* melalui *Blynk Email*, dan mencetak struk menggunakan *Printer Mini*.

3.2 Realisasi Alat

Proyek ini bertujuan untuk merancang dan membangun sebuah sistem automasi *laundry* yang memanfaatkan teknologi *Internet of Things* (IoT). Sistem ini dikembangkan untuk meningkatkan efisiensi dan kualitas layanan pada bisnis jasa laundry. Berikut pada Gambar 3.5 merupakan realisasi alat pada penelitian ini.



Gambar 3.5 Realisasi Akhir Alat

Berdasarkan gambar di atas terdapat *printer thermal* berwarna biru yang digunakan untuk mencetak struk atau *output*. Di sebelah kiri terdapat timbangan besar yang digunakan untuk mengukur berat barang atau objek. Perangkat LCD layar sentuh yang berfungsi sebagai antarmuka interaktif untuk mengoperasikan sistem tersebut. Pengguna dapat memasukkan data atau memberikan perintah melalui layar sentuh ini, sehingga seluruh sistem dapat berjalan dengan baik, mulai dari penimbangan, pencatatan, hingga pencetakan *output*.

Proses realisasi alat ini diawali dengan identifikasi permasalahan pada layanan *laundry* konvensional, seperti kesulitan dalam memantau status pesanan,



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

kesalahan dalam pencatatan, dan kurangnya transparansi informasi bagi pelanggan. Tampilan *website* yang menarik dan intuitif mencerminkan komitmen mereka untuk memberikan layanan yang serba digital dan *user-friendly*. Pengguna dapat dengan mudah melacak status pesannya, mulai dari "Diambil", "Selesai", hingga "Dikirim Kembali", dilengkapi dengan indikator visual yang jelas.

Tidak hanya itu, platform ini juga menawarkan berbagai opsi layanan, mulai dari cuci biasa, cuci kering, hingga setrika, sehingga pelanggan dapat memilih paket yang sesuai dengan kebutuhan mereka. Semua informasi tentang layanan, harga, dan status pesanan dapat diakses dengan mudah melalui platform digital ini. Berikut ini merupakan tampilan *webiste* yang dapat dilihat sebagai berikut.

No	ID	Nama	Harga	Jumlah	Berat	Layanan	Delete
1	406	zack	10994.60	9	1.10	Cepat	Delete
2	407	gerry	25000.00	18	2.50	Cepat	Delete
3	408	juminten	33200.00	20	3.32	Cepat	Delete
4	413	pahakim	21300.00	16	2.13	Cepat	Delete
5	415	mutia	16700.00	11	1.67	Cepat	Delete
6	419	yusuf	12960.00	13	1.44	Sedang	Delete
7	420	ghozi	9590.00	16	1.37	Biasa	Delete
8	421	zhafran	19283.37	21	2.14	Sedang	Delete
9	422	hardi	18224.64	24	2.60	Biasa	Delete

Gambar 3.6 Tampilan *Website*

Berdasarkan Gambar 3.6 tampilan *website* yang ditampilkan, ini adalah sebuah layanan pencucian dan pembersihan pakaian (*laundry*) yang bernama "*Wash Clear*". Beberapa hal yang dapat dijelaskan dari tampilan *website* tersebut:



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1. Tampilan dan Navigasi:

Halaman ini terdiri dari tiga bagian utama: Utama, Antrian, dan Status. Desain menggunakan warna-warna cerah seperti biru dan pink, memberikan kesan modern dan menarik.

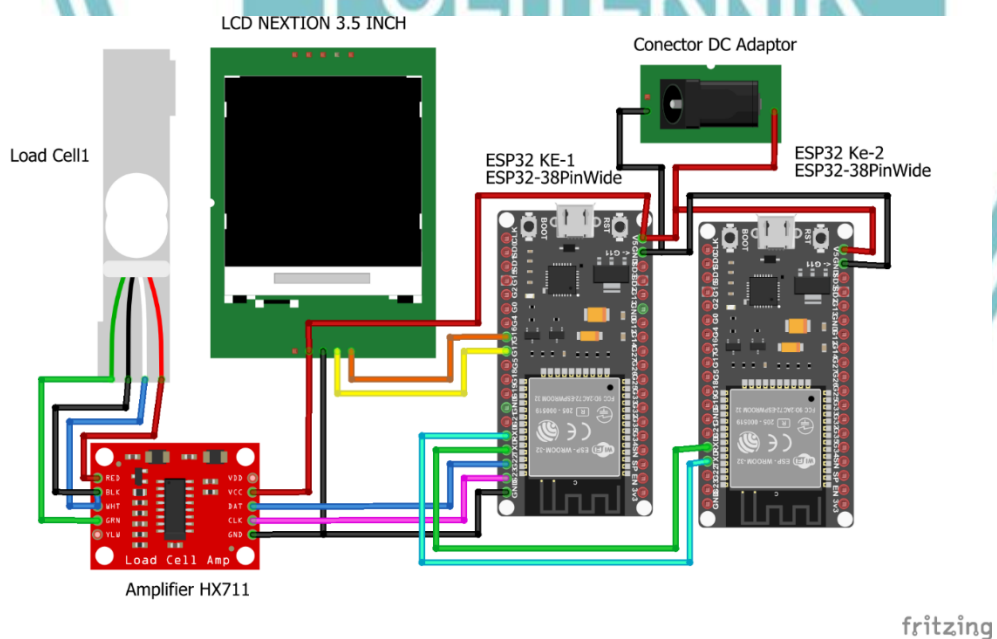
2. Fitur Utama:

Pada bagian "Utama", terdapat tombol "Registrasi" yang memungkinkan pengguna untuk mendaftar atau membuat akun baru. Pada bagian "Antrian", terdapat daftar nama, tanggal, status, dan lokasi untuk pesanan cucian yang sedang diproses. Pada bagian "Status", terdapat informasi rinci tentang status pesanan, termasuk nama, tanggal, status (Diambil atau Selesai), dan lokasi.

3. Informasi Layanan:

Pada bagian "Wash & Clean", terdapat informasi tentang layanan yang ditawarkan, termasuk nama pengguna, berat total, harga, tanggal layanan, jenis layanan, dan jumlah. Terdapat juga diagram yang menampilkan status pesanan, mulai dari P01 hingga P08, dengan warna-warna yang berbeda untuk menunjukkan status masing-masing.

Kemudian terdapat pada Gambar 3.7 diagram *wiring* dari alat yang dapat dilihat sebagai berikut.



Gambar 3.7 Diagram *Wiring*



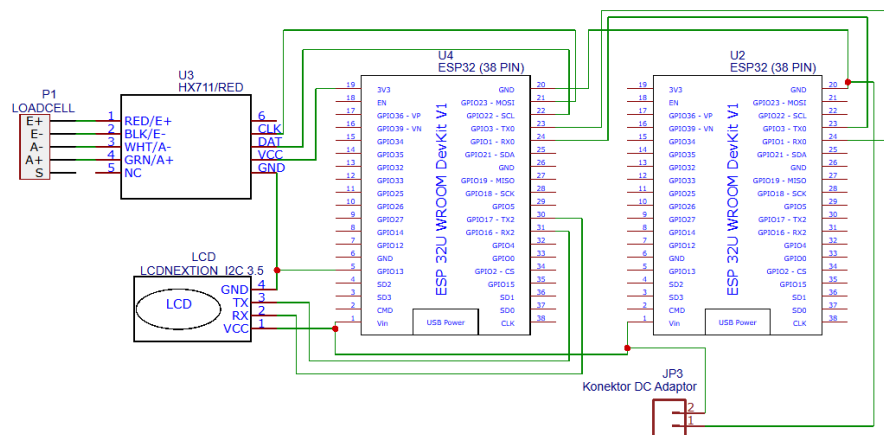
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Berdasarkan diagram di atas menunjukkan bagaimana semua komponen terhubung satu sama lain. LCD *Nextion* 3.5 Inch terhubung ke kontroler ESP32, sementara load cell terhubung ke *amplifier* HX711, yang kemudian *output*-nya dihubungkan ke kontroler ESP32. Kontroler ESP32 juga terhubung ke sumber daya melalui *connector* DC Adaptor.

Sistem yang digambarkan dalam diagram ini memiliki alur kerja yang terintegrasi. Proses dimulai dari *load cell* yang berfungsi untuk mengukur berat atau massa suatu objek. Sinyal dari load cell kemudian diolah oleh *amplifier* HX711 agar dapat dibaca dengan baik oleh kontroler. Kontroler utama yang digunakan adalah dua modul ESP32, yang bertugas memproses data berat yang diterima dari *amplifier*, lalu mengirimkannya ke LCD *Nextion* untuk ditampilkan. LCD *Nextion* 3.5 inci berfungsi sebagai antarmuka visual, menampilkan hasil pengukuran berat yang dilakukan oleh sistem. Seluruh komponen dalam sistem, mulai dari *load cell*, *amplifier*, kontroler, hingga LCD, terhubung dan dioperasikan menggunakan sumber daya listrik yang disalurkan melalui *connector* DC Adaptor. Alur kerja yang terintegrasi ini memungkinkan sistem untuk melakukan pengukuran berat secara efektif dan menampilkan hasilnya kepada pengguna.

Pada alat ini menggunakan *load cell*, *amplifier* HX711, ESP32 ke-1 dan ESP32 ke-2, LCD *Nextion* 3.5, *connector* DC Adaptor. Rancangan skematik pada alat ini dapat dilihat pada Gambar 3.8 di bawah ini.



Gambar 3.8 Skematik Alat



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Berdasarkan gambar di atas, *load cell* merupakan sensor utama yang berfungsi untuk mengukur massa atau berat suatu objek. *Load cell* terhubung ke *amplifier* HX711 melalui kabel yang memiliki 5 koneksi: RED/E+, BLK/E-, WHT/A+, GRN/A+, NC. Dilanjutkan dengan *Amplifier* HX711 berfungsi untuk mengkondisikan sinyal dari *load cell* agar dapat dibaca dengan baik oleh kontroler. *Amplifier* HX711 terhubung dengan pin-pin tertentu pada modul ESP32 KE-1 dan KE-2. Modul ESP32 ke-1 dan ke-2 merupakan dua buah mikrokontroler ESP32 yang berfungsi sebagai kontroler utama dalam sistem.

Mereka menerima data berat dari *amplifier* HX711, memproses, dan mengirimkannya ke LCD *Nextion*. Masing-masing modul ESP32 terhubung dengan pin-pin tertentu untuk komunikasi dan pengoperasian. LCD *Nextion* 3.5 inci merupakan perangkat penampil visual yang digunakan untuk menampilkan hasil pengukuran berat. LCD *Nextion* terhubung dengan pin-pin tertentu pada modul ESP32 ke-1 dan ke-2. Konektor DC Adaptor digunakan untuk menyalurkan sumber daya listrik ke seluruh komponen dalam sistem. Konektor DC Adaptor terhubung dengan pin-pin tertentu pada modul ESP32 ke-1 dan ke-2. Dengan begitu, data berat dari *load cell* akan diproses oleh kontroler ESP32 dan ditampilkan pada LCD *Nextion*, sehingga pengguna dapat melihat hasil pengukuran dengan jelas.

3.3 Pembuatan Program ESP Untuk Pembacaan Sensor

Pemrograman menggunakan *software* Arduino IDE dan program berikut dibuat berdasarkan tugas dan fungsi terkait alat yang dibuat pada penelitian. Program yang dibuat dapat dilihat sebagai berikut :

1. Pemanggilan *Library*

Kode ini digunakan untuk proses awal yaitu mempersiapkan *library*.

```
#include <HX711_ADC.h> //timbangan

#ifdef(ESP32) || defined(ESP32) || defined(AVR)

#include <EEPROM.h> //fungsinya buat penyimpanan

#endif

#include <HardwareSerial.h> //komunikasi serial
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
#include "Adafruit_Thermal.h" //printer
#include "BluetoothSerial.h" //Bluetooth
```

Kode ini menyiapkan beberapa *library* yang akan digunakan dalam pengembangan program alat timbangan *laundry* otomatis berbasis IoT. Berikut penjelasan singkat mengenai fungsi masing-masing *library*:

- a. `HX711_ADC`: *Library* ini digunakan untuk berkomunikasi dengan sensor Loadcell, yang akan mengukur berat pakaian yang diletakkan di atas timbangan.
- b. `EEPROM`: *Library* ini digunakan untuk menyimpan dan membaca data pada memori *non-volatile* EEPROM, yang dapat digunakan untuk menyimpan konfigurasi atau data penting pada alat.
- c. `HardwareSerial`: *Library* ini digunakan untuk komunikasi serial, yang dapat digunakan untuk mengirim dan menerima data antara *board* mikrokontroler dan perangkat lain, seperti komputer atau perangkat IoT lainnya.
- d. `Adafruit_Thermal`: *Library* ini digunakan untuk mengontrol *printer* termal yang terhubung dengan *board* mikrokontroler, sehingga dapat mencetak struk transaksi *laundry*.
- e. `BluetoothSerial`: *Library* ini digunakan untuk komunikasi *Bluetooth*, yang dapat dimanfaatkan untuk menghubungkan alat timbangan dengan perangkat lain, seperti *smartphone* atau *tablet*.

2. Inisialisasi Objek

Kode ini menyiapkan semua yang dibutuhkan untuk mengembangkan sebuah alat timbangan *laundry* otomatis berbasis IoT, termasuk komunikasi *Bluetooth*, *printer* termal, dan antarmuka pengguna dengan *display* *Nextion*. Variabel-variabel yang dideklarasikan akan digunakan untuk menyimpan dan memproses data yang diperlukan oleh aplikasi.

```
BluetoothSerial SerialBT;
Adafruit_Thermal printer(&SerialBT); // Pass addr to
printer constructor
uint8_t address[6] = { 0x66, 0x32, 0x8E, 0x66, 0xD0, 0x4E
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
}; 0-255

String name = "MTP-2";
char *pin = "0000";  //<- standard pin would be provided
                        by default
bool connected;
#define RX_PIN 17
#define TX_PIN 16

HardwareSerial nextionSerial(2);  // Gunakan port serial
                                   kedua pada ESP32
int count = 0;
String data, nama, jumlah, layanan, formattedTime,
        currentDate;
float harga, i; //float tipe data bilangan pecahan
float massa_kg, massa_terkini, massa_sebelumnya;
bool flag, selisih;
bool massa_flag = false;
char receivedChar;
```

Berikut merupakan penjelasan tentang kode diatas:

- a. `BluetoothSerial SerialBT;` Ini adalah deklarasi objek `SerialBT` dari class `BluetoothSerial`. Objek ini akan digunakan untuk komunikasi Bluetooth pada program.
- b. `Adafruit_Thermal printer(&SerialBT);` Ini adalah deklarasi objek `printer` dari class `Adafruit_Thermal`. Objek ini akan digunakan untuk mengirim data ke *printer thermal* melalui komunikasi Bluetooth menggunakan `SerialBT`.
- c. `uint8_t address[6] = { 0x66, 0x32, 0x8E, 0x66, 0xD0, 0x4E };` Ini adalah deklarasi *array address* yang berisi 6 byte alamat Bluetooth perangkat. Dalam hal ini, alamat Bluetooth yang akan digunakan untuk terhubung.
- d. `String name = "MTP-2";` Ini adalah deklarasi variabel `name` yang berisi nama perangkat Bluetooth.
- e. `char *pin = "0000";` Ini adalah deklarasi variabel `pin` yang berisi PIN Bluetooth default.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- f. `bool connected;` Ini adalah deklarasi variabel *connected* yang akan digunakan untuk menyimpan status koneksi Bluetooth.
- g. `#define RX_PIN 17` dan `#define TX_PIN 16;` Ini adalah deklarasi konstanta pin RX dan TX yang akan digunakan untuk komunikasi serial.
- h. `HardwareSerial nextionSerial(2);` Ini adalah deklarasi objek *nextionSerial* dari class *HardwareSerial* yang akan digunakan untuk komunikasi serial dengan *display Nextion*.
- i. `int count = 0;` Ini adalah deklarasi variabel *count* yang akan digunakan untuk menghitung jumlah item.
- j. `String data, nama, jumlah, layanan, formattedTime, currentDate;` Ini adalah deklarasi variabel-variabel tipe *String* yang akan digunakan untuk menyimpan data.
- k. `float harga, i;` Ini adalah deklarasi variabel-variabel tipe *float* yang akan digunakan untuk menyimpan harga dan bilangan pecahan.
- l. `float massa_kg, massa_terkini, massa_sebelumnya;` Ini adalah deklarasi variabel-variabel tipe *float* yang akan digunakan untuk menyimpan data massa.
- m. `bool flag, selisih;` Ini adalah deklarasi variabel-variabel tipe boolean yang akan digunakan untuk menyimpan status.
- n. `bool massa_flag = false;` Ini adalah deklarasi variabel *massa_flag* yang akan digunakan untuk menyimpan status pengukuran massa.
- o. `char receivedChar;` Ini adalah deklarasi variabel *receivedChar* tipe *char* yang akan digunakan untuk menyimpan karakter yang diterima melalui komunikasi serial.

3. Konfigurasi dan Inisialisasi dari Modul Sensor

Kode berikut merupakan proses rangkaian inisialisasi sensor yang dapat dilihat pada di bawah ini:

```
const int HX711_dout = 22; //mcu > HX711 dout pin
const int HX711_sck = 23; //mcu > HX711 sck pin
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
//HX711 constructor:
```

```
HX711_ADC LoadCell(HX711_dout, HX711_sck);
```

```
const int calVal_calVal_eepromAddress = 0;
```

```
unsigned long t = 0; //tipe data panjang tanpa nilai minus
```

```
unsigned long waktu_terkini = 0;
```

Kode yang berikan menunjukkan proses konfigurasi dan inisialisasi dari modul sensor berat HX711 yang akan digunakan dalam proyek ini. Berikut penjelasannya:

a. `const int HX711_dout = 22;` dan `const int HX711_sck = 23;;`

Ini adalah deklarasi konstanta untuk pin digital yang terhubung ke modul HX711. HX711_dout adalah pin sinyal data dari HX711, dan HX711_sck adalah pin sinyal clock.

b. `HX711_ADC LoadCell(HX711_dout, HX711_sck);` Ini adalah inisialisasi objek *LoadCell* dari kelas HX711_ADC. Objek ini akan digunakan untuk berkomunikasi dengan modul HX711 menggunakan pin-pin yang sudah ditentukan sebelumnya.

c. `const int calVal_calVal_eepromAddress = 0;;` Ini adalah deklarasi konstanta yang menunjukkan alamat EEPROM (*Electrically Erasable Programmable Read-Only Memory*) di mana nilai kalibrasi HX711 akan disimpan.

d. `unsigned long t = 0;;` Ini adalah deklarasi variabel *t* bertipe *unsigned long* yang akan digunakan untuk menyimpan nilai waktu.

e. `unsigned long waktu_terkini = 0;;` Ini adalah deklarasi variabel *waktu_terkini* bertipe *unsigned long* yang akan digunakan untuk menyimpan nilai waktu terkini.

4. Mengambil Data Berat

Kode berikut merupakan proses pengambilan data dan merespon hasil data yang ada.

```
void inisial_loadcell() {
    float calibrationValue;           // calibration value
    calibrationValue = 146245.45;     // uncomment this if you want
    to set this value in the sketch
    #if defined(ESP32) || defined(ESP32)
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
//EEPROM.begin(512); // uncomment this if you use ESP32 and
    want to fetch this value from eeprom
#endif

LoadCell.begin();
//LoadCell.setReverseOutput();
unsigned long stabilizingtime = 2000; // tare preciscion can
    be improved by adding a few seconds of stabilizing time
boolean _tare = true;
LoadCell.start(stabilizingtime, _tare);
if (LoadCell.getTareTimeoutFlag()) {
    delay(10);
} else {
    LoadCell.setCalFactor(calibrationValue); // set
    calibration factor (float)
}
while (!LoadCell.update())
    ;
}

void read_massa() {
    for (int j = 0; j <= 100; j++) {
        static boolean newDataReady = 0;
        const int serialPrintInterval = 100; //increase value to
        slow down serial print activity500
        // check for new data/start next conversion:
        if (LoadCell.update()) newDataReady = true;

        // get smoothed value from the dataset:
        if (newDataReady) {
            if (millis() > t + serialPrintInterval) {
                i = LoadCell.getData();
                massa_kg = i;
                posting("t1.txt=\"", String(massa_kg)); //posting data
                ke lcd
                newDataReady = 0;
                t = millis();

                if (millis() - waktu_terkini >= 1500 && selisih) {
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

    waktu_terkini = millis();
    massa_terkini = massa_kg;
    if (abs(massa_terkini - massa_sebelumnya) <= 0.05 &&
    massa_kg > 0.5) {
        flag = true;
        massa_flag = true;
        count = 0;
        data = "";
        break;
    }
    selisih = false;
  } else if (millis() - waktu_terkini >= 1500 && !selisih)
  {
    waktu_terkini = millis();
    massa_sebelumnya = massa_kg;
    selisih = true;
  }
}
}
}

void posting(String alamat, String value) { //posting data ke
  lcd nextion
  nextionSerial.print(alamat); //t0.txt="\
  nextionSerial.print(value); //Value
  nextionSerial.print("\");
  nextionSerial.write(0xFF); // Akhiri perintah dengan kode
  akhir 0xFF
  nextionSerial.write(0xFF);
  nextionSerial.write(0xFF);
}

void serial_transmit() { //ngirim data dari esp 1 ke esp2
  melalui serial
  Serial.print("*");
  Serial.print("test");
  Serial.print(",");
  Serial.print(nama);

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

Serial.print(",");
Serial.print(jumlah);
Serial.print(",");
Serial.print(massa_kg);
Serial.print(",");
Serial.print(layanan);
Serial.print(",");
Serial.print(harga);
Serial.println("#");
}

void restart() {
  // SerialBT.end();
  count = 0;
  data = "";
  nama = "";
  jumlah = "";
  layanan="";
  delay(1000);
  // esp_restart();
}

void setup() {
  pinMode(2, OUTPUT);
  digitalWrite(2, LOW);
  Serial.begin(115200); //baudrate
  SerialBT.setPin(pin);
  SerialBT.begin("ESP32test", true);
  connected = SerialBT.connect(address);
  if(connected) {
    //Serial.println("Connected Succesfully!");
  } else {
    while(!SerialBT.connected(10000)) {
      //Serial.println("Failed to connect. Make sure remote
      device is available and in range, then restart app.");
    }
  }
}

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
//this would reconnect to the name(will use address, if
  resolved) or address used with connect(name/address).
SerialBT.connect();
printer.begin();
nextionSerial.begin(9600, SERIAL_8N1, RX_PIN, TX_PIN);  //
  Inisialisasi komunikasi serial dengan Nextion LCD
inisial_loadcell();
}

void loop() {
while (!SerialBT.connected(4000)){
  //this would reconnect to the name(will use address, if
    resolved) or address used with connect(name/address).
  SerialBT.connect();
}

while (!massa_flag) {
  read_massa();
}

massa_terkini = 0;
massa_sebelumnya = 0;
delay(100);

while (flag) {
  if (nextionSerial.available() > 0) {      // Periksa apakah
    ada data yang tersedia dari Nextion LCD
    receivedChar = nextionSerial.read();    // Baca karakter
    yang diterima
    if (receivedChar != 0) {
      data += receivedChar;
    }
    if (receivedChar == '#') {
      count = 1;
      data = "";
    } else if (receivedChar == '*') {
      count = 2;
      data = "";
    } else if (receivedChar == '+') {
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
count = 3;
data = "";
} else if (data == "re") {
count = 0;
data = "";
massa_flag = false;
flag = false;
// restart();
}
```

Kode yang dijelaskan merupakan sebuah program Arduino yang digunakan untuk menginisialisasi dan membaca data dari sebuah *loadcell* (sensor berat). Berikut penjelasan untuk setiap fungsi:

a. `inisial_loadcell()`:

Fungsi ini digunakan untuk menginisialisasi dan konfigurasi awal pada *loadcell*. Dilakukan pengaturan nilai kalibrasi *loadcell* (*calibrationValue*), yang bisa diatur secara manual atau diambil dari EEPROM (untuk board ESP32). Inisialisasi objek *LoadCell* dan menjalankan proses stabilisasi selama 2 detik sebelum melakukan pembacaan data. Jika proses stabilisasi berhasil, maka diatur nilai faktor kalibrasi pada objek *LoadCell*.

b. `read_massa()`:

Fungsi ini digunakan untuk membaca data massa dari *loadcell* secara berulang. Dilakukan pembacaan data dan pengambilan nilai yang sudah dihaluskan (*smoothed value*). Nilai massa yang terbaca kemudian dikirimkan ke tampilan LCD *Nextion* menggunakan fungsi `posting()`. Terdapat logika untuk mendeteksi perubahan massa yang stabil (selisih massa kurang dari 0,05 kg) selama 1,5 detik, dan menandai nilai massa tersebut dengan flag `massa_flag`.

c. `posting(String alamat, String value)`:

Fungsi ini digunakan untuk mengirimkan data ke LCD *Nextion*. Data dikirim dengan format *string* yang terdiri dari alamat yang dituju dan nilai yang akan dikirimkan. Akhir perintah ditandai dengan kode 0xFF.

d. `serial_transmit()`:



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Fungsi ini digunakan untuk mengirimkan data melalui komunikasi serial. Data yang dikirim terdiri dari beberapa parameter, seperti nama, jumlah, massa, layanan, dan harga. Data dikirim dengan format: `*test,nama,jumlah,masse_kg,layanan,harga#`.

e. `restart()` :

Fungsi ini digunakan untuk me-reset beberapa variabel yang digunakan, seperti count, data, nama, jumlah, dan layanan. Terdapat juga komentar untuk memanggil fungsi `esp_restart()`, yang mungkin digunakan untuk me-restart perangkat ESP32.

f. `setup()` :

Fungsi ini dijalankan saat program dieksekusi. Dilakukan inisialisasi pin, komunikasi serial, komunikasi Bluetooth, *printer*, dan komunikasi dengan Nextion LCD. Fungsi inisial `loadcell()` juga dipanggil di sini untuk menginisialisasi `loadcell`.

g. `loop()` :

Fungsi ini akan dijalankan secara berulang. Dilakukan pengecekan koneksi Bluetooth, dan jika tidak terhubung, akan dilakukan *reconnect*. Selama `masse_flag` bernilai `false`, fungsi `read_masse()` akan dipanggil untuk membaca data massa. Setelah `masse_flag` menjadi `true`, variabel `masse_terkini` dan `masse_sebelumnya` akan diatur ulang. Dilakukan pembacaan data dari Nextion LCD, dan jika menerima karakter tertentu (re, #, *, +), akan dilakukan penanganan sesuai dengan logika program.

5. Perhitungan

Berikut ini merupakan kode untuk menghitung berat yang diterima oleh `loadcell`.

```
if (count == 0) {
    nama = data;
    posting("t0.txt=\"", nama);
} else if (count == 1 && data != "") {
    jumlah = data;
    posting("t2.txt=\"", jumlah);
} else if (count == 2 && data != "") {
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

layanan = data;

/*=====If Nasted from Layanan=====
untuk penentun harga berdasarkan layanan yang dipilih oleh
User*/

if (layanan == "bi" && data != "") { //jika user memilih
layanan biasa masa akan masuk ke perhitungan ini
    harga = 7000 * massa_kg;
    layanan = "Biasa";
    serial_transmit();
    printing(); // proses print
    printing(); //dipanggil 2 kali agar mencetak struk
    sebanyak 2 buah
    massa_flag = false;
    flag = false;
    restart();
} else if (layanan == "se" && data != "") { // jika
user memilih layanan sedang maka akan masuk ke perhitungan
harga ini
    harga = 9000 * massa_kg;
    layanan = "Sedang";
    serial_transmit();
    printing(); // proses print
    printing(); //dipanggil 2 kali agar mencetak struk
    sebanyak 2 buah
    massa_flag = false;
    flag = false;
    restart();
} else if (layanan == "ce" && data != "") { // jika
user memilih layanan cepat maka akan masuk ke proses ini
    harga = 10000 * massa_kg;
    layanan = "Cepat";
    serial_transmit();
    printing(); // proses print
    printing(); //dipanggil 2 kali agar mencetak struk
    sebanyak 2 buah
    massa_flag = false;
    flag = false;
    restart();

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

    }
    }
    }
    delay(10);
  }
}

void printing() { // fungsi void untuk membuat format print
  di struck
  printer.underlineOn();
  printer.println(F("==JUANS LAUNDRY=="));
  printer.underlineOff();
  printer.print(F("Tanggal : "));
  printer.println(tanggal);
  printer.print(F("Nama : "));
  printer.println(nama);
  printer.print(F("Harga : Rp."));
  printer.println(harga);
  printer.print(F("Jumlah : "));
  printer.print(jumlah);
  printer.println(F(" PCS"));
  printer.print(F("Berat : "));
  printer.print(massa_kg);
  printer.println(F(" KG"));
  printer.print(F("Layanan : "));
  printer.print(layanan);
  printer.feed(1);
  printer.println("Seller: +6285776254687(Juan)");
  printer.feed(1);
  printer.println("NOTE: Simpan Struk, Karena setiap 10X
    Laundry, Gratis 1X Laundry");
  printer.feed(3);
  printer.setDefault();
  delay(3000);
  nextionSerial.begin(9600, SERIAL_8N1, RX_PIN, TX_PIN);
  delay(1000);
}

```

Berdasarkan kode yang diberikan, ini adalah penjelasan proses yang terjadi:

a. Proses Utama:

Kode tersebut berada dalam sebuah *loop*. Variabel *count* digunakan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

untuk melacak tahap pemrosesan data. Setiap *iterasi loop*, kode akan memeriksa nilai *count* dan melakukan tindakan berdasarkan kondisinya.

b. Kondisi Pertama: `count == 0`

Jika *count* sama dengan 0, maka nilai data disimpan dalam variabel *nama*. Fungsi posting dipanggil dengan argumen "t0.txt=" dan variabel *nama*.

c. Kondisi Kedua: `count == 1 && data != ""`

Jika *count* sama dengan 1 dan *data* tidak kosong, maka nilai data disimpan dalam variabel *jumlah*. Fungsi posting dipanggil dengan argumen "t2.txt=" dan variabel *jumlah*.

d. Kondisi Ketiga: `count == 2 && data != ""`

Jika *count* sama dengan 2 dan *data* tidak kosong, maka nilai data disimpan dalam variabel *layanan*. Selanjutnya, terdapat blok if-else yang mengecek nilai *layanan* dan melakukan tindakan berdasarkan kondisi tersebut.

e. Kondisi Keempat : `if (layanan == "bi" && data != "")`

Jika *layanan* bernilai "bi" dan *data* tidak kosong, maka harga akan dihitung sebagai $7000 * \text{massa_kg}$. Nilai *layanan* akan diubah menjadi "Biasa". Fungsi `serial_transmit()` akan dipanggil. Fungsi `printing()` akan dipanggil dua kali untuk mencetak struk sebanyak dua buah. Variabel *massa_flag* dan *flag* akan diatur menjadi false. Fungsi `restart()` akan dipanggil.

f. Kondisi Kelima : `else if (layanan == "se" && data != "")`

Jika *layanan* bernilai "se" dan *data* tidak kosong, maka harga akan dihitung sebagai $9000 * \text{massa_kg}$. Nilai *layanan* akan diubah menjadi "Sedang". Fungsi `serial_transmit()` akan dipanggil. Fungsi `printing()` akan dipanggil dua kali untuk mencetak struk sebanyak dua buah. Variabel *massa_flag* dan *flag* akan diatur menjadi false. Fungsi `restart()` akan dipanggil.

g. Kondisi Keenam : `else if (layanan == "ce" && data != "")`

Jika *layanan* bernilai "ce" dan *data* tidak kosong, maka harga akan dihitung sebagai $10000 * \text{massa_kg}$. Nilai *layanan* akan diubah menjadi



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

"Cepat". Fungsi `serial_transmit()` akan dipanggil. Fungsi `printing()` akan dipanggil dua kali untuk mencetak struk sebanyak dua buah. Variabel `massa_flag` dan `flag` akan diatur menjadi `false`. Fungsi `restart()` akan dipanggil.

h. Blok Kondisi Untuk Penentuan Harga Berdasarkan Layanan:

Terdapat beberapa kondisi yang perlu ditangani berdasarkan nilai layanan. Jika layanan sama dengan "bi" dan data `massa_kg` tidak kosong, maka harga dihitung sebagai $7000 * \text{massa_kg}$, layanan diubah menjadi "Biasa", fungsi `printing()` dan `serial_transmit()` dipanggil, beberapa flag diatur ke `false`, dan fungsi `restart()` dipanggil. Jika layanan sama dengan "se" dan data `massa_kg` tidak kosong, maka harga dihitung sebagai $9000 * \text{massa_kg}$, layanan diubah menjadi "Sedang", fungsi `printing()` dan `serial_transmit()` dipanggil, beberapa flag diatur ke `false`, dan fungsi `restart()` dipanggil. Jika layanan sama dengan "ce" dan data `massa_kg` tidak kosong, maka harga dihitung sebagai $10000 * \text{massa_kg}$, layanan diubah menjadi "Cepat", fungsi `printing()` dan `serial_transmit()` dipanggil, beberapa flag diatur ke `false`, dan fungsi `restart()` dipanggil.

i. Fungsi `printing()`:

Fungsi ini bertanggung jawab untuk mencetak informasi transaksi pada *printer*. Informasi yang dicetak meliputi: judul "JUANS LAUNDRY", nama, harga, jumlah, berat, layanan, dan kontak penjual. Setelah mencetak, fungsi `nextionSerial.begin` dipanggil untuk mengatur ulang serial *communication*.

3.4 Pembuatan Program ESP Untuk Mengirimkan Database ke MySQL

Pada penelitian ini membuat sebuah program menggunakan platform ESP (*Espressif Systems*) yang akan mengirimkan data dari sebuah database lokal ke sebuah database MySQL yang terletak di *server* jauh. Tujuan dari proyek ini adalah untuk membangun sebuah sistem yang dapat mengumpulkan data secara otomatis dan menyimpannya ke dalam *database* online, sehingga data

tersebut dapat diakses dan dianalisis dari mana saja. Berikut program yang dibuat dapat dilihat di bawah ini:

1. Inisialisasi ESP

Berikut merupakan bagian awal dari program yang akan menggunakan mikrokontroler ESP32 untuk melakukan koneksi WiFi, komunikasi HTTP, dan mungkin juga komunikasi I2C dengan perangkat lain.

```
#include <Arduino.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include <Wire.h>
#include <DNSServer.h>
#include <WiFiManager.h>
```

Berikut penjelasan dari masing-masing baris kode:

- `#include <Arduino.h>`: Ini merupakan header file utama dari platform Arduino yang digunakan untuk mengakses fungsi-fungsi dasar Arduino.
- `#include <WiFi.h>`: Header file ini digunakan untuk mengakses fungsionalitas WiFi pada mikrokontroler ESP32.
- `#include <HTTPClient.h>`: Header file ini digunakan untuk mengakses fungsionalitas HTTP client pada ESP32, yang memungkinkan program untuk melakukan komunikasi HTTP dengan server web.
- `#include <Wire.h>`: Header file ini digunakan untuk mengakses fungsionalitas I2C pada mikrokontroler ESP32, yang memungkinkan program untuk berkomunikasi dengan perangkat I2C lainnya.
- `#include <DNSServer.h>`: Header file ini digunakan untuk mengakses fungsionalitas DNS server pada ESP32, yang memungkinkan program untuk menyediakan layanan DNS lokal.
- `#include <WiFiManager.h>`: Header file ini digunakan untuk mengakses fungsionalitas WiFi Manager pada ESP32, yang memungkinkan program untuk melakukan konfigurasi WiFi secara otomatis.

2. Pengiriman Data ke MySQL

Setelah berhasil menghubungkan mikrokontroler ESP32 dengan jaringan WiFi, program selanjutnya akan melakukan pengiriman data ke *server database* MySQL. Untuk dapat melakukan hal ini, program perlu mengakses *library*





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

tambahan yang dapat berkomunikasi dengan MySQL.

```
#define MAX_ARRAY_SIZE 6 // Ukuran maksimum array

String dt[MAX_ARRAY_SIZE]; // Deklarasi array data
String dataIn;

const char* serverName =
    "http://espdatajuan29.000webhostapp.com/index.php";
String apiKeyValue = "tPmAT5Ab3j7F9"; // API Key
String nama, harga, jumlah, berat, layanan;

bool parsing = false;
bool httpRequestDone = false;
```

Kode ini menyiapkan beberapa variabel dan konstanta yang akan digunakan dalam program ESP untuk mengirimkan data ke server *http://espdatajuan29.000webhostapp.com/index.php*. Variabel-variabel ini akan digunakan untuk menyimpan data yang diterima dari server, serta informasi lain seperti API key dan status proses pengiriman data. Berikut penjelasan detailnya :

- a. `#define MAX_ARRAY_SIZE 6`: Ini adalah definisi konstanta yang menetapkan ukuran maksimum array `dt` menjadi 6.
- b. `String dt[MAX_ARRAY_SIZE];`: Ini adalah deklarasi array `dt` yang dapat menampung hingga 6 elemen bertipe `String`.
- c. `String dataIn;`: Ini adalah deklarasi variabel `dataIn` bertipe `String` yang akan digunakan untuk menyimpan data yang diterima dari server.
- d. `const char* serverName = "http://espdatajuan29.000webhostapp.com/index.php";`: Ini adalah deklarasi konstanta `serverName` yang berisi URL endpoint server tempat data akan dikirimkan.
- e. `String apiKeyValue = "tPmAT5Ab3j7F9";`: Ini adalah deklarasi konstanta `apiKeyValue` yang berisi nilai API key yang akan digunakan untuk autentikasi ke server.
- f. `String nama, harga, jumlah, berat, layanan;`: Ini adalah deklarasi beberapa variabel `String` yang akan digunakan untuk menyimpan data yang diterima dari server.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

g. `bool parsing = false;;` Ini adalah deklarasi variabel parsing bertipe `bool` yang digunakan untuk melacak apakah proses parsing data sedang berlangsung.

h. `bool httpRequestDone = false;;` Ini adalah deklarasi variabel `httpRequestDone` bertipe `bool` yang digunakan untuk melacak apakah permintaan HTTP sudah selesai.

3. Pemrosesan Data

Setelah data berhasil dikirimkan dan disimpan di server database MySQL, langkah selanjutnya adalah memproses data tersebut untuk mendapatkan informasi yang berguna. Proses ini dapat dilakukan di sisi server atau di sisi aplikasi client yang terhubung dengan database.

```
void parsingData() {
    int j = 0;

    for (int i = 0; i < MAX_ARRAY_SIZE; i++) {
        dt[i] = ""; // Reset semua elemen array data menjadi string
                    // kosong
    }

    for (int i = 0; i < dataIn.length(); i++) {
        if (dataIn[i] == '#') {
            break; // Hentikan pembacaan serial jika karakter '#'
                  // ditemukan
        } else if (dataIn[i] == ',') {
            j++; // Tambahkan indeks jika karakter ',' ditemukan
        } else {
            dt[j] += dataIn[i]; // Tambahkan karakter ke data
        }
    }

    // Set variabel data
    nama = dt[1];
    jumlah = dt[2];
    berat = dt[3];
    layanan = dt[4];
    harga = dt[5];
}
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
void sendHTTPData() {
    HTTPClient http;

    // Your Domain name with URL path or IP address with path
    http.begin(serverName);

    // Specify content-type header
    http.addHeader("Content-Type", "application/x-www-form-
        urlencoded");

    // Prepare your HTTP POST request data
    String httpRequestData = "api_key=" + apiKeyValue + "&nama=" +
        nama +
        "&harga=" + harga + "&jumlah=" + jumlah
        +
        "&berat=" + berat + "&layanan=" +
        layanan;
    Serial.print("httpRequestData: ");
    Serial.println(httpRequestData);

    // Send HTTP POST request
    int httpResponseCode = http.POST(httpRequestData);

    // Handle response
    if (httpResponseCode > 0) {
        Serial.print("HTTP Response code: ");
        Serial.println(httpResponseCode);
    } else {
        Serial.print("Error code: ");
        Serial.println(httpResponseCode);
    }

    // Free resources
    http.end();

    // Set flag bahwa pengiriman HTTP telah selesai
    httpRequestDone = true;
}
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
void connect_wifi() {
    WiFiManager wifiManager;
    if (!wifiManager.autoConnect("Laundry Juan")) {
        Serial.println("Failed to connect and hit timeout");
        delay(3000);
        esp_restart();
        delay(5000);
    }
}

void setup() {
    Serial.begin(115200);
    connect_wifi();
}

void loop() {
    if (!parsing) {
        // Baca data serial jika tidak dalam proses parsing
        if (Serial.available() > 0) {
            char inChar = Serial.read();
            // Cek apakah karakter pertama adalah '*'
            if (inChar == '*') {
                // Mulai menerima data serial jika karakter pertama adalah '*'
                dataIn = ""; // Reset dataIn
                dataIn += inChar;
                parsing = true;
            }
        }
    } else {
        // Terus menerima data serial sampai karakter '#' diterima
        while (Serial.available() > 0) {
            char inChar = Serial.read();
            if (inChar == '#') {
                // Selesai menerima data serial ketika karakter '#'
                diterima
                parsing = false;
                break;
            }
        }
    }
}
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

    }
    // Tambahkan karakter ke dataIn
    dataIn += inChar;
  }

  // Proses parsing data setelah selesai menerima data serial
  if (!parsing) {
    // Cek apakah jumlah karakter yang diterima sesuai dengan
    yang diharapkan (6 karakter)
    if (dataIn.length() < 50) {
      // Proses dataIn sesuai kebutuhan
      parsingData();
      // Kirim data HTTP
      sendHTTPData();
    } else {
      Serial.println("Data serial tidak valid");
    }
  }
}

// Tambahkan keterlambatan untuk menghindari pembacaan serial
yang terlalu cepat
delay(100);
}

```

Kode yang Anda berikan merupakan bagian dari program ESP (*Embedded Systems Programming*) yang berfungsi untuk membaca data serial, memproses data tersebut, dan mengirimkannya ke *server* melalui HTTP *request*. Berikut penjelasan lebih rinci:

a. `parsingData()`:

Fungsi ini bertanggung jawab untuk memproses data yang diterima melalui serial. Pertama-tama, *loop* digunakan untuk mengosongkan semua elemen array `dt` menjadi string kosong. Kemudian, *loop* lain digunakan untuk membaca setiap karakter dari `dataIn` string. Jika karakter adalah '#', *loop* dihentikan. Jika karakter adalah ',', indeks `j` ditambahkan. Jika tidak, karakter ditambahkan ke elemen array `dt[j]`. Setelah proses parsing, nilai-nilai dari `dt` array disimpan ke dalam variabel-variabel yang telah dideklarasikan sebelumnya (nama, jumlah, berat, layanan, harga).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

b. sendHTTPData():

Fungsi ini bertanggung jawab untuk mengirimkan data ke server melalui HTTP POST request. Pertama-tama, sebuah objek HTTPClient dibuat dan diarahkan ke URL *server* yang telah dideklarasikan. Kemudian, header konten-tipe ditambahkan dengan nilai "application/x-www-form-urlencoded". Data HTTP *request* dibuat dengan menggabungkan nilai API key, nama, harga, jumlah, berat, dan layanan. HTTP POST *request* dikirimkan ke *server*, dan kode respons HTTP ditangani. Setelah pengiriman selesai, flag `httpRequestDone` diatur menjadi true.

c. connect_wifi():

Fungsi ini bertanggung jawab untuk menyambungkan perangkat ESP ke jaringan WiFi. Menggunakan WiFiManager, perangkat akan mencoba menyambung ke jaringan WiFi "Laundry Juan". Jika gagal, perangkat akan dihidupkan ulang setelah 3 detik.

d. setup():

Fungsi ini dijalankan saat program dimulai. Serial dimulai dengan kecepatan 115200 baud. `connect_wifi()` dipanggil untuk menyambungkan perangkat ke jaringan WiFi.

e. loop():

Fungsi ini dijalankan secara berulang. Jika parsing flag bernilai false, program akan memeriksa apakah ada data serial yang tersedia. Jika karakter pertama adalah '*', `dataIn` akan diatur dengan karakter tersebut, dan parsing flag diatur menjadi true. Jika parsing flag bernilai true, program akan terus menerima data serial sampai karakter '#' diterima. Setelah selesai menerima data serial, `parsingData()` akan dipanggil untuk memproses data, dan `sendHTTPData()` akan dipanggil untuk mengirimkan data ke server. Terdapat juga penundaan 100 milidetik untuk menghindari pembacaan serial yang terlalu cepat.

3.5 Pembuatan Program untuk Website

A. Application Programing Interface (API)

```
<?php
// Set header untuk menunjukkan bahwa respons adalah JSON
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
header('Content-Type: application/json');

// Koneksi ke database (ganti dengan koneksi ke database
    Anda)

$servername = "localhost";
$dbname = "id22160939_espdatajuan29";
$username = "id22160939_espdatajuan29";
$password = "Ghozi123#";

$conn = new mysqli($servername, $username, $password,
    $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Jika metode yang digunakan adalah DELETE, hapus data
    dari tabel SensorData
if ($_SERVER['REQUEST_METHOD'] === 'DELETE') {
    parse_str(file_get_contents("php://input"),
        $delete_vars);
    $id_to_delete = $delete_vars['id']; // asumsikan bahwa
        'id' adalah parameter yang dikirim untuk menentukan data
        yang akan dihapus

    // Query DELETE untuk menghapus data dari tabel
    $sql_delete = "DELETE FROM SensorData WHERE id =
        $id_to_delete";

    if ($conn->query($sql_delete) === TRUE) {
        $response = array('message' => 'Data deleted
            successfully');
        echo json_encode($response);
    } else {
        $response = array('error' => 'Error deleting data:
            ' . $conn->error);
        echo json_encode($response);
    }
} else { // Jika bukan metode DELETE, maka lakukan query
    untuk mengambil data dari tabel SensorData
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
// Query untuk mengambil data dari database (ganti
dengan query sesuai kebutuhan Anda)
$sql = "SELECT * FROM SensorData";
$result = $conn->query($sql);
```

```
// Buat array untuk menyimpan data dari database
$data = array();
if ($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
        $data[] = $row;
    }
}
```

Program diatas merupakan *script* yang digunakan untuk berinteraksi dengan database dan mengelola data dari tabel "SensorData". *Script* ini menggunakan bahasa pemrograman PHP. Berikut penjelasan rinci mengenai proses kode tersebut:

1. Set Header untuk JSON Response

Baris pertama mengatur header HTTP untuk menunjukkan bahwa respons yang akan dikirimkan adalah dalam format JSON.

2. Koneksi ke Database

Script ini melakukan koneksi ke database menggunakan library mysqli. Koneksi database diatur dengan menggunakan konstanta \$servername, \$dbname, \$username, dan \$password. Jika koneksi gagal, maka akan menampilkan pesan error.

3. Penanganan Permintaan DELETE

Jika metode permintaan yang digunakan adalah DELETE, *script* akan memproses penghapusan data. Pertama-tama, *script* akan mengekstrak nilai id dari data yang dikirimkan melalui permintaan DELETE. Kemudian, *script* akan menjalankan query SQL DELETE untuk menghapus data dari tabel "SensorData" berdasarkan id yang diterima. Jika penghapusan berhasil, *script* akan mengembalikan respons JSON dengan pesan "Data deleted successfully". Jika terjadi kesalahan, skrip akan mengembalikan respons JSON dengan pesan error.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4. Pengambilan Data dari Tabel "SensorData"

Jika permintaan bukan DELETE, skrip akan mengambil data dari tabel "SensorData" menggunakan query SQL `SELECT * FROM SensorData`. Hasil query akan disimpan dalam variabel `$result`.

5. Menyiapkan Data untuk Respons JSON

Script kemudian akan membuat array `$data` untuk menyimpan data yang diambil dari tabel "SensorData". Jika ada data, skrip akan mengiterasi melalui hasil query dan menambahkan setiap baris data ke dalam array `$data`.

6. Menampilkan Data sebagai JSON

Akhirnya, *script* akan menampilkan data dalam array `$data` sebagai respons JSON menggunakan fungsi `json_encode()`.

7. Menutup Koneksi Database

Setelah semua proses selesai, *script* akan menutup koneksi database.

B. Index

```
<?php
$servername = "localhost";
// REPLACE with your Database name
$dbname = "id22160939_espdatajuan29";
// REPLACE with Database user
$username = "id22160939_espdatajuan29";
// REPLACE with Database user password
$password = "Ghozi123#";

// Keep this API Key value to be compatible with the ESP32
// code provided in the project page.
// If you change this value, the ESP32 sketch needs to
// match
$api_key_value = "tPmAT5Ab3j7F9";

$api_key= $nama = $harga = $jumlah= $berat = $layanan =
"";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

$api_key = test_input($_POST["api_key"]);
if($api_key == $api_key_value) {
    $nama = test_input($_POST["nama"]);
    $harga = test_input($_POST["harga"]);
    $jumlah = test_input($_POST["jumlah"]);
    $berat = test_input($_POST["berat"]);
    $layanan = test_input($_POST["layanan"]);

    // Create connection
    $conn = new mysqli($servername, $username,
    $password, $dbname);
    // Check connection
    if ($conn->connect_error) {
        die("Connection failed: " . $conn-
    >connect_error);
    }

    $sql = "INSERT INTO SensorData (nama, harga,
    jumlah, berat, layanan)
    VALUES ('" . $nama . "', '" . $harga . "', '" .
    $jumlah . "', '" . $berat . "', '" . $layanan . "')";

    if ($conn->query($sql) === TRUE) {
        echo "New record created successfully";
    }
    else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }

    $conn->close();
}
else {
    echo "Wrong API Key provided.";
}

}
else {
    echo "No data posted with HTTP POST.";
}

```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```

Program di atas digunakan untuk menerima data dari perangkat ESP32 melalui permintaan HTTP POST dan menyimpannya ke dalam database MySQL. Berikut adalah penjelasan proses kode secara rinci:

1. Pengaturan Koneksi Database:

- `$servername = "localhost";` Berfungsi untuk menetapkan nama server database, dalam contoh ini menggunakan "localhost".
- `$dbname = "id22160939_espdatajuan29";` Berfungsi untuk menetapkan nama database yang akan digunakan.
- `$username = "id22160939_espdatajuan29";` Berfungsi untuk menetapkan nama pengguna untuk mengakses database.
- `$password = "Juan123#";` Berfungsi untuk Menetapkan kata sandi untuk mengakses database.
- `$api_key_value = "tPmAT5Ab3j7F9";` Berfungsi untuk Menetapkan kunci API yang akan digunakan untuk memeriksa keabsahan permintaan dari perangkat ESP32.

2. Menerima Data dari Permintaan HTTP POST:

- `if ($_SERVER["REQUEST_METHOD"] == "POST")` Berfungsi untuk Memeriksa apakah permintaan yang diterima adalah permintaan HTTP POST.
- `$api_key = test_input($_POST["api_key"]);` Berfungsi untuk Menerima dan memproses nilai kunci API yang dikirimkan dari permintaan POST.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- `$nama = test_input($_POST["nama"])` Berfungsi untuk Menerima dan memproses nilai "nama" yang dikirimkan dari permintaan POST.
- `$harga = test_input($_POST["harga"]);` Berfungsi untuk Menerima dan memproses nilai "harga" yang dikirimkan dari permintaan POST.
- `$jumlah = test_input($_POST["jumlah"]);` Berfungsi untuk Menerima dan memproses nilai "jumlah" yang dikirimkan dari permintaan POST.
- `$berat = test_input($_POST["berat"]);` Berfungsi untuk Menerima dan memproses nilai "berat" yang dikirimkan dari permintaan POST.
- `$layanan = test_input($_POST["layanan"]);` Berfungsi untuk Menerima dan memproses nilai "layanan" yang dikirimkan dari permintaan POST.

3. Menyimpan Data ke Database:

- `$conn = new mysqli($servername, $username, $password, $dbname);` Berfungsi untuk Membuat koneksi baru ke database.
- `if ($conn->connect_error) { die("Connection failed: " . $conn->connect_error); }` Berfungsi untuk Memeriksa apakah koneksi berhasil, jika tidak maka menampilkan pesan kesalahan.
- `$sql = "INSERT INTO SensorData (nama, harga, jumlah, berat, layanan) VALUES ('" . $nama . "', '" . $harga . "', '" . $jumlah . "', '" . $berat . "', '" . $layanan . "')";` Berfungsi untuk Menyusun query SQL untuk menyimpan data ke tabel "SensorData".
- `if ($conn->query($sql) === TRUE) { echo "New record created successfully"; } else { echo`



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
"Error: " . $sql . "<br>" . $conn->error; }
```

Berfungsi untuk Menjalankan query SQL dan memeriksa apakah data berhasil disimpan atau terjadi kesalahan.

```
$conn->close(); : Menutup koneksi database.
```

4. Pengecekan Kunci API:

```
if($api_key == $api_key_value) { ... }
```

Berfungsi untuk Memeriksa apakah kunci API yang diterima dari permintaan POST cocok dengan kunci API yang ditetapkan. Jika tidak cocok, akan menampilkan pesan "Wrong API Key provided."

5. Fungsi Pembersihan Data:

```
function test_input($data) { $data = trim($data); $data = stripslashes($data); $data = htmlspecialchars($data); return $data; }
```

Fungsi ini digunakan untuk membersihkan dan memvalidasi data yang diterima dari permintaan POST sebelum diproses lebih lanjut.

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB IV PEMBAHASAN

Pada bab ini akan membahas mengenai hasil data Pengujian Rancang Bangun Automasi berbasis IoT pada usaha *laundry* meliputi berat beban yang ditimbang hingga penjadwalan otomatis berdasarkan permintaan dan kebutuhan pelanggan. Alat ini terdiri dari beberapa komponen yaitu terdapat *printer thermal* berwarna biru yang digunakan untuk mencetak struk atau *output*, dan terdapat timbangan besar yang digunakan untuk mengukur berat barang atau objek. Perangkat LCD layar sentuh berfungsi sebagai antarmuka interaktif untuk mengoperasikan sistem tersebut, memungkinkan pengguna memasukkan data atau memberikan perintah, sehingga seluruh sistem berjalan dengan baik.

Pesanan pelanggan/customer akan menjadi data yang diinputkan, data customer akan diolah sistem yang kemudian hasil data tersebut dapat dilihat pada *Web*. Kemudian disandingkan database MySQL, data pada API dan Hasil data pada tampilan *Web* untuk melihat apakah data sudah sesuai sehingga dapat dikatakan bahwa alat dapat berjalan dengan baik.

4.1 Deskripsi Pengujian

Pada pengujian kali ini, timbangan akan mengukur berat dari pakaian kotor tersebut, dan hasil pengukuran akan ditampilkan pada layar LCD *touchscreen*. Selanjutnya, kotak *input* akan muncul pada layar *touchscreen* untuk memasukkan nama pelanggan. Setelah nama pelanggan dimasukkan, opsi untuk memilih layanan laundry Reguler atau Ekspres akan muncul. Pelanggan dapat memilih layanan yang mereka inginkan dengan menekan tombol di layar *touchscreen*, lalu menekan tombol "OK" untuk melanjutkan proses. Setelah pelanggan menekan tombol "OK", informasi tentang berat pakaian, nama pelanggan, dan layanan yang mereka pilih akan diketahui Informasi seperti nama pelanggan, tanggal laundry, berat pakaian kotor, layanan yang dipilih, dan harga tagihan yang harus dibayar oleh pelanggan saat mengambil pakaian yang telah selesai dicuci harus dimasukkan ke dalam struktur transaksi.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2 Prosedur Pengujian

Dengan menjalankan skenario pengujian yang mencakup seluruh alur proses untuk memastikan bahwa sistem layanan laundry berfungsi dengan baik dan memenuhi kebutuhan pelanggan, berikut merupakan prosedur pengujian pada penelitian ini :

1. Pengukuran Berat Pakaian

- Letakkan pakaian kotor di atas timbangan digital.
- Verifikasi bahwa berat pakaian terukur dan ditampilkan dengan benar di layar *touchscreen* LCD.

2. Input Data Pelanggan

- Sentuh area *input* nama di layar *touchscreen*.
- Masukkan nama pelanggan menggunakan keyboard virtual.
- Verifikasi bahwa nama pelanggan tertulis dengan benar di layar.

3. Pemilihan Layanan Laundry

- Sentuh tombol "Reguler" atau "Ekspres" di layar *touchscreen*.
- Verifikasi bahwa layanan yang dipilih terlihat jelas di layar.

4. Konfirmasi Proses

- Sentuh tombol "OK" untuk melanjutkan proses.
- Verifikasi bahwa informasi berat pakaian, nama pelanggan, dan layanan yang dipilih terkirim ke *database*.

5. Pencetakan Struk Transaksi

- Verifikasi bahwa informasi transaksi (nama pelanggan, tanggal, berat, layanan, total biaya) tercetak dengan benar di struk.

4.3 Data Hasil Pengujian

Hasil pngujian utama didapatkan berupa id, nama, berat, layanan serta waktu pemesanan yang terdapat pada *database web* myphpadmin yang diakses melalui xampp. Data-data tersebut dapat dilihat pada Tabel 4.1 sebagai berikut.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Tabel 4.1 Data Customer

Customer				
No	Nama	Berat (Kg)	Layanan	Waktu Pemesanan
1	Zack	1,1	Cepat	2024/05/25 08:31:58
2	Gerry	1,1	Cepat	2024/07/29 20:20:26
3	Juminten	1,1	Cepat	2024/07/29 20:20:46
4	Pahakim	10	Cepat	2024/07/29 20:20:13
5	Mutia	1,1	Cepat	2024/07/29 20:20:01
6	Yusuh	0,78	Sedang	2024/07/29 20:19:36
7	Ghozi	1,54	Biasa	2024/07/21 12:52:45
8	Zhafran	2,14	Sedang	2024/07/29 20:19:20
9	Hardi	2,6	Biasa	2024/07/21 15:28:38

Pada tabel diatas, terdapat 9 pelanggan dengan berat dan layanan yang berbeda-beda, kemudian terdapat data waktu pemesanan yang dimana ini menjadi acuan nantinya untuk bisa mengestimasi pengambilan sesuai dengan layanan yang dipilih.

A. Tampilan Data Hasil pada MySQL

Kemudian data hasil pengolahan sistem dapat dilihat pada Gambar 4.1 sebagai berikut.

		id	nama	harga	jumlah	berat	layanan	reading_time
<input type="checkbox"/>	Edit	406	zack	10994.60	9	1.10	Cepat	2024-05-25 08:31:58
<input type="checkbox"/>	Edit	407	gerry	10994.60	9	1.10	Cepat	2024-07-29 20:20:26
<input type="checkbox"/>	Edit	408	juminten	10994.60	9	1.10	Cepat	2024-07-29 20:20:46
<input type="checkbox"/>	Edit	413	pahakim	10988.51	1.10	10	Cepat	2024-07-29 20:20:13
<input type="checkbox"/>	Edit	415	mutia	10999.04	67	1.10	Cepat	2024-07-29 20:20:01
<input type="checkbox"/>	Edit	419	yusuf	7048.96	7	0.78	Sedang	2024-07-29 20:19:36
<input type="checkbox"/>	Edit	420	ghozi	10796.08	4	1.54	Biasa	2024-07-21 12:52:45
<input type="checkbox"/>	Edit	421	zhafran	19283.37	8	2.14	Sedang	2024-07-29 20:19:20
<input type="checkbox"/>	Edit	422	hardi	18224.64	2	2.60	Biasa	2024-07-21 15:28:38

Gambar 4.1 Database pada tampilan data MySQL



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada Gambar 4.1 di atas merupakan data pada MySQL, data ini digunakan untuk mencatat informasi pesanan pakaian pelanggan yang akan atau sedang dicuci di bisnis *laundry*. Hal ini terlihat dari kolom-kolomnya seperti ID pesanan, nama pelanggan, jenis pakaian, berat, dan status. Dengan mencatat detail setiap pesanan secara terstruktur dalam basis data MySQL, petugas *laundry* dapat dengan gampang memonitor dan mengelola pesanan yang masuk dan sedang diproses. Mereka bisa melihat antrian pekerjaan beserta prioritasnya berdasarkan waktu pesanan. Fitur pencarian dan pengurutan yang dimiliki MySQL juga memudahkan petugas untuk menemukan pesanan tertentu berdasarkan kriteria seperti nama pelanggan atau status. Hal ini sangat membantu pelayanan kepada pelanggan bila terjadi keluhan atau pemeriksaan.

B. Tampilan Data Hasil pada API

```
[
  {
    "id": "406",
    "nama": "zack",
    "harga": "10994.60",
    "jumlah": "0",
    "berat": "1.10",
    "layanan": "Cepat",
    "reading_time": "2024-05-25 08:31:58"
  },
  {
    "id": "407",
    "nama": "gerry",
    "harga": "25000.00",
    "jumlah": "18",
    "berat": "2.50",
    "layanan": "Cepat",
    "reading_time": "2024-07-24 13:42:08"
  },
  {
    "id": "408",
    "nama": "juminten",
    "harga": "33200.00",
    "jumlah": "20",
    "berat": "3.33",
    "layanan": "Cepat",
    "reading_time": "2024-07-24 13:56:04"
  },
  {
    "id": "413",
    "nama": "bahakim",
    "harga": "21300.00",
    "jumlah": "16",
    "berat": "2.13",
    "layanan": "Cepat",
    "reading_time": "2024-07-24 14:53:12"
  },
  {
    "id": "415",
    "nama": "mutia",
    "harga": "16700.00",
    "jumlah": "11",
    "berat": "1.67",
    "layanan": "Cepat",
    "reading_time": "2024-07-24 15:09:01"
  },
  {
    "id": "419",
    "nama": "yusuf",
    "harga": "12960.00",
    "jumlah": "13",
    "berat": "1.44",
    "layanan": "Sedang",
    "reading_time": "2024-07-24 15:19:36"
  },
  {
    "id": "420",
    "nama": "ghozi",
    "harga": "9590.00",
    "jumlah": "16",
    "berat": "1.37",
    "layanan": "Biasa",
    "reading_time": "2024-07-24 14:02:09"
  },
  {
    "id": "421",
    "nama": "zhafran",
    "harga": "19283.37",
    "jumlah": "21",
    "berat": "2.14",
    "layanan": "Sedang",
    "reading_time": "2024-07-24 20:19:20"
  },
  {
    "id": "422",
    "nama": "hardi",
    "harga": "18224.64",
    "jumlah": "24",
    "berat": "2.60",
    "layanan": "Biasa",
    "reading_time": "2024-07-24 17:28:38"
  }
]
```

Gambar 4.2 Data Hasil pada tampilan data API



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada Gambar 4.2 memperlihatkan kolom-kolom data seperti `id_pesanan`, `nama_pelanggan`, `jenis_pakaian` dan `status_pesanan` merupakan parameter wajib yang dibutuhkan API untuk mengelola pesanan cucian dengan terstruktur. Fitur pencarian berdasarkan parameter tertentu akan memudahkan aplikasi menangani *request* dari pengguna seperti menampilkan detail pesanan. Data dalam format JSON akan membuat proses integrasi antara backend sistem dengan aplikasi *mobile/website* menjadi lebih lancar.

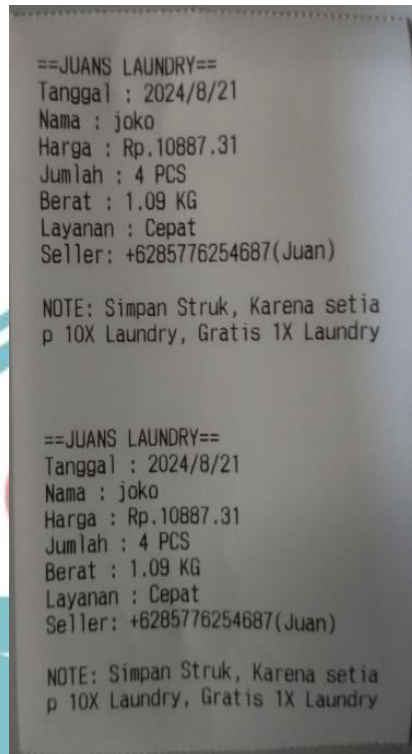
C. Tampilan Data Hasil pada Web

No	ID	Nama	Harga	Jumlah	Berat	Layanan	Delete
1	406	zack	10994.80	9	1.10	Cepat	Delete
2	407	gerry	25000.00	18	2.50	Cepat	Delete
3	408	juminten	33200.00	20	3.32	Cepat	Delete
4	413	pahakim	21300.00	16	2.13	Cepat	Delete
5	415	mutia	16700.00	11	1.67	Cepat	Delete
6	419	yusuf	12960.00	13	1.44	Sedang	Delete
7	420	ghozi	9590.00	16	1.37	Biasa	Delete
8	421	zhafran	19283.37	21	2.14	Sedang	Delete
9	422	hardi	18224.64	24	2.60	Biasa	Delete

Gambar 4.3 Data Hasil pada tampilan Web

Pada Gambar 4.3 tampilan *website* di atas, menu utama yang terdapat di bagian atas sangat mudah diakses pengguna. Hanya dengan satu klik, pengguna sudah bisa mendapatkan informasi yang dibutuhkan. Kemudian secara desain warna *background* dan tulisan yang kontras membuat mata pengguna nyaman saat membaca dan tidak terlalu menyilaukan atau membingungkan. Ukuran huruf dan spasi antar konten didesain proporsional, sesuai standar tampilan *website* yang ergonomis.

D. Tampilan Data Hasil pada Struk



Gambar 4.4 Data Hasil pada tampilan Struk

Pada gambar diatas merupakan struk yang dicetak oleh *printer thermal*, dimana alat akan mencetak 2 struk, yaitu 1 untuk pelanggan dan 1 untuk *seller*. Pada struk terdapat tanggal transaksi pembayaran, nama, harga, jumlah baju, berat yang ditimbang serta layanan yang dipilih. Sehingga struk ini menjadi informasi terkait transaksi agar dapat menjadi bukti ketika terjadi kesalahan perhitungan dalam transaksi. Kemudian terdapat NOTE untuk penginformasian bahwa terdapat promo pada layanan ini dengan menyimpan struk transaksi. Untuk setiap 10x pemesanan mendapatkan gratis 1 x *laundry* dengan berat kurang dari 2 kg dengan syarat menyimpan 10 struk transaksi.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.4 Analisa Data / Evaluasi

Berdasarkan Tabel 4.1 dapat dilihat bahwa terdapat 9 customer yang memesan jasa *laundry* dengan berat dan layanan yang berbeda-beda. Pada usaha ini sistem untuk pembayaran setiap layanan berbeda-beda sesuai dengan layanan yang dipilih oleh *customer*. Layanan cepat akan memproses *laundry* dengan cepat yaitu dalam 1 hari selesai dengan tarif Rp.10.000,-. Kemudian layanan sedang akan diproses dalam 2 hari dengan tariff Rp.9.000,- dan untuk layanan biasa akan diproses dalam 3 hari dengan tarif Rp.7.000,-.

Pada Gambar 4.2 dapat dilihat sistem telah memproses data customer yang memesan jasa *laundry*. Ketika customer melakukan pemesanan maka setiap user akan memiliki *primary key* sehingga dapat menjadi pengaturan referensi untuk proses selanjutnya dan terhindar dari *error data*. Kemudian berat pakaian yang dibaca oleh sensor *loadcell* akan diterima oleh mikrokontroller yang kemudian data tersebut akan di kalikan dengan nominal dari setiap layanan yang digunakan sehingga sistem akan memberikan informasi harga atau tarif dari pesanan user. Kemudian terdapat *reading_time* dimana merupakan waktu pemesanan sehingga data ini akan disesuaikan dengan layanan yang dipilih lalu mikrokontroller akan dapat menentukan penjadwalan pengambilan layanan. Dari semua data yang telah diproses tadi akan di print-out dengan fitur cetak struk. Informasi-informasi yang tercantum dalam struk digital ini dapat berupa nama pelanggan, tanggal transaksi, jumlah barang, jenis layanan, dan total biaya yang harus dibayarkan.

Data yang diterima database disimpan dalam tabel dan dapat dilihat melalui antarmuka phpMyAdmin. Tampilan database menunjukkan data terkirim dari ESP berhasil tersimpan rapi dalam tabel. Selanjutnya data dalam *database* diolah menggunakan bahasa PHP menjadi API berstandar RESTful. API bertujuan memudahkan akses dan penarikan data dari luar melalui permintaan HTTP.

Percobaan akses API menggunakan client REST menunjukkan respons JSON yang sesuai dengan format data asli di database. Hal ini mengindikasikan API dapat mewakili data *database* dengan baik. Terakhir, *website* dibangun untuk menampilkan hasil akhir proyek secara publik. *Website* mengakses data

lewat API dan menampilkannya secara dinamis. Pengujian menunjukkan data yang ditampilkan website sama dengan data asli pada *database*, menandakan integrasi antarkomponen berjalan sinkron.

Hasil data pada struk menunjukkan bahwa struk tersebut merupakan bukti transaksi dimana ini menjadi informasi terkait transaksi agar dapat menjadi bukti ketika terjadi kesalahan perhitungan dalam transaksi untuk mencegah terjadi kesalahan perhitungan tersebut, layanan juga memberikan informasi kontak yang dapat dihubungi ketika terjadinya suatu kesalahan. Perlu dilakukan percobaan secara berkala tiap minggu selama satu bulan untuk menguji performa alat jika digunakan dalam waktu yang lama serta untuk pengembangan selanjutnya dapat memberikan fitur ulasan pada website sehingga pelanggan dapat memberikan apresiasi atau evaluasi untuk layanan usaha *laundry* kedepannya.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

