

**RANCANGBANGUN DAN IMPLEMENTASI SISTEM INFORMASI BANK
SAMPAH BERBASIS INTERNET OF THINGS
DESIGN AND IMPLEMENTATION INFORMATION SYSTEM
WASTE BANK BASED ON INTERNET OF THINGS**

Muhammad Rifki¹, Ir. Agus Ganda Permana, M.T.², Asep Mulyana S.T., M.T.³

¹Prodi D3 Teknologi Telekomunikasi

^{2,3}Prodi D3 Teknologi Telekomunikasi, Fakultas Ilmu Terapan, Universitas Telkom

¹muhrifki@student.telkomuniversity.ac.id, ²agusgandapermana@tass.telkomuniversity.ac.id,

³asepmulyana@tass.telkomuniversity.ac.id

Abstrak

Bank sampah merupakan salah satu solusi untuk mengurangi permasalahan sampah yang ada, karena bank sampah suatu pengelolaan sampah berbasis lingkungan yang dilakukan dengan cara mendaur ulang sampah-sampah menjadi barang yang bernilai ekonomis. Namun pada praktiknya, proses transaksi dan arus data dalam pengelolaan bank sampah terkendala dalam pencatatan masih menggunakan buku dalam pengelolaan nasabah dan manajemen. Hal ini memungkinkan terjadinya kehilangan data dan kurang efektifitas dari merekap jumlah sampah dan dari segi pengepul harus mendatangi Bank Sampah untuk mengetahui rincian jumlah stok sampah yang ingin dibelinya.

Berdasarkan hal tersebut maka di rancanglah suatu sistem informasi Bank Sampah berbasis internet of things dapat membantu permasalahan yang ada untuk saat ini. Perancangan tersebut menggunakan website sebagai penampilannya yang berfungsi untuk memudahkan masyarakat dalam meningkatkan kinerja pada Bank Sampah dalam mengelola data dan menyediakan informasi dengan akurat. Fitur Realtime Database yang telah disediakan Google Firebase akan sangat berguna melihat data secara realtime.

Dalam perancangannya menggunakan sensor RFID yang terhubung dengan Firebase Realtime Database untuk setiap nasabah supaya pada proses menabung meminimalisir kesalahan dalam penginputan data dan sensor loadcell yang terhubung dengan Firebase Realtime Database untuk memonitoring berat sampah. Data – data yang diambil secara realtime dari perangkat mikrokontroler yang kemudian akan ditampilkan di halaman website, website dapat diakses dimana saja untuk mempermudah mobilitas masyarakat.

Hasil pengujian menunjukkan website dapat terintegrasi dengan hardware. Website dapat memonitor berat sampah dan data nasabah saat menabung sampah dengan akurat. Dalam hal performa website dari setiap halaman yang diuji secara fungsionalitas berjalan dengan sesuai harapan. Pengujian delay pada hardware mendapatkan rata - rata delay 2.23 detik dan dari pengujian keakuratan timbangan mendapat 0% kesalahan dapat dinyatakan hardware berjalan dengan sesuai harapan. Menggunakan lighthouse mendapat rata-rata nilai 75, dan menurut parameter dari lighthouse dinyatakan performa website dari setiap halaman yang diuji mendapat nilai normal, dan menggunakan Apache Jmeter mendapat nilai average rata-rata respon diberikan 335 millisecond atau 0.335 second tidak melebihi 1 detik dinyatakan memiliki respon yang bagus.

Kata kunci : Bank Sampah, Sistem Informasi, Internet of Things, Monitoring, Hardware, Website, Google Firebase Realtime Database.

Abstract

The waste bank is one solution to reduce the existing waste problem, because the waste bank is an environmentally based waste management which is carried out by recycling waste into goods of economic value. However, in practice, the transaction process and data flow in waste bank management are constrained by the use of books in customer management and management. This allows data loss and the ineffectiveness of recapitulating the amount of waste.

Based on this, an internet of things-based Waste Bank information system was designed to help current problems. The design uses a website as its appearance which functions to facilitate the public in improving the performance of the Waste Bank in managing data and providing accurate information. The Realtime Database feature that Google Firebase provides will be very useful to see realtime data.

In its design, it uses an RFID sensor that is connected to the Firebase Realtime Database for each customer so that the saving process minimizes errors in data entry and load cell sensors are connected to the Firebase Realtime Database to monitor waste weight. Data taken in real time from the microcontroller device which will then be displayed on the website page, the website can be accessed anywhere to facilitate the mobility of the community.

The test results show that the website can be integrated with hardware. The website can monitor waste weight and customer data when saving waste accurately. In terms of website performance, each page is tested for functionality as expected. The delay test on the hardware gets an average delay of 2.23 seconds and from testing the accuracy of the scales gets 0% error, it can be stated that the hardware is running as expected. Using the lighthouse, it gets an average value of 75, and according to the parameters of the lighthouse it is stated that the website performance of each page tested gets a normal value, and using Apache Jmeter, it gets an average response value of 335 milliseconds or 0.335 seconds, not more than 1 second, which is stated to have a good response. Currently, various IP-based software has emerged that lead to the Next Generation Network. Some of the software includes OpenSIPS, Trixbox, OpenIMS, and Asterisk. Then came a new Kamailio VoIP platform which has reliable performance. A problem arose when the VoIP platform was expanding but still wanted to use the previous VoIP platform. Therefore it is necessary to do inter-server interconnection so that communication can be done.

Keyword : Waste Bank, Information System, Internet of Things, Monitoring, Hardware, Website, Google Firebase Realtime Database.

1. PENDAHULUAN

Pengelolaan sampah saat ini seharusnya sudah sangat didukung oleh perkembangan teknologi yang sudah maju dengan melakukan implementasi otomatisasi pengolahan sampah baik di pusat pengolahan sampah maupun pada daerah pemukiman. Oleh karena itu untuk mengatasi sampah dilakukan dengan berbagai cara, salah satunya dengan bank sampah [1]. Bank sampah sebagai suatu program yang dirancang oleh pemerintah untuk mengurangi volume sampah yang ada, bank sampah selain digunakan untuk menabung dan juga mengajarkan pada masyarakat untuk mengolah sampah yang ada menjadi hal yang berguna serta memiliki nilai jual [2].

Proses di dalam bank sampah dimana masyarakat menyetorkan sampah berupa sampah organik atau non-organik kepada pihak bank sampah sedangkan proses pencatatan masih manual yaitu melakukan pencatatan di kertas sehingga rentan terjadi kehilangan data. Untuk itu perlu dibuat suatu sistem yang dapat memudahkan petugas mencatat transaksi dan nasabah mengecek informasi secara mandiri. Selain itu juga berfungsi untuk meningkatkan kinerja pada Bank Sampah dalam mengelola data dan menyediakan informasi dengan cepat dan akurat [3].

Pada proyek akhir ini akan dilakukan pembuatan sistem informasi yang memudahkan proses pencatatan dan pemantauan arus data transaksi bank sampah, jadi dalam perancangannya sistem bank sampah yang dapat melakukan identifikasi masyarakat yang menabung secara otomatis menggunakan sensor RFID dan *Loadcell* yang terhubung dengan *Firebase Realtime Database* dan berat sampah yang ditimbang dapat termonitoring langsung di *website*, jadi nasabah dari bank sampah ini dapat memantau proses menabung sampah dari awal hingga akhir, dari sisi pengepul dapat melihat persediaan barang yang ingin diambil secara mudah melalui *website* dapat diakses dimana saja dan mempermudah aktivitas tanpa perlu mendatangi lokasi, dari sisi petugas pusat dapat mengetahui rincian transaksi serta stok yang ada untuk memberikan fasilitas terbaik untuk nasabah dan target dalam perancangan bank sampah ini dapat memberikan solusi terkait masalah pengelolaan sampah.

2. DASAR TEORI

2.1 Internet of Things (IoT)

Internet of Things dapat didefinisikan sebagai sebuah konsep yang mampu mempermudah aktivitas [4]. IoT juga dapat diartikan sebagai suatu konsep dimana beberapa objek tertentu mempunyai kemampuan untuk mentransfer data melalui jaringan internet tanpa memerlukan adanya interaksi dari manusia ke manusia ataupun dari perangkat komputer ke manusia.

2.2 Bank Sampah

Bank Sampah berbasis *internet of things* adalah sebuah metode hasil karya anak bangsa sebagai solusi terbaik dalam hal penanganan dan pengelolaan permasalahan sampah di Indonesia berbasis energi terbarukan dan aplikasi teknologi ICT.

Bank Sampah dapat membantu dalam masalah sampah yang dapat membedakan jenis-jenis sampah berdasarkan nilai jualnya, seperti sampah organik yang berupa kulit buah, sayuran, dan sampah non organik yang berupa plastik air kemasan atau gelas yang dapat dijual kembali. Dengan adanya Bank Sampah ini warga setempat dapat menabung dan juga dapat dijual untuk dibuat sebuah kerajinan oleh pengrajin dan dapat menjadi barang yang tepat guna.

2.3 Website

Website adalah sebuah halaman depan yang terdapat pada suatu *domain* di internet yang dibuat dengan tujuan tertentu dan dapat saling berhubungan satu sama lain dan dapat diakses dengan menggunakan sebuah peramban dengan cara menuliskan alamat *website*. Web terbangun dari kumpulan bahasa pemrograman atau *script* yang dibuat *programmer* sehingga dapat terbentuk sebuah web. Halaman-halaman web pada jaringan internet dikomunikasikan melalui protokol *Hypertext Transfer Protocol* (HTTP) [5]. Secara umum web dapat digolongkan menjadi 3, antara lain web statis dimaksudkan jika informasi pada web tidak berubah-ubah seperti *website* profil

perusahaan, sedangkan web dinamis apabila informasi pada web sering berubah-ubah biasanya web seperti ini dipakai untuk blog, situs belanja *online* [6]. Kemudian untuk web interaktif dimana web tersebut bersifat dua arah sehingga *user* dan pemilik web dapat saling berinteraksi seperti *facebook*, *twitter* dan *instagram*.

2.4 Hypertext Markup Language (HTML)

HTML bisa disebut bahasa yang digunakan untuk menampilkan dan mengelola hypertext. HTML digunakan untuk menampilkan berbagai informasi di dalam sebuah penjelajah web internet [7]. HTML mempunyai dua macam ekstensi yaitu *.htm* dan *.html*, format ekstensi *.htm* awalnya hanya untuk mengakomodasi penggunaan *html* dalam operasi Disk Operating System (DOS). HTML saat ini telah mencapai versi 5, dimana beberapa fitur unggulan telah dimasukkan ke dalam pemrograman tersebut. Misal audio, video, dan lain-lain. OpenSIPS adalah sebuah perangkat lunak server yang bersifat open source yang digunakan sebagai SIP server. OpenSIPS bukan hanya sekadar proxy server untuk menangani layanan VoIP, melainkan memiliki fungsionalitas penting seperti instant messaging server, SIP load balancer, SIP IP gateway, SIP media controller, dan masih banyak lagi.

2.5 Hypertext Preprocessor (PHP)

PHP adalah sebuah bahasa pemrograman *open source* yang digunakan untuk membuat dan mengembangkan situs web. PHP merupakan suatu bahasa pemrograman yang dijalankan/diproses di sisi server atau istilah lainnya *server-Side programming* [8], jadi PHP membutuhkan web server untuk menjalankannya. Bahasa pemrograman ini dapat digunakan bersamaan dengan bahasa pemrograman HTML, dimana HTML digunakan sebagai pembangun atau pondasi dari kerangka *layout* web dan PHP difungsikan sebagai prosesnya.

2.6 Cascading Style Sheet (CSS)

CSS merupakan suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi dan tertata. CSS saat ini dikembangkan oleh World Wide Web Consortium (W3C) dan menjadi bahasa standar dalam pembuatan web, CSS pemrograman yang wajib dikuasai oleh setiap *programmer* [8]. CSS difungsikan sebagai pendukung dan pelengkap dari *file html* yang berperan dalam penataan atau mendesain *layout*.

2.7 Javascript

JavaScript merupakan suatu bahasa pemrograman yang dibuat supaya *website* menjadi lebih dinamis. JavaScript adalah bahasa pemrograman yang dapat dikolaborasikan dengan file HTML [5]. Meskipun memiliki nama yang hampir serupa dengan pemrograman Java, JavaScript berbeda dengan bahasa pemrograman Java. Untuk penulisannya, JavaScript dapat disisipkan di dalam *file HTML* ataupun dijadikan *file* tersendiri yang kemudian disambungkan dengan dokumen lain yang dituju. JavaScript mengimplementasikan fitur yang dirancang untuk mengendalikan bagaimana sebuah halaman web berinteraksi dengan penggunaannya [8].

2.8 Bootstrap

Dalam membangun situs web atau aplikasi web, *bootstrap* dikembangkan oleh tim twitter dan pertama kali muncul pada ajang Hackweek dan kini sudah mulai penyempumaan. *Platform* ini hanya menggunakan sedikit coding CSS dan JavaScript namun tetap bisa membuat *website* yang powerful mengikuti perkembangan browser. *Website* yang menggunakan *bootstrap* akan menjadi *website* yang fleksibel dan nyaman dilihat dari tampilannya. Keunggulan ketika menggunakan *Bootstrap* adalah semua bagian untuk antarmuka pengguna menggunakan *Style CSS* [8]. *Bootstrap* dapat diintegrasikan dengan JavaScript untuk membuat tampilan *website* menjadi lebih menarik dengan efek-efek yang dapat diberikan oleh JavaScript.

2.9 Google Firebase

Firebase merupakan layanan yang disediakan Google yang biasa digunakan untuk mempermudah para *programmer* aplikasi dalam mengembangkan aplikasi. Google Firebase telah menyediakan fitur autentikasi pengguna (menggunakan *email* dan *password*), *storage* (sebagai ruang penyimpanan file) dan *cloud messaging* (dapat digunakan untuk mengirim notifikasi). Selain fitur tersebut, Google Firebase juga menyediakan fitur *realtime database NoSQL* dengan struktur data *Java Script Object Notation* (JSON) yang dapat dengan mudah diakses melalui kode web. Dengan hadirnya Google Firebase telah menghemat waktu para *programmer* ketika ingin mengembangkan aplikasi [9].

2.9.1 Firebase Realtime Database

Firebase Realtime Database adalah sebuah *Cloud-Hosted* yang dapat menyimpan dan melakukan sinkronisasi data secara *realtime* untuk setiap *client* yang terhubung. Saat membuat aplikasi lintas platform dengan SDK iOS, Android, dan Javascript, semua *client* akan berbagi sebuah *instance Realtime Database* dan menerima *update* data terbaru secara otomatis [10].

2.10 Use Case Diagram

Use case atau disebut juga *use case diagram* merupakan sebuah pemodelan untuk perilaku sebuah sistem informasi yang akan dibuat. *Use case* mempresentasikan sebuah interaksi antara satu aktor atau lebih dengan sebuah sistem beserta fungsionalitas. Penamaan *use case* dibuat sesederhana mungkin agar mudah dipahami.

2.11 Diagram Aktivitas

Diagram aktivitas atau *activity diagram* adalah sebuah diagram yang menggambarkan aliran kerja (*workflow*) atau aktivitas dari sebuah sistem atau menu yang ada pada perangkat lunak. Aktivitas yang digambarkan adalah

aktivitas sistem bukan yang dilakukan oleh aktor. Diagram aktivitas juga banyak digunakan untuk mendefinisikan banyak hal seperti rancangan proses bisnis setiap urutan atau pengelompokan, dan rancangan menu yang ditampilkan pada perangkat lunak [11].

2.12 Pengertian *Framework*

Framework adalah struktur konseptual dasar yang berisi kumpulan fungsi untuk tujuan tertentu yang sudah siap untuk digunakan, sehingga pembuatan aplikasi dapat dilakukan dengan cepat karena kode programnya tidak di buat dari awal [12].

2.13 *Codeigniter*

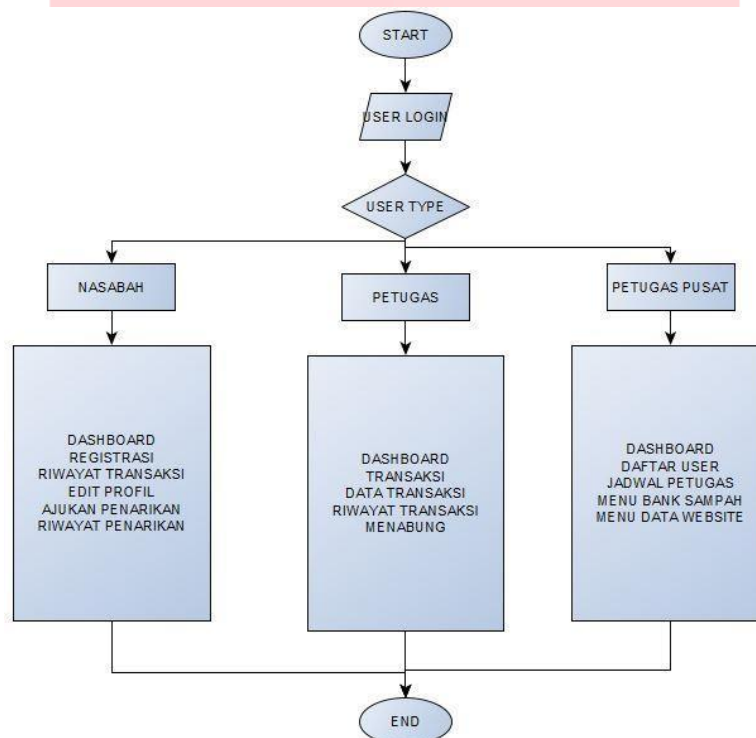
Codeigniter adalah sebuah *framework* php yang bersifat open *source* dan menggunakan metode MVC (*Model, View, Controller*). *Codeigniter* bersifat *free* alias tidak berbayar jika anda menggunakannya. *Framework codeigniter* dibuat dengan tujuan sama seperti *framework* lainnya yaitu untuk memudahkan *developer* atau *programmer* dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal [13].

3. PERANCANGAN DAN IMPLEMENTASI

3.1 Blok Diagram Sistem

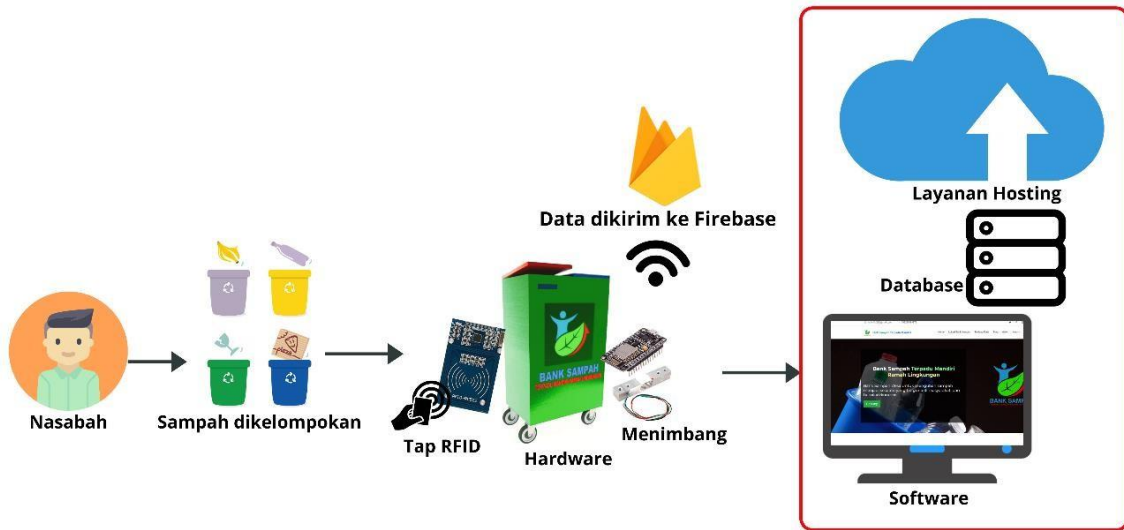
Perancangan pembuatan *website* ini akan dibagi menjadi dua bagian, yang pertama adalah *interface website* yang terdiri dari berbagai bagian. Bagian yang akan ditampilkan berupa halaman selamat datang. Informasi *website* dari Bank Sampah, serta pada bagian atas terdapat *profile* dan *login*. Bagian kedua yaitu server (*database*) yang berfungsi untuk menyimpan semua data pengelolaan sampah penduduk yang telah terdaftar.

Pada sistem akan dijelaskan mengenai rancangan *website* dengan menggunakan *database* berbasis MySQL. Adapun model sistem perancangan yang telah dibuat, dapat dilihat pada gambar dibawah ini.



Gambar 1 Model Sistem Perancangan Website

Cara kerja web *interface* dibedakan menjadi 3 macam hak akses yaitu Nasabah, Petugas dan Petugas pusat. Pertama yaitu hak petugas pusat sebagai akun yang memiliki hak akses menginput data *user*, mengedit data *user*, menghapus data user, menginput jadwal petugas, menginput informasi baru dan terdapat notifikasi jika ada penarikan tabungan dari nasabah. Kemudian hak akses Petugas sebagai akun yang memiliki akses menambah data transaksi, menginput transaksi terbaru, menabung dan melihat riwayat transaksi. Kemudian hak akses Nasabah sebagai akun yang memiliki akses mengedit profil, menginput transaksi baru, melihat riwayat penarikan dan mengajukan penarikan tabungan. Pada proyek akhir ini, bagian yang dikerjakan adalah bagian web *interface* dengan *database*, *authentication* dan *hosting*. Proses akan berlanjut seperti pada gambar dibawah ini.

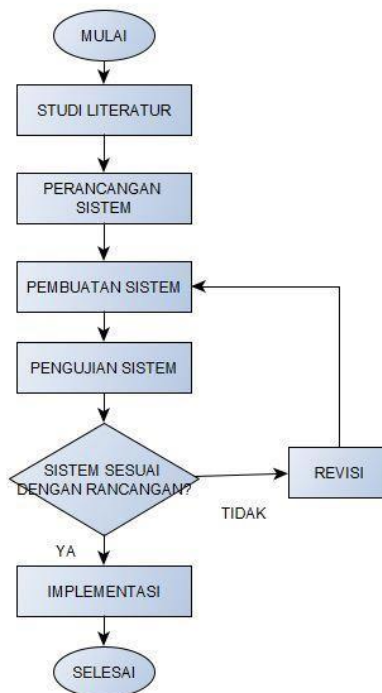


Gambar 2 Blok Diagram Sistem

Pada gambar 2 *software* dirancang dengan data dari *hardware* dengan output berupa data dikirimkan ke firebase tepatnya pada realtime database lalu akan di koneksi ke software dan akan dikirimkan ke layanan hosting supaya website dapat diakses oleh masyarakat.

3.2 Perancangan Pengerjaan Proyek Akhir

Dalam melakukan perancangan untuk proyek akhir ini terdapat beberapa tahapan yang akan dikerjakan. Pada gambar 3 merupakan flowchart langkah pengerjaan proyek akhir ini.



Gambar 3 Flowchart Pengerjaan Proyek Akhir

3.3 Perancangan Use Case Diagram

Berikut ini merupakan tampilan *Use Case Diagram* pada perancangan proyek akhir ini dengan tiga *Use Case diagram* diantaranya sebagai berikut:

3.3.1 Perancangan Use Case Petugas Pusat

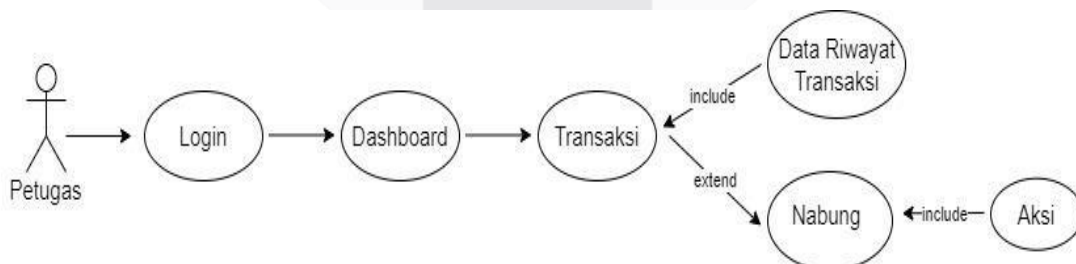
User pertama untuk *website* bank sampah yaitu Petugas Pusat yang memiliki akses penuh sebagai admin di *website* bank sampah.



Gambar 4 Use Case Diagram Petugas Pusat

3.3.2 Perancangan Use Case Petugas Sampah

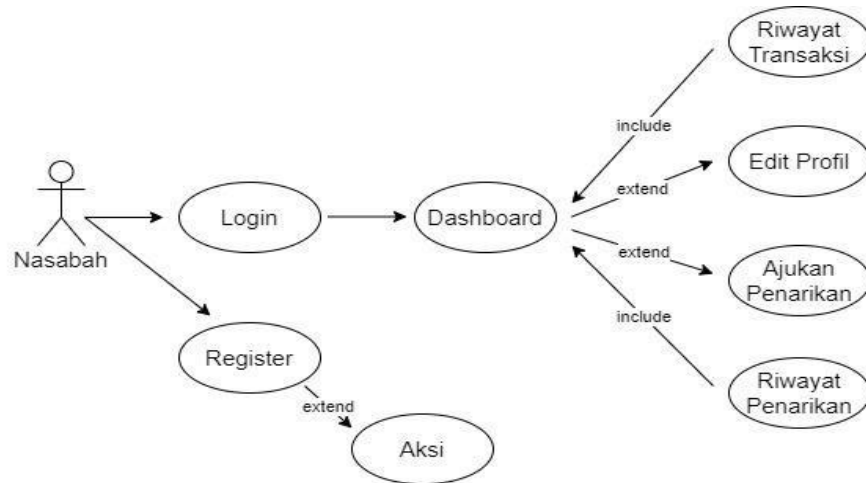
Sebagai *user* kedua yang memiliki akses sebagai orang kedua transaksi dengan nasabah, petugas ini memiliki fitur-fitur penting dalam *website* bank sampah.



Gambar 5 Use Case Diagram Petugas Sampah

3.3.3 Perancangan Use Case Nasabah

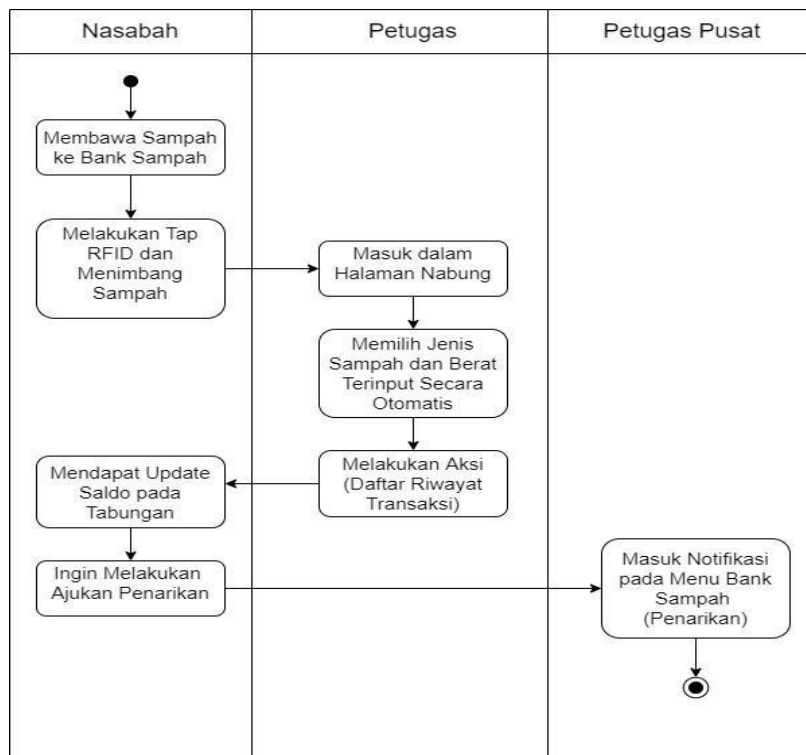
Nasabah merupakan *user* yang akan menggunakan hak penuh sebagai konsumtif di *website* bank sampah dimana nasabah akan diberikan akses untuk menggunakan sistem alat tempat sampah yang telah disediakan oleh produsen.



Gambar 6 Use Case Diagram Nasabah

3.3.4 Diagram Aktivitas

Diagram Aktivitas adalah representasi grafis dari seluruh tahapan alur kerja yang mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas tersebut. Diagram ini dapat digunakan untuk menjelaskan proses bisnis dan alur kerja operasional secara langkah demi langkah dari komponen suatu sistem [3]. Adapun *diagram activity* untuk pengambilan tempat sampah adalah sebagai berikut.



Gambar 7 Diagram Activity

3.4 Skenario Pengujian

Pada tabel berikut merupakan skenario pengujian yang akan dilakukan. Skenario pengujian meliputi pengujian fungsionalitas *website* dan pengujian secara keseluruhan.

Tabel 1 Tabel Skenario Pengujian

Identifikasi	Skenario	Tujuan
Skenario 1	Fungsionalitas Website	Mengetahui apakah seluruh menu berjalan dengan baik
Skenario 2	Pengujian Delay Hardware	Mengetahui delay hardware pada saat proses penginputan data (menabung)
Skenario 3	Performansi Website dengan Lighthouse	Menguji kualitas website menghasilkan seberapa bagus laman itu menjalaninya
Skenario 4	Performansi Website dengan Apache Jmeter	Menguji performa website jika diberi cukup banyak virtual user yang menggunakan web tersebut.

4. HASIL PENGUJUAN DAN ANALISIS

4.1 Pengujian Fungsional

Pengujian fungsionalitas ini bertujuan untuk mengetahui program yang dibuat sesuai dengan awal yang sudah dibuat sebelumnya dengan cara membandingkan data yang diharapkan dengan data pengujian langsung.

Tabel 2 Tabel Pengujian Fungsional Pada Halaman Login

No	Nama Pengujian	Aksi	Harapan Hasil	Hasil Pengerjaan	Hasil
1	Login akun nasabah, petugas, dan admin	Input Username dan Password	Berhasil Login	Berhasil Login	Berhasil
2	Register	Input Username, Password, Alamat, no.Telepon, no.KTP, Foto	Berhasil Menambahkan ke database	Berhasil Menambahkan ke database	Berhasil

Tabel 3 Tabel Pengujian Fungsional Pada Halaman Petugas Pusat

No	Nama Pengujian	Aksi	Harapan Hasil	Hasil Pengujian	Hasil
1	Mengelola Users (Petugas, Nasabah)	Tambah, Edit, Hapus Users	Berhasil mengelola data users	Berhasil mengelola data users	Berhasil
2	Mengelola Jadwal Petugas	Tambah, Edit, Hapus Jadwal	Berhasil mengelola jadwal	Berhasil mengelola jadwal	Berhasil
3.	Mengelola Kategori Sampah	Tambah, Edit, Hapus Kategori Sampah	Berhasil mengelola kategori sampah	Berhasil mengelola kategori sampah	Berhasil
4	Mengelola Jenis sampah	Tambah, Edit, Hapus Jenis Sampah	Berhasil mengelola jenis sampah	Berhasil mengelola jenis sampah	Berhasil
5	Mengelola Riwayat Transaksi	Melihat data Transaksi Nasabah	Berhasil Mendapat up-date data	Berhasil Mendapat up-date data	Berhasil
6	Mengelola Penarikan	Melakukan persetujuan	Berhasil melakukan aksi	Berhasil melakukan aksi	Berhasil

		penarikan tabungan			
7	Artikel	Tambah, Edit, Hapus Artikel	Dapat memberikan update Artikel	Dapat memberikan update Artikel	Berhasil
8	Galeri	Tambah, Edit, Hapus Galeri	Dapat memberikan up-date Galeri	Dapat memberikan up-date Galeri	Berhasil
9	Kategori	Tambah, Edit, Hapus Kategori	Dapat memberikan update Kategori	Dapat memberikan update Kategori	Berhasil
10	Informasi Web	Tambah, Edit, Hapus Informasi Web	Dapat memberikan update Informasi Web	Dapat memberikan update Informasi Web	Berhasil
11	Login	Login akun	Berhasil Login	Berhasil Login	Berhasil
12	Logout	Logout akun	Berhasil Logout	Berhasil Logout	Berhasil

Tabel 4 Tabel Pengujian Fungsional Pada Halaman Petugas

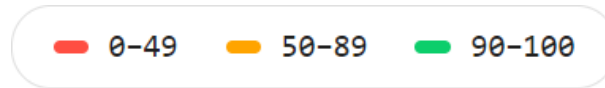
No	Nama Pengujian	Aksi	Harapan Hasil	Hasil Pengerjaan	Hasil
1	Transaksi	Dapat melakukan aksi nabung pada halaman transaksi	Hasil menimbang berat sampah dapat menjadi tabungan	Hasil menimbang berat sampah dapat menjadi tabungan	Berhasil
2	Login	Login Akun	Berhasil Login	Berhasil Login	Berhasil
3	Logout	Logout Akun	Berhasil Logout	Berhasil Logout	Berhasil
4	Monitoring Berat	Menampilkan Berat Sampah dari firebase	Berhasil menampilkan secara realtime	Berhasil menampilkan secara realtime	Berhasil

Tabel 5 Tabel Pengujian Fungsional Pada Halaman Nasabah

No	Nama Pengujian	Aksi	Harapan Hasil	Hasil Pengujian	Hasil
1	Edit Profil	Mengedit informasi dan data akun	Dapat mengedit akun dan menyimpan	Dapat mengedit akun dan menyimpan	Berhasil
2	Ajukan Penarikan	Melihat riwayat penarikan tabungan	Dapat melihat riwayat penarikan tabungan	Dapat melihat riwayat penarikan tabungan	Berhasil
3	Riwayat Penarikan	Melihat riwayat penarikan saldo tabungan	Dapat melihat riwayat penarikan saldo tabungan	Dapat melihat riwayat penarikan saldo tabungan	Berhasil
4	Riwayat Transaksi	Melihat riwayat menabung	Dapat melihat riwayat menabung	Dapat melihat riwayat menabung	Berhasil
5	Login	Login akun	Berhasil login	Berhasil login	Berhasil
6	Logout	Logout akun	Berhasil logout	Berhasil logout	Berhasil

4.2 Pengujian Website Menggunakan Lighthouse

Lighthouse adalah alat (bantu) sumber terbuka otomatis untuk meningkatkan kualitas aplikasi web. Dapat bisa menjalankannya sebagai Ekstensi Chrome atau dari baris perintah. Dengan cara memberi *Lighthouse* sebuah URL yang ingin di audit, maka *Lighthouse* menjalankan serangkaian pengujian terhadap laman tersebut, kemudian menghasilkan sebuah laporan mengenai seberapa bagus laman itu menjalaninya.



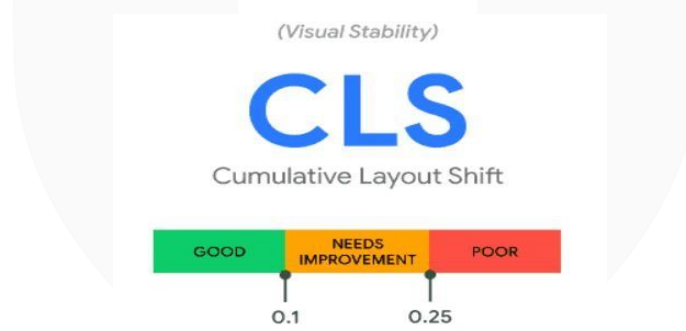
Gambar 8 Parameter Score Lighthouse

Pada Gambar 8 menunjukkan index parameter untuk menentukan bagus atau tidaknya halaman yang telah diukur tersebut. Score parameter 0 - 49 menunjukkan bahwa halaman yang diukur jelek dan banyak yang harus dioptimalkan, score parameter 50-89 menunjukkan bahwa halaman yang diukur normal, score parameter 90-100 menunjukkan bahwa halaman yang diukur sangat baik. Adapun beberapa parameter dalam pengujian performa website menggunakan lighthouse yaitu :

- First Contentful Paint (FCP) adalah waktu di mana gambar atau teks pertama muncul.
- Speed Index (SI) menunjukkan seberapa cepat isi halaman terlihat jelas. Semakin rendah nilainya semakin bagus.
- Time to Interactive (Ttl) adalah waktu pertama yang menunjukkan sebuah halaman dianggap interaktif secara penuh.
- Largest Contentful Paint (LCP) waktu yang dibutuhkan sebuah website dalam menyajikan konten utama serta siap untuk berinteraksi.
- Total Blocking Time (TBT) mengukur jumlah total waktu halaman diblokir dari menanggapi masukan pengguna, seperti klik mouse, ketukan layar, atau penekan keyboard. Jumlahnya dihitung dengan menambahkan porsi pemblokiran dari FCP dan Ttl.

Cumulative Layout Shift (CLS) adalah metrik yang mengukur pergeseran tata letak yang tak terduga pada suatu halaman. Pergeseran ini umumnya terjadi saat halaman di download.

Pengujian server dilakukan untuk mengetahui apakah semua server telah berjalan dan menerima proses registrasi oleh user client. Pada gambar 3, 4, dan 5 menunjukkan setiap server telah berjalan dan menerima registrasi yang selanjutnya akan digunakan untuk melayani panggilan.



Gambar 9 Parameter Score Cumulative Layout Shift

Tabel 6 Tabel Pengujian Pada Halaman Login dan Register

No	Nama Halaman (URL)	Performa	Performa					
			FCP (s)	SI (s)	Ttl (s)	LCP (s)	TBT (ms)	CLS
1	Index.php	64	3.1	4.4	5.1	8.0	130	0.057
2	Login.php	80	3.1	3.1	3.6	3.9	70	0.165
3	Daftar.php	71	3.1	3.7	4.6	3.9	180	0.538

Tabel 7 Tabel Pengujian Website Pada Halaman Petugas Pusat

No	Nama Halaman (URL)	Performa	Performa					
			FCP (s)	SI (s)	Ttl (s)	LCP (s)	TBT (ms)	CLS
1	Admin.php	72	3.8	4.2	3.9	4.4	10	0.189
2	Users.php	76	3.1	4.2	4.1	3.9	280	0.189
3	Jadwal.php	77	3.1	3.1	4.1	3.9	280	0.189
4	Kategori.php	79	3.1	3.2	4.0	3.9	190	0.189
5	Jenis.php	79	3.1	3.4	4.0	3.9	160	0.189
6	Transaksi.php	73	3.1	4.4	3.7	3.9	340	0.189
7	Penarikan.php	76	3.1	3.1	3.8	3.9	330	0.189
8	Artikel.php	76	3.1	3.1	4.1	3.9	340	0.189
9	Galeri.php	78	3.1	3.1	4.0	3.9	260	0.189
10	Dashboard.php	80	3.1	3.1	3.9	3.9	170	0.189

Tabel 8 Tabel Pengujian Website Pada Halaman Petugas

No	Nama Halaman (URL)	Performa	Performa					
			FCP (s)	SI (s)	Ttl (s)	LCP (s)	TBT (ms)	CLS
1	Petugas.php	80	3.1	3.6	3.6	3.9	110	0.189
2	Transaksi.php	78	3.1	3.1	3.9	4.1	180	0.189

Tabel 9 Tabel Pengujian Website Pada Halaman Petugas

No	Nama Halaman (URL)	Performa	Performa					
			FCP (s)	SI (s)	Ttl (s)	LCP (s)	TBT (ms)	CLS
1	Nasabah.php	79	3.1	3.4	3.7	3.9	220	0.189
2	Detail.php	74	3.3	3.5	4.1	4.2	260	0.189
3	Penarikan.php	79	3.0	3.0	3.8	3.9	230	0.189
4	Penarikan/Riwayat.php	78	3.1	3.1	4.0	3.9	260	0.189
5	Transaksi/Riwayat.php	78	3.1	3.8	4.0	3.9	190	0.189

4.3 Pengujian Delay dari Alat ke Website

Pengujian Delay dilakukan untuk mengetahui rata-rata waktu yang diperlukan untuk mengirim dan menerima data dalam satuan detik. Pengujian delay ini melakukan pengukuran dari alat bank sampah hingga ke halaman website.

4.4.1 Pengujian Terhadap Keakuratan Timbangan

Tabel 10 Tabel Pengujian Keakuratan Timbangan

No. Percobaan	Kapasitas		Delay (s)
	Berat Sampah (g)	Web	
1	80	80	2.03
2	169	169	2.54
3	254	254	2.66
4	464	464	2.13
5	513	513	2.15
6	674	674	2.76
7	775	775	2.31
8	880	880	2.56
9	911	911	2.44
10	1000	1000	2.89
Rata-rata			2.23

Pada pengujian Tabel 10 dilakukan dengan cara memasukan sampah oleh nasabah, petugas bank sampah akan menerima data dari firebase ke halaman web petugas berupa berat dari sampah dalam satuan gram (g). Kemudian didapatkan hasil delay, delay pada pengujian kali ini didapatkan dari sisi website ke firebase dan dari firebase ke alat dan dipengaruhi oleh jaringan yang digunakan. Diperoleh hasil dari pengujian rata-rata delay pengiriman adalah 2.23 detik.

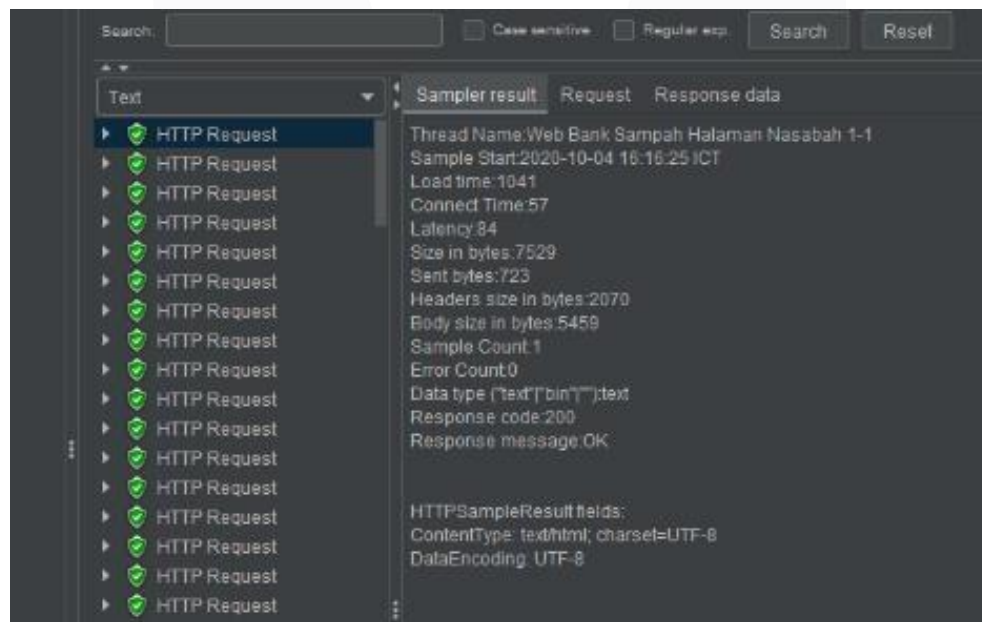
4.4.2 Pengujian Ketepatan Pembacaan Tak RFID

Tabel 11 Tabel Pengujian Ketepatan Pembacaan RFID

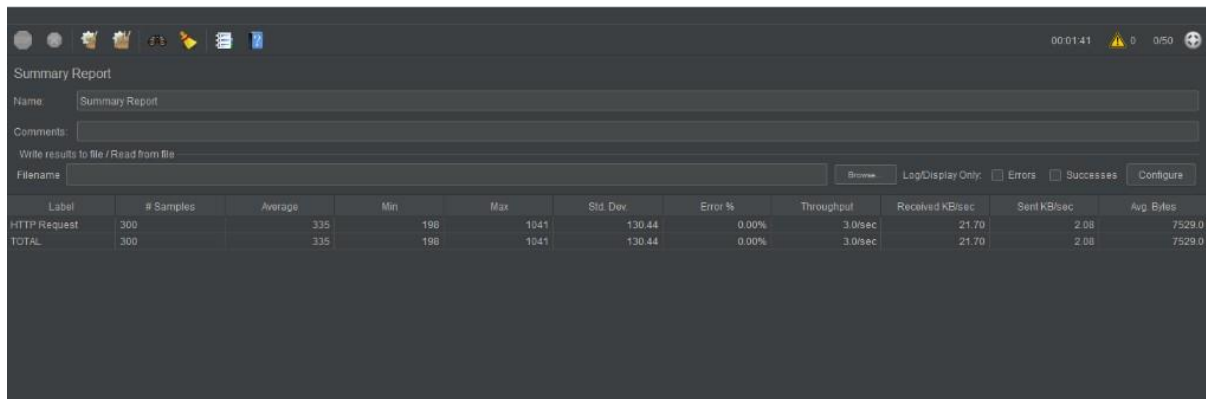
Pengujian	Status Realtime Firebase	Kondisi RFID	Kondisi Door Lock	Persentase Error (%)
Tag terdaftar 1	✓	✓	✓	0
Tag terdaftar 2	✓	✓	✓	0
Tag terdaftar 3	✓	✓	✓	0
Tag tidak terdaftar 4	✓	✓	✓	0
Tag tidak terdaftar 5	✓	✓	✓	0
Tag tidak terdaftar 6	✓	✓	✓	0

Pada pengujian sensor RFID dapat disimpulkan bahwa sensor RFID dapat membaca tag RFID dengan baik, hal ini dibuktikan dengan ketiga tag RFID yang sudah terdaftar ditempelkan ke sensor RFID status alat pada firebase bernilai 1 dan data RFID terbaca sehingga sensor doorlock dapat terbuka, begitu juga sebaliknya ketika tag RFID tidak terdaftar di tempelkan pada sensor RFID status firebase bernilai 0, data RFID tidak terbaca, dan door lock tidak terbuka.

4.4.3 Pengujian Stress Load Testing menggunakan Apache Jmeter



Gambar 10 Hasil Result in Tree



Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename: Log/Display Only Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev	Error %	Throughput	Received KB/Sec	Sent KB/Sec	Avg. Bytes
HTTP Request	300	335	198	1041	130.44	0.00%	3.0/sec	21.70	2.08	7529.0
TOTAL	300	335	198	1041	130.44	0.00%	3.0/sec	21.70	2.08	7529.0

Gambar 11 Hasil Listener Summary Report

5. KESIMPULAN

Berdasarkan hasil perancangan, pengujian dan Analisa hasil yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

1. Sistem informasi bank sampah berbasis internet of things telah berhasil diimplementasikan pada hardware.
2. Berdasarkan tahap-tahap pengujian yang sudah dilakukan, fitur pada website ini sesuai dengan harapan dan berjalan dengan baik
3. Hasil pengujian fungsionalitas website sudah sesuai dengan yang diharapkan dan 100% berhasil.
4. Berdasarkan data pengujian performa website yang menggunakan tools light house yang telah dilakukan, didapatkan rata – rata nilai skor parameter tiap halaman website yaitu 75 bahwa dinyatakan performa website normal dari setiap halaman.
5. Berdasarkan data pengujian delay yang telah dilakukan didapatkan nilai rata – rata delay sebesar 2.23 detik.
6. Ketepatan pembacaan dari sensor RFID sangat baik, karena pada pengujian keberhasilan alat 100%. Baik dalam tag RFID yang sudah terdaftar ataupun yang belum terdaftar.
7. Pada Pengujian *Stress Load Testing* di dapatkan nilai rata-rata respon yang diberikan server untuk setiap request 335 millisecond yaitu 0.335 detik.

REFERENCE

- [1] M. W. R. Fakhrun and S. F. S. Gumilang, "Rancang Web Service Dengan Metode REST API Untuk Integrasi Aplikasi Mobile dan Website Pada Bank Sampah," in *Konferensi Nasional Sistem Informasi 2018*, Pangkalpinang, 2018.
- [2] M. Dewanti, E. P. Purnomo and L. Salsabila, "ANALISA EFEKTIVITAS BANK SAMPAH SEBAGAI ALTERNATIF PENGELOLAAN SAMPAH DALAM MENCAPAI SMART CITY DI KABUPATEN KULON PROGO," *Jurnal Ilmu Administrasi Publik*, vol. 5 (1) 2020, no. PUBLISIA, pp. 21-29, 2020.
- [3] A. Sansprayada and K. Mariskhana, "IMPLEMENTASI APLIKASI BANK SAMPAH BERBASIS ANDROID STUDI KASUS PERUMAHAN VILLA DAGO TANGERANG SELATAN," *Jurnal Inovasi Informatika Universitas Pradita*, vol. V, pp. 24-234, 2020.
- [4] S. Anwar, "RANCANGBANGUN DAN IMPLEMENTASI SISTEM INFORMASI PENGELOLAAN SAMPAH DAN TEMPAT OLAH SAMPAH SETEMPAT TERPADU MANDIRI (TOSS TM) DENGAN KONTROL TERPUSAT BERBASIS INTERNET OF THINGS," Universitas Telkom, Bandung, 2019.
- [5] P. Hidayatullah and J. K. Kawistara, *Pemrograman Web*, Bandung: Informatika, 2017.
- [6] H. Priyanto and K. K. Jauhari, "Pemrograman WEB Edisi Revisi," Penerbit Informatika, Bandung, 2017.
- [7] A. Saputra and R. T. Subagjo, *Membangun Aplikasi E-Library untuk Panduan Skripsi*, Jakarta: Elex Media Komputindo, 2017.
- [8] NIAGAHOSTER blog, "Pengertian PHP dan Fungsinya," Penyedia Layanan Hosting dan Domain, 17 Januari 2019. [Online]. Available: <https://www.niagahoster.co.id/blog/pengertian-php/>. [Diakses 29 september 2020].
- [9] M. Ilhami, "Pengenalan Google Firebase Untuk Hybrid Mobile Apps Berbasis Cordova," *IT CIDA*, vol. 3, 2017.
- [10] Google, "Firebase Realtime Database," Developers, 2017. [Online]. Available: <https://firebase.google.com/docs/database/>. [Diakses 29 September 2020].

- [11] R. A. Sukamto and S. M, *Rekayasa Perangkat Lunak*, Bandung: Informatika Bandung, 2013.
- [12] CloudHost, “Mengenal Apa itu Framework CodeIgniter,” Hosting Provider, 4 Agustus 2017. [Online]. Available: <https://idcloudhost.com/panduan/mengenal-apa-itu-framework-codeigniter/>. [Diakses 29 September 2020].
- [13] NIAGAHOSTER blog, “Yuk Mengenal Apa Itu Framework dan Fungsinya,” Penyedia Layanan Hosting dan Domain, 13 May 2020. [Online]. Available: <https://niagahoster.co.id/blog/apa-itu-framework/>. [Diakses 29 September 2020].
- [14] FERRY, “STRESS TESTING,” Binus University, 13 May 2019. [Online]. Available: <https://sis.binus.ac.id/2019/05/13/stress-testing/>. [Diakses 1 oktober 2020].

