

Rancang Bangun Website SARANG (Sampah Terang) untuk Membantu Masyarakat Kota Mataram dalam Jual Beli Sampah Daur Ulang secara Online

(Design and Build a Website Application to Help Citizen of Mataram City in Buying and Selling Recycled Waste)

Andy Hidayat Jatmika, Muhammad Lanang Agung Anggoro
Dept Informatics Engineering, Mataram University
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA
Email: andy@unram.ac.id, lanangagung11@gmail.com

Abstrak

Kota Mataram, yang terletak di Pulau Lombok, Provinsi Nusa Tenggara Barat (NTB) menghadapi tantangan yang signifikan terkait manajemen sampah. Pertumbuhan penduduk yang cepat dan aktivitas ekonomi yang berkembang telah menghasilkan peningkatan produksi sampah, yang dapat memiliki dampak negatif terhadap lingkungan dan kesehatan masyarakat. Daur ulang adalah salah satu solusi yang sangat efektif untuk mengatasi permasalahan sampah yang semakin memburuk di berbagai komunitas di seluruh dunia, termasuk di Kota Mataram. Berdasarkan hasil survey dan wawancara, banyak masyarakat di Kota Mataram yang berusaha untuk berkontribusi dalam upaya daur ulang sampah, tetapi mereka sering kali merasa kebingungan tentang bagaimana menjual sampah daur ulang dengan efisien dan menguntungkan. Untuk mengatasi permasalahan tersebut, diusulkan pembuatan website SARANG (Sampah Terang) sebagai solusi inovatif untuk mengatasi tantangan dalam penjualan dan pembelian sampah daur ulang. Website ini bertujuan untuk memberi kemudahan bagi pembeli untuk menemukan pasokan bahan baku daur ulang yang berkualitas dan murah. Dari uji coba yang dilakukan selama 3 hari yang dimulai dari tanggal 7 – 10 Agustus 2023 didapatkan 22 responden dengan nilai rata – rata 10% menjawab sangat kurang setuju, 2.7% tidak setuju, 3.62% cukup, 36.36% setuju, dan 47.32% menjawab sangat setuju.

Kata kunci : Daur ulang Sampah, Website pengelolaan sampah, MongoDB, ExpressJS

PENDAHULUAN

1.1. Latar Belakang

Pertumbuhan penduduk Kota Mataram yang cepat dan aktivitas ekonomi yang berkembang telah menghasilkan peningkatan produksi sampah, yang dapat memiliki dampak negatif terhadap lingkungan dan kesehatan masyarakat. Sampah merupakan persoalan besar dalam rumah dan lingkungan. Penyelesaian masalah sampah harus menyeluruh dari hulu ke hilir dan seluruh pihak turut terlibat supaya persoalan sampah dapat diatasi [10]. Sampah memiliki berbagai macam jenis yaitu sampah padat, sampah cair, dan sampah dalam bentuk gas. Berdasarkan zat kimia yang dikandungnya, sampah dikelompokkan menjadi sampah anorganik dan sampah organik. Sampah anorganik adalah sampah yang umumnya tidak dapat membusuk, misalnya logam atau besi, pecahan gelas, dan plastik. Sampah organik adalah jenis sampah yang dapat membusuk, misalnya sisa-sisa makanan, daun-daun, dan buah-buahan [1]. Daur ulang adalah salah satu solusi yang sangat efektif untuk mengatasi permasalahan sampah yang semakin memburuk di berbagai komunitas di seluruh dunia, termasuk di Kota Mataram.

Berdasarkan hasil *survey* dan wawancara yang telah dilakukan, saat ini Masyarakat masih kesulitan mencari informasi tentang tempat-tempat jual beli sampah daur ulang di Kota Mataram. Beberapa masyarakat ada yang telah menemukan beberapa tempat, tetapi sebagian besar telah tutup atau tidak beroperasi. Masyarakat juga tidak tahu berapa harga yang diharapkan untuk sampah-sampah yang mereka miliki untuk dijual. Dalam era digital yang terus berkembang, aktivitas jual beli sampah daur ulang secara *online* melalui *website* dapat menjadi solusi yang memudahkan banyak orang, termasuk warga Kota Mataram

Maka dalam permasalahan yang telah dijabarkan, diusulkan pembuatan *website* SARANG (Sampah Terang) untuk sebagai solusi inovatif untuk mengatasi tantangan dalam penjualan dan pembelian sampah organik. *Website* ini bertujuan untuk memberi kemudahan bagi pembeli untuk menemukan pasokan bahan baku daur ulang yang berkualitas dan murah. Dan *website* ini juga membantu penjual untuk bisa memudahkan pembeli untuk menjual

sampah yang telah dikumpulkan dan pupuk kompos yang telah diolah. *Website* ini dibangun menggunakan *framework* Express JS dan MongoDB.

Dari hasil uji coba yang dilakukan selama 3 hari yang dimulai dari tanggal 7 – 10 Agustus 2023 didapatkan 22 responden dengan nilai rata – rata 10% menjawab sangat kurang setuju, 2.7% tidak setuju, 3.62% cukup, 36.36% setuju, 47.32% sangat setuju. Berdasarkan hasil kuesioner tersebut, *website* yang dibangun telah sesuai dengan apa yang diharapkan oleh Masyarakat.

TINJAUAN PUSTAKA

2.1 *Application Programming Interface (API)*

API merupakan antarmuka yang bisa digunakan fitur lunak buat mengakses aplikasi ataupun layanan. Alih-alih mengawali dari dini, API memungkinkan pengembang untuk menggunakan kembali fungsionalitas yang telah terdapat lebih dahulu dari program lain. API mengacu pada pemanggilan guna di *website* yang memakai *Hyper Text Transfer Protocol (HTTP)* serta mengembalikan *respons* dalam *JavaScript Object Notation (JSON)* ataupun *Extensible Markup Language (XML)* [1]. Tujuan pemakaian API merupakan buat mentransfer informasi antar aplikasi yang berbeda, tetapi tujuan pemakaian API lain merupakan buat memesatkan pengembangan aplikasi dengan menawarkan guna yang berbeda sehingga pengembang tidak butuh membuat fitur seragam. Aplikasi bisa berkomunikasi dengan susunan dasar dan satu sama lain menggunakan API tingkat sistem operasi.

2.2 *Representational State Transfer (REST)*

Memanfaatkan antarmuka standar, seperti HTTP, REST merupakan sekumpulan konsep arsitektur. Seperti aplikasi *online* pada biasanya, REST API beroperasi. Klien memperoleh memanfaatkan protokol HTTP guna membuat permintaan ke *server*, yang setelah itu mengembalikan *respons* ke klien [1]. Dalam arsitektur REST, sumber daya (sumber daya/ informasi) disediakan oleh *server* REST, yang diakses serta ditampilkan oleh klien REST buat digunakan nanti. URI (Pengidentifikasi Sumber Daya Umum), kerap diketahui sebagai ID global, digunakan buat mengenali tiap sumber daya. Sumber daya ditampilkan selaku bacaan, JSON, ataupun XML. JSON serta XML biasanya digunakan dalam *format*.

2.3 *Activity Diagram*

diagram menggambarkan *workflow* (alur kerja) aktivitas dari sebuah sistem. Diagram menyerupai *flowchart* karena memodelkan *workflow* aktivitas dari satu aktivitas ke aktivitas lainnya. Pembuatan activity diagram pada awal pemodelan proses dapat membantu memahami keseluruhan proses [5]. Disamping itu, *Activity* diagram digunakan untuk menggambarkan interaksi antara beberapa *use case*.

2.4 *Use Case Diagram*

Use case diagram merupakan pemodelan untuk perilaku (*behavior*) sistem informasi yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [5].

2.5 *Entity Relationship Diagram (ERD)*

ERD adalah model jaringan data yang menekankan pada struktur-struktur dan *relationship* data. ERD melihat objek nyata dapat sebagai sebuah entitas - entitas yang memiliki relasi antara entitas yang satu ataupun yang lain. Dengan ERD sendiri dapat membantu mengurangi kesalahan-kesalahan dalam melakukan perancangan *database* dari gambaran dunia nyata dan struktur *database* seperti *redundancy* data, hubungan-hubungan antara entitas, dan lain sebagainya. Elemen - elemen dalam ERD adalah [3].

2.6 *Visual Studio Code*

Visual Studio Code adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang via *marketplace* Visual Studio Code (seperti C++, C#, Python, Go, Java, dst).

Banyak sekali fitur-fitur yang disediakan oleh Visual Studio Code, di antaranya Intellisense, Git Integration, Debugging, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur-fitur tersebut akan terus bertambah

seiring dengan bertambahnya versi Visual Studio Code. Pembaruan versi Visual Studio Code ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan Visual Studio Code dengan teks editor yang lain.

Teks editor Visual Studio Code juga bersifat *open source*, yang mana kode sumbernya dapat dilihat dan dapat berkontribusi untuk pengembangannya. Hal ini juga yang membuat Visual Studio Code menjadi favorit para pengembang aplikasi, karena para pengembang aplikasi bisa ikut serta dalam proses pengembangan Visual Studio Code ke depannya [4].

2.7 Web

Secara teknis, Web adalah sebuah sistem dengan informasi yang disajikan dalam bentuk teks, gambar, suara dan lain-lain yang tersimpan dalam sebuah server Web Internet yang disajikan dalam bentuk hiperteks. Informasi Web dalam bentuk teks umumnya ditulis dalam format HTML (Hypertext Markup Language). Informasi lainnya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG), suara (dalam format AU, WAV), dan objek multimedia lainnya (seperti MIDI, Shockwave, Quicktime Movie, 3D World) [5].

2.8 Netlify

Netlify sendiri merupakan sebuah layanan *hosting* dan pengembangan *static website* yang menyediakan layanan *continuous deployment* yang memungkinkan untuk mengunggah perubahan *website* secara otomatis setiap kali ada perubahan kode di git repository [6]. Netlify menyediakan *hosting* statis yang sangat cepat dan dapat diandalkan untuk situs web dan aplikasi web statis. Anda dapat mengunggah *file* HTML, CSS, dan JavaScript Anda ke Netlify, dan mereka akan meng-*hosting* dan menyajikan situs web Anda dengan kecepatan tinggi melalui jaringan pengiriman konten (CDN) mereka.

2.9 MongoDB

MongoDB merupakan sebuah sistem basis data yang berbasis dokumen (Document Oriented Database) dan termasuk sistem basis data yang menganut paham NoSQL. NoSQL singkatan dari Not Only SQL, artinya sebuah sistem basis data tidak hanya harus menggunakan perintah SQL untuk melakukan proses manipulasi data. MongoDB tidak memiliki tabel, kolom, dan baris. Pada MongoDB yang ada hanyalah koleksi dan dokumen. Dokumen yang terdapat dalam mongoDB dapat memiliki atribut yang berbeda dengan dokumen lain walaupun berada dalam satu koleksi. Hal ini tidak dapat dilakukan dalam RDBMS, di mana sebuah baris dalam tabel tidak mungkin memiliki kolom yang berbeda dengan baris yang lain jika berada dalam satu tabel [7].

2.10 ExpressJS

Express.js adalah kerangka kerja (*framework*) aplikasi web yang berbasis Node.js. Ini adalah salah satu kerangka kerja paling populer untuk pengembangan aplikasi web di sisi server dengan menggunakan JavaScript. Express.js memungkinkan Anda untuk dengan mudah mengelola rute (*routes*) dan permintaan (*requests*) HTTP, serta menyediakan berbagai fitur dan utilitas untuk membangun aplikasi web yang *skalabel* dan efisien. penggunaan Express JS untuk membangun REST API sebagai jembatan antara front-end dan back-end untuk meningkatkan kinerja dan modernisasi *website* tersebut. Karena Express Js memudahkan dalam pengelolaan model, tampilan, dan kontroler. Selain itu Express JS juga mempunyai skalabilitas aplikasi yang baik dan memiliki fitur *caching* di mana kita tidak harus mengeksekusi kode terus - menerus sehingga halaman web dapat memuat lebih cepat dan dapat berpengaruh terhadap sisi performa dari aplikasi yang dibuat [8].

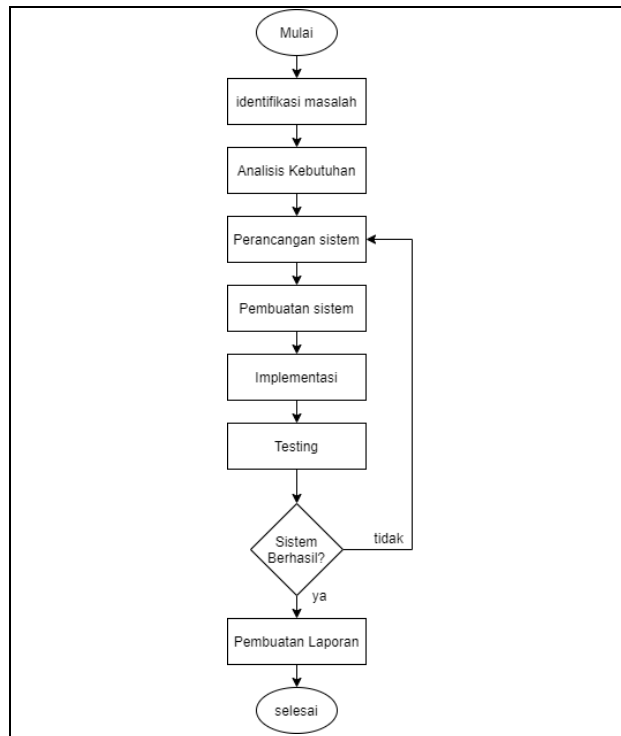
2.11 Vercel

Layanan hosting yang dipilih untuk dijadikan server yaitu Vercel, Hal ini dikarenakan Next JS merupakan produk dari Vercel sehingga Next JS didukung secara penuh. Gitlab dan Vercel dapat diintegrasikan, sehingga apabila Gitlab menerima push dari local project, maka secara otomatis akan diteruskan ke Vercel untuk dilakukan proses deploy [9]. Vercel adalah platform cloud yang memungkinkan Anda untuk dengan mudah menghosting dan men-deploy aplikasi web dan situs web secara cepat. Awalnya dikenal sebagai platform untuk hosting aplikasi Next.js, Vercel telah berkembang menjadi solusi hosting statis dan server-side rendering untuk berbagai jenis proyek web.

METODE PENGABDIAN MASYARAKAT

3.1 Alur Pengembangan Sistem

Gambar 1 merupakan alur pengembangan sistem *website* SARANG (Sampah Terang) yang akan dijelaskan secara bertahap.



Gambar 1. Alur Pengembangan Sistem

Pada Gambar 1, hal pertama yang harus dilakukan adalah melakukan identifikasi masalah terlebih dahulu tujuannya untuk mengetahui kebutuhan yang dibutuhkan. Kemudian dilakukan analisis kebutuhan untuk mengetahui lebih detail lagi apa yang diinginkan oleh sistem. Selanjutnya melakukan perancangan sistem seperti menentukan alur dari program dan bagaimana sistem yang ingin dibuat berdasarkan kebutuhan, dan melakukan bahasa pemrograman yang ingin digunakan, serta merancang *database* untuk menyimpan data relevan dengan proyek, seperti informasi pengguna, data sampah, dan lainnya. Kemudian melakukan pembuatan sistem untuk mengimplementasikan logika dan fungsionalitas *backend* berdasarkan desain dan arsitektur yang telah dibuat. Dan juga mengintegrasikan *frontend* dengan *backend* untuk memastikan data dan informasi dapat diproses dan ditampilkan dengan benar. Kemudian melakukan implementasi, menyiapkan wadah untuk pengembangan *backend*, baik itu server lokal atau *cloud*, untuk meng-*host* aplikasi *backend*. Dan memastikan aplikasi *backend* berjalan lancar dan dapat diakses melalui *endpoint* yang sesuai. Kemudian melakukan testing pada aplikasi *backend* untuk memastikan bahwa fungsi-fungsi yang diharapkan berjalan dengan benar. Dan berkolaborasi dengan *frontend* developer untuk menguji fungsionalitas secara keseluruhan. maka akan dilanjutkan dengan pembuatan laporan. Namun sebaliknya, jika program belum berhasil, maka akan dilakukan perbaikan dimulai dari perancangan sistem, dan akan terus diulang sampai program berhasil dan selesai.

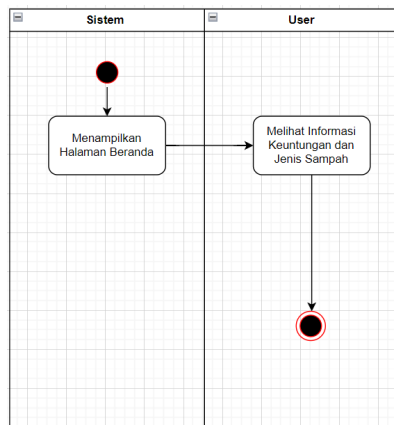
3.2 Desain Sistem

Pada sistem aplikasi SARANG (Sampah Terang) memiliki desain yang akan dijelaskan secara detail pada sub bab ini.

3.2.1 Activity Diagram

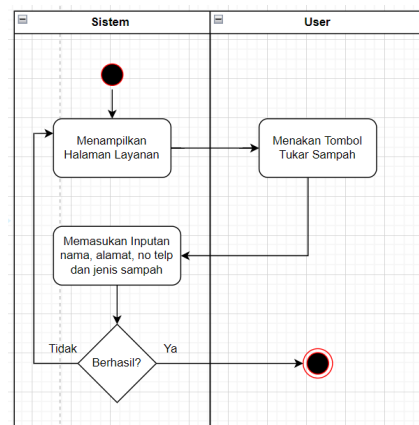
Activity diagram dari Sistem Informasi Inventaris Barang Badan Pengelolaan Pendapatan Daerah Provinsi Nusa Tenggara Barat dapat dilihat pada sub bab ini.

a. Activity Diagram Halaman Beranda



Gambar 2. Activity Diagram Halaman Beranda

b. Activity Diagram Tukar Sampah

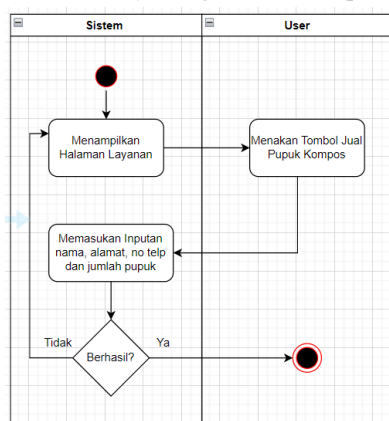


Gambar 3. Activity Diagram Tukar Sampah

Pada Gambar 2. dapat dijelaskan bahwa saat *user* pertama kali mengakses *website* SARANG (Sampah Terang) akan melihat tampilan beranda yang berisikan tentang informasi keuntungan daur ulang sampah dan berbagai macam jenis – jenis sampah.

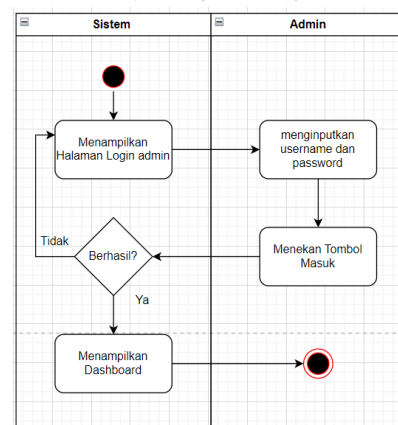
Pada Gambar 3. dapat dijelaskan bahwa saat *user* ingin menukar sampah *user* harus terlebih dahulu membuka halaman layanan. Kemudian pada halaman layanan terdapat fitur tukar sampah. Kemudian *user* diharuskan untuk mengisi *form*, yang di mana pada *form* tersebut terdapat beberapa data informasi yang harus diisi untuk melakukan transaksi. Pada *form* tersebut *user* diharuskan untuk meng-*input*-kan nama. Alamat, no telepon, jenis sampah dan jumlah sampah. *Form* Ini bertujuan untuk mengetahui siapa saja yang akan menukarkan sampahnya pada *website* SARANG (Sampah Terang). Kemudian setelah *user* mengisi *form* tersebut nantinya akan muncul *popup* sebagai bukti transaksi. Apabila gagal akan kembali ke menu layanan.

c. Activity Diagram Jual Pupuk



Gambar 4. Activity Diagram Jual Pupuk

d. Activity Diagram Login Admin



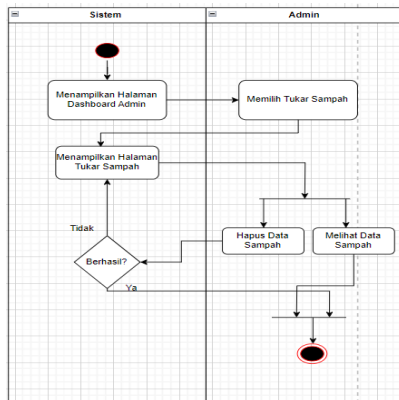
Gambar 5. Activity Diagram Login Admin

Pada Gambar 4. dapat dijelaskan bahwa saat *user* ingin menukar sampah *user* harus terlebih dahulu membuka halaman layanan. Kemudian pada halaman layanan terdapat fitur Jual pupuk. Kemudian *user* diharuskan untuk mengisi *form*, yang dimana pada *form* tersebut terdapat beberapa data informasi yang harus diisi untuk melakukan transaksi. Pada *form* tersebut *user* diharuskan untuk menginputkan nama. Alamat, no telepon, jenis pupuk dan jumlah sampah. *Form* Ini bertujuan untuk mengetahui siapa saja yang akan menjual pupuknya pada *website* SARANG (Sampah Terang). Kemudian setelah *user* mengisi *form* tersebut nantinya akan muncul *popup* sebagai bukti transaksi. Apabila gagal akan kembali ke menu layanan.

Pada Gambar 5 dijelaskan bagaimana proses login pada awalnya *user* akan melakukan *login*, setelah *user* melakukan *login* maka akan ditampilkan *form login* lalu memasukan username dan password, apabila *username* dan *password* yang dimasukan benar maka *user* akan masuk ke halaman *Dashboard*. Kemudian

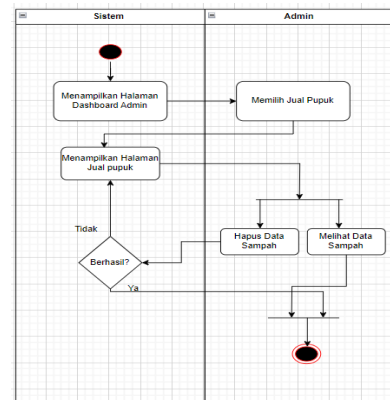
system akan memvalidasi informasi dengan memberikan *username* dan *password* yang telah di-input dan menyimpannya ke dalam *database*. Kemudian akan ditampilkan halaman *login* masuk ke halaman *Dashboard*. Apabila admin salah menginputkan *username* dan *password* maka akan kembali ke halaman *login* admin.

e. *Activity Diagram Data Tukar Sampah*



Gambar 6. *Activity Diagram Data Tukar Sampah*

f. *Activity Diagram Data Jual Pupuk*



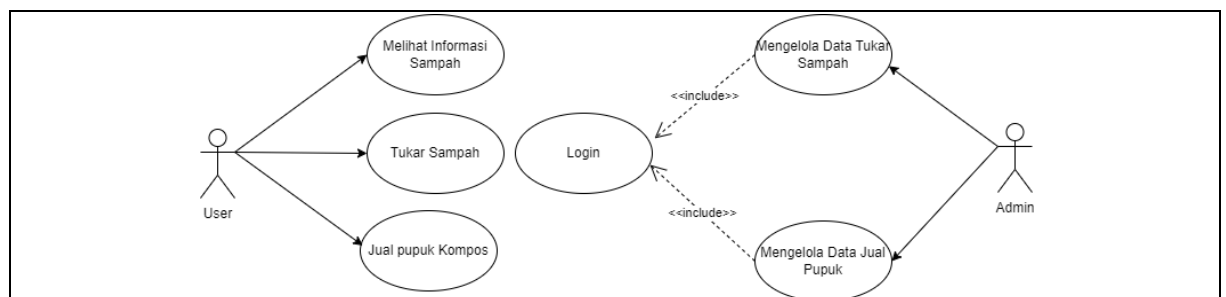
Gambar 7. *Activity Diagram Data Jual Pupuk*

Pada Gambar 6. Pada *Actifity* diagram ini admin akan melihat data tukar sampah. Di mana admin akan melihat halaman *dashboard*. Kemudian admin akan memilih fitur tukar sampah untuk melihat data tukar sampah yang sudah masuk. Kemudian akan menampilkan halaman data tukar sampah. Pada halaman ini admin memiliki dua pilihan yaitu hapus data sampah dan melihat data sampah, apabila sampah sudah berhasil ditukarkan kepada *website* SARANG (Sampah Terang) maka admin akan menghapus data sampah. Dan yang kedua admin dapat melihat data sampah yang masuk dan belum berhasil ditukarkan. Apabila tidak berhasil maka akan kembali ke halaman tukar sampah.

Pada Gambar 7. Pada *Actifity* diagram ini admin akan melihat data jual pupuk. Di mana admin akan melihat halaman *dashbord*. Kemudian admin akan memilih fitur jual pupuk untuk melihat data jual pupuk yang sudah masuk. Kemudian admin dapat melihat halaman data jual pupuk. Pada halaman ini admin memiliki dua pilihan yaitu hapus data jual pupuk dan melihat data jual pupuk, apabila pupuk sudah berhasil ditukarkan kepada *website* SARANG (Sampah Terang) maka admin akan menghapus data jual pupuk. Dan yang kedua admin dapat melihat data jual pupuk yang masuk dan belum berhasil dijual. Apabila tidak berhasil maka akan kembali ke halaman jual pupuk.

3.2.2 Use Case Diagram

Pada sistem aplikasi SARANG (Sampah Terang) memiliki *use case* Sebagai Berikut:



Gambar 8. *Use Case Diagram SARANG*

Pada Gambar 8 terdapat *use case* diagram dari *website* SARANG (Sampah Terang) memiliki dua aktor diantaranya sebagai berikut:

a. *User*

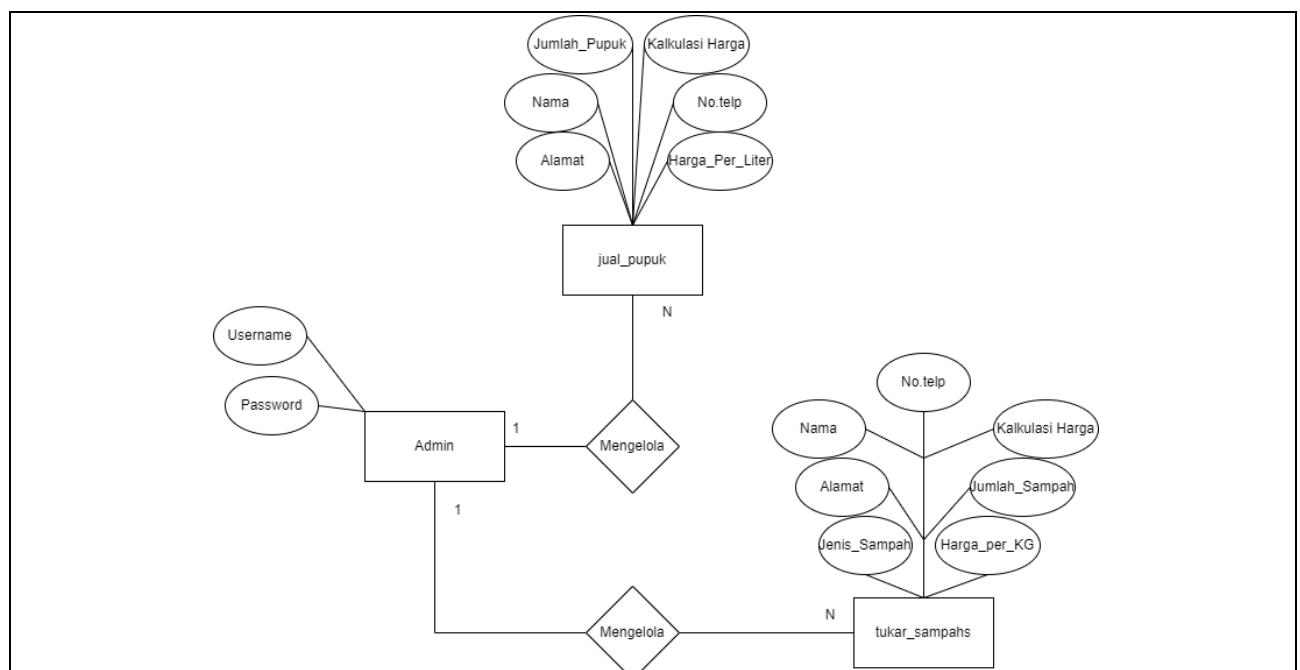
Pada *website* ini *user* memiliki 3 tindakan yang dapat dilakukan. Yang pertama adalah melihat informasi sampah, *user* dapat melihat informasi sampah apa saja yang bisa ditukarkan pada *website* SARANG (Sampah Terang) dan sampah yang dapat ditukarkan adalah sampah organik dan anorganik. Yang kedua *User* dapat menukarkan sampah pada fitur tukar sampah yang sudah terdaftar pada halaman layanan. *User* akan mengisi *form* data diri dan memilih jenis sampah apa yang akan ditukarkan. Kemudian yang terakhir yaitu *user* dapat menjual pupuk kompos yang telah jadi. Sama seperti pada fitur tukar sampah, pada fitur jual pupuk *user* akan mengisi data diri. Kemudian setelah mengisi data diri akan muncul *popup* sebagai tanda bukti bahwa *user* telah berhasil menukarkan sampah dan menjual pupuknya.

b. *Admin*

Dalam *website* ini, admin atau yang mengelola sistem dapat melakukan dua proses, antara lain melihat dan menghapus data tukar sampah dan jual pupuk. Sebelum itu admin diharuskan untuk melakukan *login* terlebih dahulu untuk melakukan melihat dan menghapus data. Yang pertama admin akan melihat data yang telah masuk pada halaman admin dan yang kedua admin akan menghapus data apabila sampah atau pupuk sudah berhasil ditukar dan dijual.

3.2.3 Entity Relationship Diagram

Pada sistem *website* SARANG (Sampah Terang) memiliki ERD yang terdiri dari entitas Admin, Jual_pupuks, dan tukar_sampahs yang dapat dilihat pada Gambar 9.



Gambar 9. ERD Website SARANG

Pada ERD Gambar 9 dapat dilihat bahwa entitas Admin memiliki dua atribut yaitu *username* dan *password*. Pada entitas Admin memiliki relasi *one to many* terhadap entitas *jual_pupuks*. Dan juga entitas admin memiliki relasi yang sama terhadap entitas *tukar_sampah*, yakni *one to many*. Pada entitas *tukar_tukar sampah* memiliki tujuh entitas di antaranya adalah *jenis_sampah*, *alamat*, *nama*, *no.tel*, *kalkulasi_harga*, *jumlah_sampah*, *harga_perkg*. Di mana pada entitas ini di masukkan agar *user* mengisi *form* dari ketujuh entitas tersebut dan entitas ini memiliki relasi dengan entitas admin yaitu *many to one*. Begitu juga dengan entitas *jual_pupuk*.

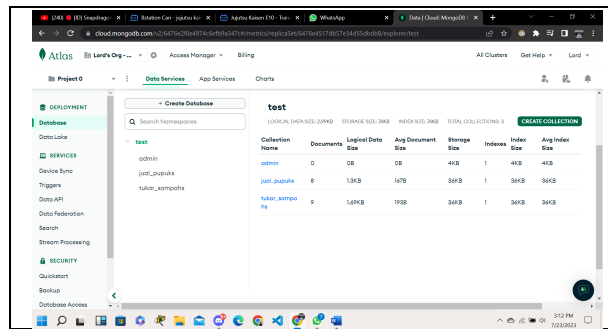
HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Pada bagian ini akan dijelaskan implementasi dari sistem yang dibangun seperti implementasi *database* dan implementasi *interface*.

4.1.1. Implementasi Database

Berikut Pada Gambar 10 merupakan implementasi *database* yang diambil dari ERD *website* SARANG (Sampah Terang):

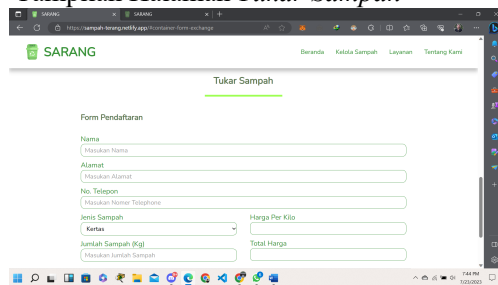


Gambar 10. Implementasi Database

Pada Gambar 10 Merupakan hasil implementasi *database* yang sesuai dengan ERD Sistem Informasi Inventaris Barang Badan Pengelolaan Pendapatan Daerah Provinsi Nusa Tenggara Barat di mana terdapat 4 entitas yg telah di implementasikan menjadi 4 tabel yaitu tabel barang, barang_keluar, *supplier* dan users. Pada implementasi *database* ini penulis menggunakan phpMyAdmin untuk membuat *database*-nya.

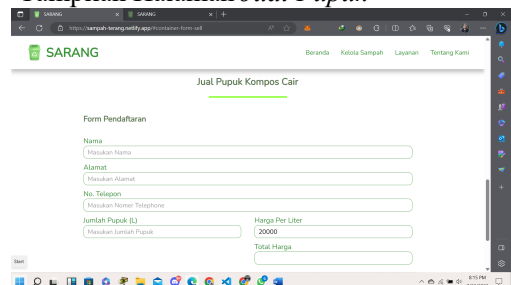
4.1.2. Implementasi Interface

a. Tampilan Halaman Tukar Sampah



Gambar 11 Tampilan Halaman Tukar Sampah

b. Tampilan Halaman Jual Pupuk

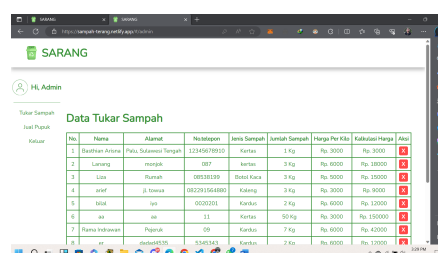


Gambar 12 Tampilan Halaman Jual Pupuk

Pada Gambar 11 di atas terdapat halaman tukar sampah yang dapat dilihat oleh *user*. Pada halaman ini *user* akan mengisi data diri seperti nama, alamat, no. telepon, jenis sampah, dan jumlah sampah.

Pada Gambar 12 di atas terdapat halaman jual pupuk yang dapat dilihat oleh *user*. Pada halaman ini *user* akan mengisi data diri seperti nama, alamat, no. telepon, jenis sampah, dan jumlah pupuk.

c. Menu Data Halaman Admin

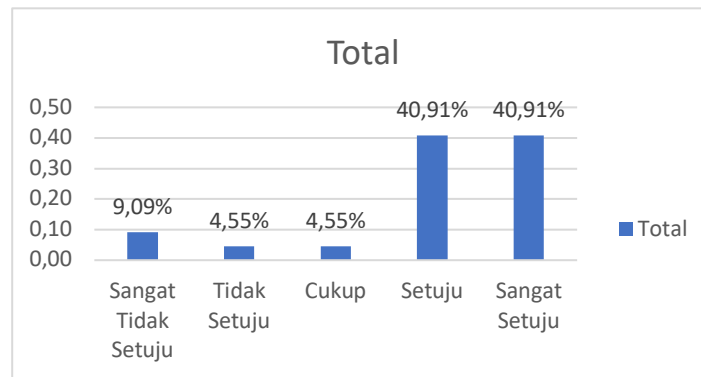


Gambar 13. Data Halaman Admin

Pada Gambar 13 di atas terdapat halaman admin. Di mana pada halaman ini setelah admin melakukan *login* maka akan dipindahkan ke halaman admin. Di halaman ini admin dapat melihat data – data yang telah masuk setelah *user* melakukan transaksi. Pada halaman ini juga admin dapat menghapus data apabila *user* telah selesai melakukan transaksi.

4.1.3. Hasil Koesioner

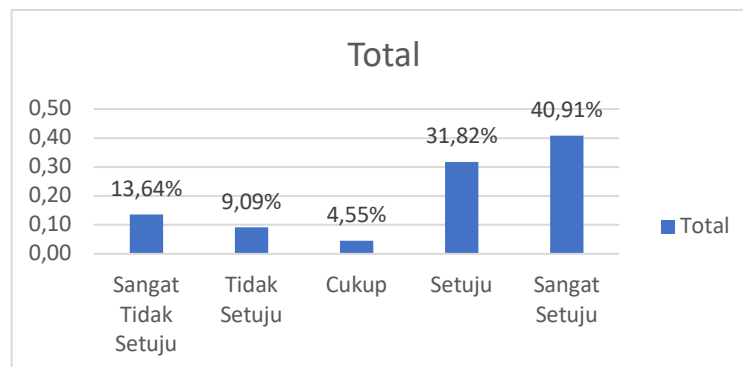
a. Kuesioner Pertanyaan satu



Gambar 14. Koesioner Pertanyaan satu

Pada Gambar 14. terdapat pertanyaan “Apakah semua fitur yang ada seperti fitur tukar sampah dan jual pupuk kompos berfungsi dengan baik?”. Dan dari pertanyaan tersebut terdapat 22 responden di mana 9,1% menjawab sangat kurang setuju, 4,5% tidak setuju, 4,5% cukup, 40,9% setuju, 40,9% sangat setuju.

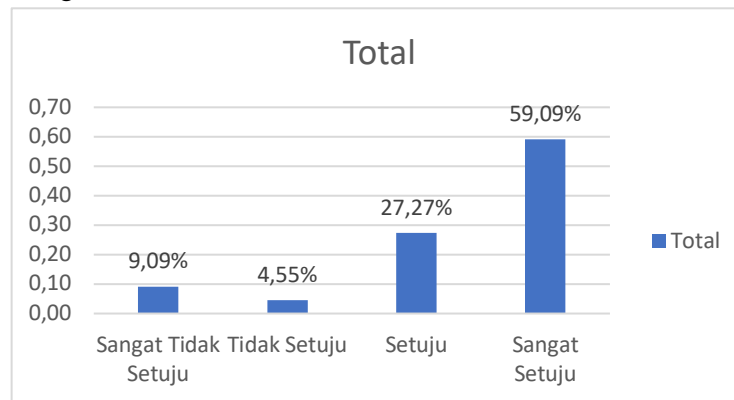
b. Kuesioner Pertanyaan Dua



Gambar 15. Koesioner Pertanyaan Dua

Pada Gambar 15. terdapat pertanyaan “Apakah video pengolahan sampah berjalan dengan baik?”. Dan dari pertanyaan tersebut terdapat 22 responden di mana 13,6% menjawab sangat kurang setuju, 9,1% tidak setuju, 4,5% cukup, 31,8% setuju, 40,9% sangat setuju.

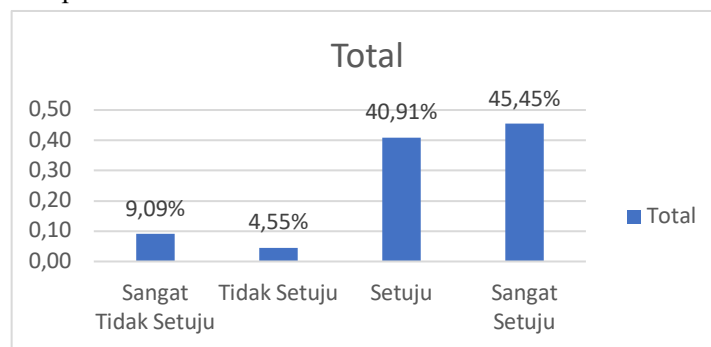
c. Kuesioner Pertanyaan Tiga



Gambar 16. Koesioner Pertanyaan Tiga

Pada Gambar 16. terdapat pertanyaan “Apakah sistem ini mudah digunakan?”. Dan dari pertanyaan tersebut terdapat 22 responden di mana 9,1% menjawab sangat kurang setuju, 4,5% tidak setuju, 0,0% cukup, 27,3% setuju, 59,9 sangat setuju.

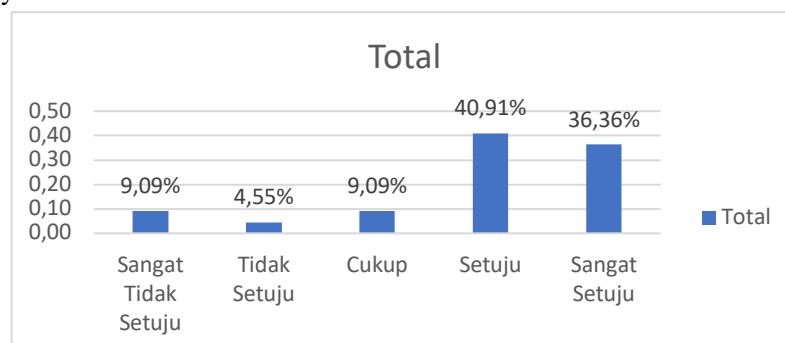
d. Kuesioner Pertanyaan Empat



Gambar 17. Koesioner Pertanyaan Empat

Pada Gambar 17. terdapat pertanyaan “Apakah fitur hapus data pada admin berfungsi dengan baik?”. Dan dari pertanyaan tersebut terdapat 22 responden di mana 9,1% menjawab sangat kurang setuju, 4,5% tidak setuju, 0,0% cukup, 40,9% setuju, 45,5% sangat setuju.

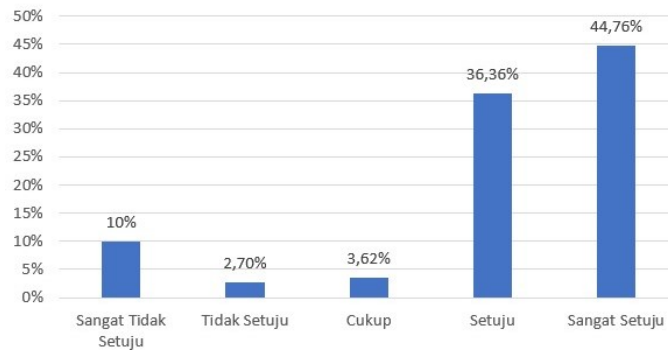
e. Kuesioner Pertanyaan Lima



Gambar 18. Koesioner Pertanyaan Lima

Pada Gambar 5.22 terdapat pertanyaan “Apakah konten dari sistem ini sudah memenuhi kebutuhan pengguna?”. Dan dari pertanyaan tersebut terdapat 22 responden di mana 9,1% menjawab sangat kurang setuju, 4,5% tidak setuju, 9,1% cukup, 40,9% setuju, 36,4% sangat setuju.

4.1.4. Nilai Rata – Rata Kuesioner



Gambar 19. Nilai Rata – Rata Koesioner

Pada Gambar 19 terdapat nilai rata-rata dari ke-5 kuesioner yaitu rata – rata 10% menjawab sangat kurang setuju, 2.7% tidak setuju, 3.62% cukup, 36.36% setuju, dan 47.32% menjawab sangat setuju

4.1.5. Uji Coba Sistem

Tahapan berikutnya yaitu uji coba sistem yang dilakukan oleh 5 orang pengguna yang sudah di wawancarai langsung secara *offline*. Di mana para pengguna tersebut memiliki karakteristik yang berbeda-beda seperti usia dan jenis kelamin. Adapun dokumentasi saat wawancara berlangsung.

a. Wawancara dan Ujicoba dengan Pembeli Sampah 1



Gambar 20. Wawancara dan uji coba dengan pembeli sampah 1

Pada Gambar 20. merupakan proses wawancara dengan ibu-ibu mengenai *website* SARANG.

b. Wawancara dan ujicoba dengan penjual sampah 1



Gambar 21. Wawancara dan uji coba dengan penjual sampah

Pada Gambar 21 merupakan proses wawancara dengan penjual sampah dan proses ujicoba *website* SARANG (Sampah Terang).

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil pembuatan *website* SARANG (Sampah Terang) Dapat disimpulkan :

- a. *Website* SARANG (Sampah Terang) dapat menjadi solusi yang efektif jika diterapkan dalam masyarakat. Melalui fitur sistem tukar sampah dan penjualan pupuk, masyarakat dapat dimotivasi untuk mengurangi dampak negatif sampah terhadap lingkungan. Hasil dari pengujian aplikasi ini dapat diukur dengan melihat respons dari uji coba terhadap 22 orang responden. Berdasarkan uji coba tersebut, ditemukan nilai rata-rata dari kelima kuesioner yang didapatkan, yaitu 10% menjawab sangat kurang setuju, 2.7% tidak setuju, 3.62% cukup setuju, 36.36% setuju, dan 47.32% menjawab sangat setuju. Angka-angka ini memberikan indikasi bahwa konsep SARANG telah diterima secara positif oleh mayoritas responden, yang menggambarkan aplikasi SARANG dapat berjalan baik sesuai kebutuhan masyarakat.
- b. Untuk mengoptimalkan penggunaan *website* SARANG (Sampah Terang) di masa mendatang, perlu dilakukan sejumlah perbaikan. Salah satunya adalah dengan menambahkan fitur kemudahan pembayaran untuk mempermudah proses transaksi. Selain itu, sangat penting untuk melakukan pengujian secara berkala guna mengidentifikasi dan mengatasi *bug* atau masalah lain sebelum mereka menjadi permasalahan yang lebih besar. Dengan demikian, kita dapat memastikan bahwa SARANG terus berkembang dan memberikan manfaat yang lebih besar bagi masyarakat dan lingkungan.

5.2. Saran

Website yang dibangun belum memiliki mekanisme pembayaran terintegrasi, sehingga jika ada transaksi jual beli sampah daur ulang, masih menggunakan metode transfer antar Bank atau dilakukan secara cash.

DAFTAR PUSTAKA

- [1] I. T. Kurniawan, Humaira, and F. Rozi, "REST API Menggunakan NodeJS pada Aplikasi Transaksi Jasa Elektronik Berbasis Android," *Jurnal Ilmiah Teknologi Sistem Informasi*, Desember. 2020, doi: 10.30630/jitsi.1.4.18.
- [2] A. Mulyo, "Simbol Entity Relationship Diagram (ERD) No Simbol Nama Keterangan Jenis entitas dapat berupa suatu elemen lingkungan , Entitas sumber daya atau transaksi yang dipergunakan aplikasi program Hubungan atau Relasi Menunjukkan nama relasi antar satu entitas," 2018.
- [3] Rozi A. Z & Community Smit Dev, *Bootstrap Design Framework*. Jakarta: PT Elex Media Komputindo, 2015.
- [4] U. G. Salamah, "Tutorial Visual Studio Code", *Media Sains Indonesia*, 2021.
- [5] Mauko C, I , Setiohardjo Mardanus Nicodemus, Noach Paulus Frederik, "PENGEMBANGAN WEBSITE UNIT PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT DAN PENERAPAN JURNAL ELEKTRONIK BERBASIS OPEN SOURCE DI POLITEKNIK NEGERI KUPANG", *Jurnal Ilmiah FLASH*, vol 3, Desember 2017.
- [6] Juan K & Budi S, "Pengembangan Menu Digital Menggunakan ReactJs Implementasi Hasil Belajar Studi Independen di Frontend Engineering Program Ruangguru CAMP (Career Acceleration Bootcamp)", *Jurnal Strategi*, Vol 5, No. 1 Mei 2023
- [7] Farozi M, "DESAIN BASIS DATA NON RELASIONAL NOSQL MONGODB PADA WEBSITE SISTEM INFORMASI AKADEMIK", *Jurnal Sistem Informasi Komputer dan Teknologi Informasi (SISKOMTI)*, Vol 1, No 1, 2019.
- [8] Pradigi H. C, Harlina T, Solehatin, "IMPLEMENTASI EXPRESS JS UNTUK MEMBANGUN REST API WEBSITE STIKOM PGRI BANYUWANGP", *Jurnal Informatika dan Komputer*.
- [9] Slamet J, Anistiyasari Y, "Pengembangan NiceRoom Sebagai Online Integrated Development Environment Tools untuk Meningkatkan Hasil Belajar Siswa pada Pembelajaran Python Programming di SMKN 1 Cerme", *Jurnal IT-EDU*, Vol 6 Nomor 2 Tahun 2021, 128-138.
- [10] Pasande P, Tari E, "Daur Ulang Sampah di Desa Paisbuloli Sulawesi Tenggara ", *Jurnal Pengabdian Kepada Masyarakat*, Vol. 5, No. 1 Februari 2021, Hal. 147-153.
- [11] Eni Setyowati. 2014. *Sampah Aktualisasi Nilai-Nilai Islam, Ekonomi Sosial dan Budaya*. Yogyakarta : IAIN Tulungagung Press. Hal: 36.