

Jurnal Informatika Terpadu

Volume 9 No. 2, September 2023



Jurnal Informatika Terpadu

Jurnal Informatika Terpadu memuat jurnal ilmiah di bidang Ilmu Komputer, Sistem Informasi dan Teknik Informatika. Jurnal Teknologi Terpadu diterbitkan oleh LPPM STT Terpadu Nurul Fikri dengan periode dua kali dalam setahun, yakni pada bulan Maret dan September. Jurnal Informatika Terpadu Telah terakreditasi nasional Sinta 5 sesuai dengan SK Nomor 105/E/KPT/2022 yang dikeluarkan oleh Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi.

Ketua Penyunting (*Editor-in-chief*)

Sirojul Munir, S.Si., M.Kom.
Teknik Informatika
STT Terpadu Nurul Fikri

Dewan Penyunting (*Editorial Board Member*)

Hilmy Abidzar Tawakal, S.T., M.Kom.
Teknik Informatika
STT Terpadu Nurul Fikri

Zaki Imaduddin, S.T., M.Kom.
Teknik Informatika
STT Terpadu Nurul Fikri

Tiffany Nabarian, S.Kom., M.T.I.
Teknik Informatika
STT Terpadu Nurul Fikri

Suhendi, S.T., M.MSI.
Sistem Informasi
STT Terpadu Nurul Fikri

Ahmad Rio Ardiansyah, S.Si., M.Si.
Teknik Informatika
STT Terpadu Nurul Fikri

Mitra Bestari (*Reviewer*)

Ahmad Jurnaidi Wahidin, M.Kom
Teknologi Informasi
Universitas Bina Sarana Informatika

Muchlis, M.Kom.
Teknik Informatika
STMIK Antar Bangsa

Krisna Panji, S.Kom., M.M.
Bisnis Digital
STT Terpadu Nurul Fikri

Tri Mukhlison Anugrah, S.E., M.M.
Bisnis Digital
STT Terpadu Nurul Fikri

Alwendi, S.Kom, M.Kom
Ilmu Komputer
Universitas Graha Nusantara

Firdha Aprilyani, S.Kom., M.Kom.
Sistem Informasi
STMIK Antar Bangsa

Ali Khumaidi, M.Kom
Teknik Informatika
Universitas Krisnadwipayana

Wiwit Supriyanti, M.Kom.
Sistem Informasi
Politeknik Indonusa Surakarta

F Rachmat Kautsar, S.Tp., M.E.
Bisnis Digital
STT Terpadu Nurul Fikri

Chairun Nas, S.Kom., M.Kom
Manajemen Informatika
Universitas Insan Cendekia

Agus Prasetyo M, S.E., M.M.
Bisnis Digital
STT Terpadu Nurul Fikri

Davied Wahyu Wismanindra, S.Kom.,
M.M.
Bisnis Digital
STT Terpadu Nurul Fikri

Betty Amalia, S.E., M.M.
Bisnis Digital
STT Terpadu Nurul Fikri

Dr. Hendra Cipta, S.Pd.I., M.Si
Matematika
Universitas Islam Negeri Sumatera
Utara

Penyunting Pelaksana (*Assistant Editors*)

Muh Syaiful Romadhon, S.Kom.
STT Terpadu Nurul Fikri

Miftahussa'adah Putri Siddiq
STT Terpadu Nurul Fikri

Hilmia Zahra, S.T.
STT Terpadu Nurul Fikri

Jurnal Informatika Terpadu telah ter indeks oleh Google Scholar, Index Copernicus International, SINTA, dan Garuda. Tanggung jawab isi artikel berada di penulis bukan pada penerbit atau editor.

Diterbitkan oleh:

LPPM STT Terpadu Nurul Fikri

Alamat Redaksi dan Distribusi:

Kampus B STT Terpadu Nurul Fikri lantai 3
Jl. Lenteng Agung Raya 20, Jakarta Selatan, DKI Jakarta, 12640
Telp. 021 – 786 3191
Email: lppm@nurulfikri.ac.id

Website: <https://journal.nurulfikri.ac.id/index.php/jit/> dan lppm.nurulfikri.ac.id

Jurnal Informatika Terpadu
Volume 9 No. 2, September 2023

Daftar Isi

Pengembangan Sistem Informasi Manajemen Keuangan Sekolah berbasis Web menggunakan <i>Whatsapp Gateway</i>	74
Nur Abidin, Anwar Aini, Muhammad Izzuddin	
UMMIBOT sebagai Media Layanan Informasi Penerimaan Mahasiswa Baru Universitas Muhammadiyah Sukabumi	82
M. Rizky Suherlan, Asriyanik, Agung Pambudi	
Pengembangan <i>REST API</i> dengan menggunakan <i>Express JS</i> untuk Mencari Mentor Pribadi	92
Nasrul, Ahmad Izhar	
Penyusunan Dokumentasi Sistem pada Perancangan Web <i>Backoffice</i> Aplikasi <i>Mobile Telemedicine</i>	103
Laily Maghfira, Esi Putri Silmina	
Rancang Bangun Sistem Informasi <i>Back Office</i> Aplikasi <i>Telemedicine</i> sebagai Media Pembelajaran	119
Anggit Widcaksono, Esi Putri Silmina	
Analisis Kriptografi Algoritma <i>Blowfish</i> pada Keamanan Data menggunakan <i>Dart</i>	126
Yoga Pratama, Tata Sutabri	
Analisa Kinerja Aplikasi Digital Forensik <i>Autopsy</i> untuk Pengembalian Data menggunakan Metode <i>NIST SP 800-86</i>	136
Dedek Julian, Tata Sutabri	
Perancangan Alat Tekanan Darah Otomatis secara Berkala menggunakan Sensor <i>MPX5050DP</i> untuk Pasien Hipertensi	143
Afdhal Kurniawan, Karli Eka Setiawan	
Pengelompokan Rumah Sakit di Jakarta Menggunakan Model <i>DBSCAN</i>, <i>Gaussian Mixture</i>, dan <i>Hierarchical Clustering</i>	149
Karli Eka Setiawan, Afdhal Kurniawan	
Pengembangan Prototipe Sistem S.K.P.I. untuk Pendamping Ijazah Berbasis <i>Google Suite</i>	157
Nuraini Febrianti, Hasna Alaurrahman, Amalia Rahmah	



PENGEMBANGAN SISTEM INFORMASI MANAJEMEN KEUANGAN SEKOLAH BERBASIS WEB MENGGUNAKAN WHATSAPP GATEWAY

Nur Abidin¹, Anwar Aini², Muhammad Izzuddin³

^{1,2,3}Teknik Informatika, Universitas Qomaruddin
Gresik, Jawa Timur, Indonesia 61152

nurabidin@uqgresik.ac.id, manwaraini@uqgresik.ac.id, 121811046@mhs.uqgresik.ac.id

Abstract

The rapid development of information and communication technology supported by adequate and competent human resources is expected to create an information technology capable of solving school administration problems. The school financial management system at Mamba'ul Ihsan Banyuurip Vocational School still uses a recording system on payment cards and then manually recapitulates payment data in books. This causes the process to take quite a long time and has a risk of data loss or damage. This study aims to develop an information system to facilitate school finance parties in transactions and managing school financial data. In addition, the information system created can provide notifications to student guardians for notifications. System design is based on object-oriented programming (OOP) using UML (Unified Modeling Language). The information system development method uses the waterfall method. The development of the system uses the Codeigniter 4 Framework. In the case of automatic Whatsapp sending to student guardians, Node.JS is used as the Whatsapp API Gateway. Development of a school financial management information system with notifications using the Whatsapp API (application programming interface) that has been made for use by the school and is working as planned with the suitability of the effectiveness and efficiency of completing work by 100% and ease of operation by 96%.

Keywords: CodeIgniter, Finance School, Node.JS, Whatsapp API, Whatsapp Gateway

Abstrak

Perkembangan teknologi informasi dan komunikasi yang begitu pesat dan didukung oleh sumber daya manusia yang memadai dan kompeten diharapkan dapat menciptakan sebuah teknologi informasi yang mampu menyelesaikan permasalahan administrasi sekolah. Sistem pengelolaan keuangan sekolah pada SMK Mamba'ul Ihsan Banyuurip saat ini masih menggunakan sistem pencatatan pada kartu pembayaran kemudian data pembayaran direkap secara manual pada buku. Hal ini menyebabkan proses memakan waktu yang cukup lama, serta memiliki risiko kehilangan atau kerusakan data. Penelitian ini bertujuan untuk mengembangkan sistem informasi yang dapat memudahkan pihak keuangan sekolah dalam transaksi dan pengelolaan data keuangan sekolah. Selain itu, sistem informasi yang dibuat dapat memberikan notifikasi kepada wali siswa guna pemberitahuan. Perancangan sistem dibuat berdasarkan *object oriented programming* (OOP) menggunakan UML (*Unified Modelling Language*). Metode pengembangan sistem informasinya menggunakan metode *Waterfall*. *Framework Codeigniter 4* digunakan sebagai kerangka pemrograman pengembangan aplikasinya. Pengiriman *Whatsapp* otomatis kepada wali siswa digunakan *Node.JS* sebagai *Whatsapp API Gateway*. Pengembangan sistem informasi manajemen keuangan sekolah dengan notifikasi menggunakan *Whatsapp API (application programming interface)* yang telah dibuat digunakan pihak sekolah, bekerja sesuai yang direncanakan dengan kesesuaian efektivitas dan efisiensi penyelesaian pekerjaan sebesar 100% serta kemudahan dalam pengoperasian sebesar 96%.

Kata kunci: Codeigniter, Keuangan Sekolah, Node.JS, Whatsapp API, Whatsapp Gateway

1. PENDAHULUAN

Pesatnya perkembangan teknologi memicu munculnya berbagai teknologi baru termasuk di bidang teknologi informasi. *Website* merupakan bagian dari revolusi di bidang teknologi informasi internet. *Website* dapat menjadi solusi alternatif untuk pengembangan teknologi informasi

yang lebih efektif pada saat ini. Menurut Idris dan kawan, *website* menjadi penting karena dapat meningkatkan kredibilitas suatu institusi [1]. *Website* dapat digunakan sebagai media promosi bagi institusi. *Website* dapat diakses dimanapun melalui jaringan internet [2].

Manajemen pengolahan data keuangan SMK Mamba'ul Ihsan masih menggunakan sistem manual atau konvensional dan dianggap tidak efisien di masa sekarang ini maupun masa mendatang. Permasalahan yang ada saat ini terkait pengelolaan keuangan di SMK Mamba'ul Ihsan adalah pembayaran keuangan siswa masih dilakukan dengan cara mencatat transaksi pembayaran pada kartu iuran sekolah. Terdapat beberapa jenis pembayaran yang ada, diantaranya; 1. pembayaran her registrasi, 2. pembayaran kunjungan industri, dan 3. pembayaran ujian akhir semester (UAS). Risiko kehilangan data terkait transaksi pembayaran menjadi masalah utama. Hal tersebut bisa disebabkan karena kesalahan siswa maupun tata usaha (TU). Selain itu, pemberitahuan kepada orang tua siswa terkait transaksi pembayaran juga menjadi masalah yang lain. Terdapat banyak wali siswa yang mengadakan pembayaran iuran sekolah yang dilakukan oleh siswa tidak benar-benar dilakukan. Wali siswa menjadi korban penipuan yang dilakukan oleh anaknya sendiri. Permasalahan utamanya adalah masih belum adanya pengelolaan keuangan yang menyeluruh, yaitu terkait transaksi uang masuk dan uang keluar serta total saldo keuangan sekolah yang *up to date*. Proses rekapitulasi keuangan membutuhkan waktu yang cukup lama. Pentingnya sebuah sistem informasi yang dapat menyelesaikan permasalahan yang telah dijelaskan di atas, maka peneliti melakukan penelitian tersebut.

Penelitian lain yang serupa terkait dengan sistem informasi web yang menggunakan notifikasi web menggunakan *Whatsapp API* telah banyak dilakukan. Penelitian tersebut menjadi acuan peneliti dalam mengembangkan sistem informasi yang dibutuhkan. Penelitian pertama dilakukan oleh lutfi hakim dan kawan, penelitian ini mengembangkan sistem informasi kas sekolah yang berfungsi untuk mengelola transaksi uang masuk dari pembayaran siswa, kemudian pengeluaran digunakan untuk pembayaran gaji karyawan. Notifikasi *Whatsapp* digunakan untuk notifikasi pembayaran SPP siswa dan pembayaran keuangan gaji karyawan [3]. Metode yang digunakan menggunakan *Rapid Application Development (RAD)* dan bahasa pemrograman PHP dengan *framework* Laravel. *Whatsapp Gateway* yang digunakan untuk mengirim informasi menggunakan *Whatsapp Gateway* berbayar. Penelitian yang kedua dilakukan Irda Silvia Ningsih dan kawan [4]. Penelitian tersebut menghasilkan sistem informasi untuk pembayaran SPP siswa. Metode pengembangannya menggunakan *system development life cycle (SDLC)*. Bahasa pemrograman yang digunakan menggunakan PHP *Native*. Penelitian yang ketiga dilakukan oleh Dian Utama dan kawan [5]. Penelitian tersebut menghasilkan sistem informasi manajemen pembayaran uang kuliah berbasis *whatsapp API*. Metode yang digunakan sebagai pengembangan aplikasi adalah *Waterfall*. Bahasa pemrogramannya menggunakan bahasa pemrograman PHP. *Whatsapp Gateway* yang digunakan menggunakan *Node.JS*. Penelitian terdahulu yang diajukan sebagai referensi kemudian dibandingkan dengan penelitian yang dilakukan

saat ini. Penelitian yang dilakukan saat ini yaitu menghasilkan sistem informasi pengelolaan keuangan sekolah berbasis web menggunakan *framework Codeigniter* dan *Node.JS* sebagai *Whatsapp Gateway*. Transaksi keuangan yang masuk tidak hanya dihasilkan dari pembayaran iuran siswa saja, melainkan diperoleh dari keuangan yang didapatkan dari dana operasional sekolah oleh dinas pendidikan dan bantuan pemerintah yang lain. Transaksi uang keluar digunakan untuk mencatat seluruh pengeluaran sekolah meliputi; gaji guru dan kebutuhan-kebutuhan sekolah yang lain. Sehingga, laporan keuangan sekolah menampilkan jumlah keuangan yang masuk dan keluar. Saldo akhir bisa langsung dilihat oleh bendahara sekolah.

Sistem Informasi

Sistem informasi merupakan sekumpulan komponen baik *software* dan *hardware* yang berfungsi untuk menghimpun, menyimpan dan mengelola data menjadi sebuah informasi yang berguna bagi para pengguna [6]. Informasi menjadi sangat dibutuhkan jika dalam kondisi yang memang dibutuhkan. Sehingga ketepatan waktu dalam pengambilan keputusan bisa dapat segera diambil [7].

Codeigniter

Codeigniter adalah sebuah kerangka kerja pengembangan aplikasi yang digunakan untuk mengembangkan sistem informasi berbasis web dengan lebih cepat dan mudah [3]. *Codeigniter* didukung oleh sebuah dokumentasi yang jelas dan lengkap [8]. *Codeigniter* lebih baik digunakan ketika digunakan untuk menyelesaikan proyek pengembangan sistem informasi dalam kerja tim [9].

Manajemen Keuangan Sekolah

Manajemen keuangan sekolah merupakan kegiatan yang dilakukan oleh pihak sekolah dalam hal menghimpun, mengelola serta bertanggungjawabkan keuangan yang berasal peserta didik dan atau dari sumber eksternal lain [10]. Pengelolaan keuangan sekolah meliputi kas penerimaan dan kas pengeluaran [3].

Whatsapp API

Application Programming Interface Whatsapp atau disingkat *API* adalah sebuah pustaka atau *library* nya *Whatsapp*. *Whatsapp API* memungkinkan pengguna bisa menggunakan fitur pengiriman pesan seperti halnya mengirim pesan menggunakan aplikasi *Whatsapp* di dalam aplikasi pihak kedua [11]. Aplikasi pihak kedua disebut sebagai *Whatsapp Gateway* [12].

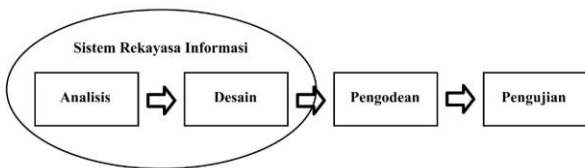
Node.JS

Menurut Wibowo dan Wiguna [13], *Node.JS* adalah perangkat lunak yang dapat digunakan untuk mengembangkan aplikasi berbasis web yang berjalan di sisi server. *Node.JS* menggunakan *Javascript* sebagai bahasa pemrogramannya. *Node.JS* bersifat asinkronis, artinya

Node.JS dapat berjalan normal, meskipun kode program yang lain juga berjalan bersamaan [14].

2. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini secara umum menggunakan metode *Waterfall*. Metode pengembangan sistem informasi menggunakan metode *Waterfall* masih populer digunakan hingga saat ini [2]. Meskipun dianggap klasik, akan tetapi metode *Waterfall* banyak dipakai dalam pengembangan perangkat lunak karena pendekatan penyelesaiannya dilakukan secara sistematis dan berurutan[15]. Metode *Waterfall* memiliki beberapa tahapan yaitu analisis, desain, pengkodean, pengujian. Seperti pada gambar 1.



Gambar 1. Ilustrasi Model *Waterfall*

Analisis kebutuhan sistem dilakukan secara mendalam untuk melihat lebih detail kebutuhan perangkat lunak agar dapat dipahami seperti yang dibutuhkan oleh pengguna. Desain merupakan proses yang dilakukan untuk pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Pengkodean adalah tahap dimana pembuatan aplikasi yang dilanjutkan dari tahap desain yang sudah dibuat, dalam hal ini Bahasa pemrograman yang digunakan adalah *PHP* dan menggunakan *Framework Codeigniter 4*. Pengujian pada perangkat lunak secara fungsional dan memastikan bahwa semua bagian sudah di uji. Hal ini

dilakukan untuk mengurangi kesalahan (*error*) dan memastikan fungsi yang diuji sesuai dengan yang diinginkan.

3. HASIL DAN PEMBAHASAN

3.1. Hasil Penelitian

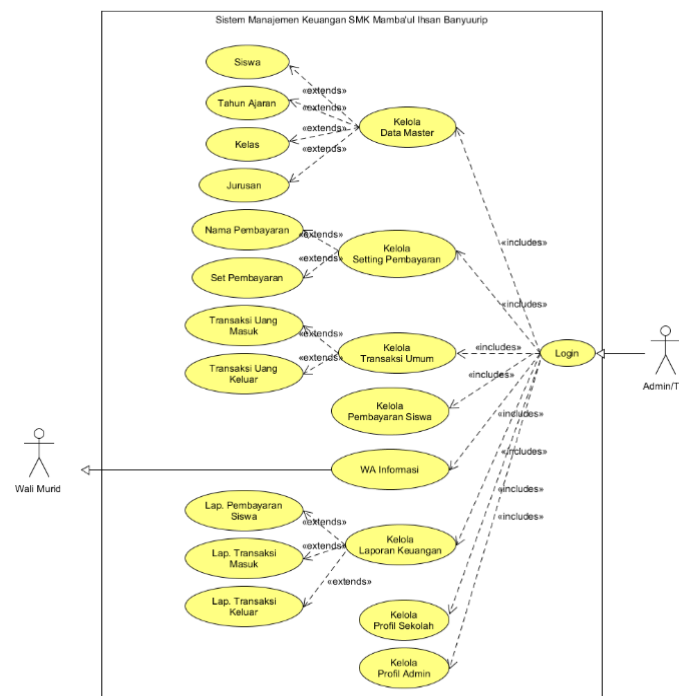
Sesuai dengan metode *Waterfall* yang digunakan pada penelitian ini, Pelaksanaan penelitian untuk tahapan pertama yaitu Analisis. Tahapan analisis dilakukan dengan cara wawancara dengan pihak Staf Tata Usaha bagian Keuangan sekolah. Hasil wawancara tersebut kemudian dilanjutkan dengan tahap Desain. Berikut hasil Desain yang telah disesuaikan dengan kebutuhan sekolah.

3.1.1. Use Case Diagram Sistem Informasi Keuangan Sekolah

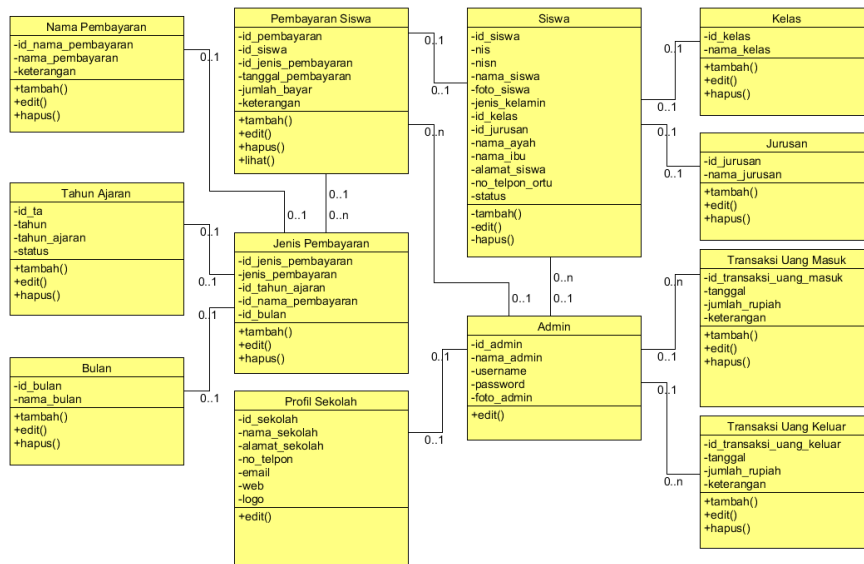
Desain rancangan sistem informasi yang telah dibuat menggunakan diagram UML yang dapat dilihat pada Gambar 2 menggambarkan hubungan aktor terhadap fitur sistem informasi. Terdapat dua aktor, yaitu staf Tata Usaha (TU) bagian keuangan dan Wali Murid. Proses pencatatan transaksi pembayaran keuangan sekolah secara penuh dilakukan oleh TU. Jika pembayaran telah berhasil dilakukan, maka Wali Murid akan mendapatkan notifikasi transaksi melalui aplikasi *Whatsapp*.

3.1.2. Class Diagram Sistem Informasi Keuangan Sekolah

Transaksi yang dilakukan pada sistem informasi keuangan ini akan disimpan dalam sebuah *Database*. Rancangan *Database* pada penelitian ini digambar menggunakan *Class Diagram*. *Class Diagram* dapat dilihat pada Gambar 3. Setelah tahapan Desain, kemudian dilanjutkan dengan tahapan pengkodean dan pengujian.



Gambar 2. Use Case Diagram sistem informasi keuangan



Gambar 3. Class Diagram sistem informasi keuangan

3.2. Implementasi Sistem

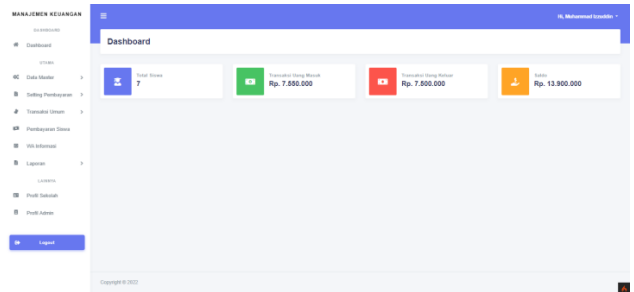
Implementasi sistem merupakan tahapan pengkodean dan pengujian. Sistem informasi manajemen keuangan sekolah yang dihasilkan kemudian diimplementasikan ke dalam bentuk aplikasi berbasis web. Implementasi *user interface* dari aplikasi sistem manajemen keuangan sekolah berbasis web sebagai berikut:

3.2.1. Form Login

Pada Gambar 4 *Form login* ini pengguna diharuskan untuk memasukkan *username* dan *password* yang selanjutnya akan dicek kebenaran oleh sistem untuk masuk ke halaman *dashboard* aplikasi.

3.2.2. Dashboard

Menu dashboard dapat dilihat pada Gambar 5, *Dashboard* menampilkan beberapa informasi seperti total siswa, total uang yang masuk dan total uang yang keluar.



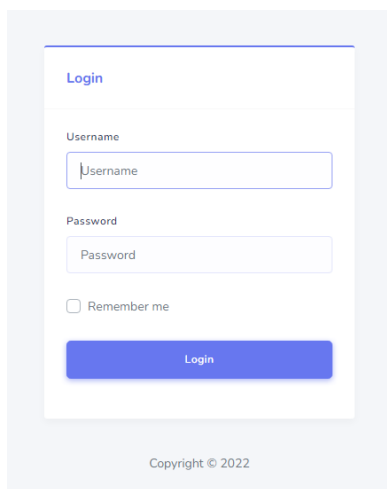
Gambar 5. Halaman Dashboard

3.2.3. Data Master

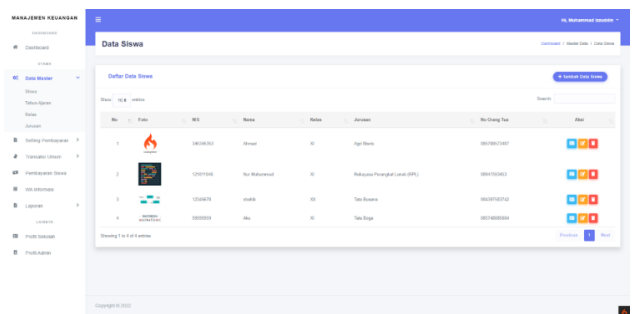
Menu ini berfungsi untuk memasukkan data siswa, data tahun ajaran, data kelas, dan data jurusan.

3.2.4. Data Siswa

Menu data siswa ini menampilkan semua data siswa dan juga di *form* data siswa ini adalah tempat untuk menambahkan, mengedit, dan menghapus data siswa. Halaman data siswa dapat dilihat pada Gambar 6.



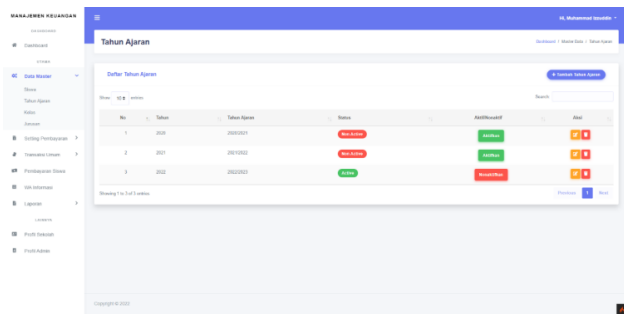
Gambar 4. Form Login



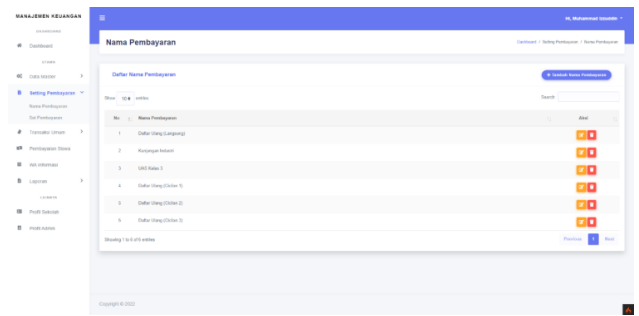
Gambar 6. Halaman Data Siswa

3.2.5. Tahun Ajaran

Pada Gambar 7 Menu tahun ajaran ini menampilkan seluruh data tahun ajaran yang sudah diinputkan admin, di menu ini juga admin bisa *input*, edit, dan hapus data tahun ajaran.



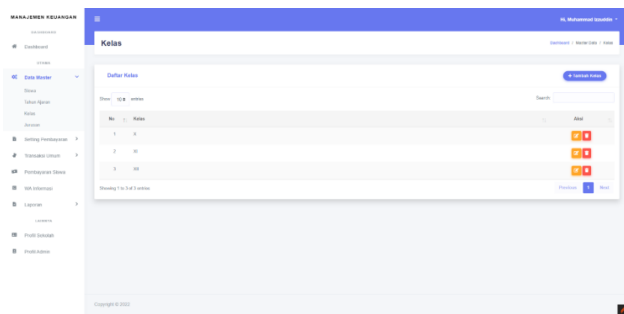
Gambar 7. Halaman Tahun Ajaran



Gambar 10. Halaman Nama Pembayaran

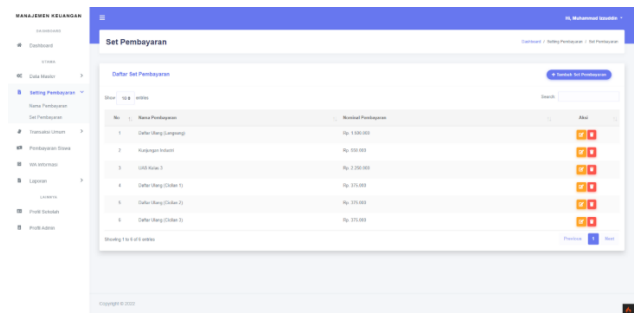
3.2.6. Kelas

Menu kelas ini menampilkan seluruh data kelas yang sudah dimasukkan admin, di menu ini dapat dilihat pada Gambar 8 admin juga bisa *input*, edit, dan hapus data kelas.



Gambar 8. Halaman Kelas

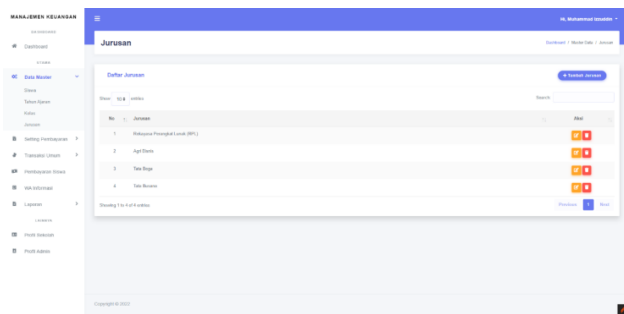
Sedangkan menu set pembayaran ini menampilkan seluruh set pembayaran yang sudah dimasukkan oleh admin melalui *form input*, di menu ini juga admin bisa *input*, edit dan hapus set pembayaran.



Gambar 11. Halaman Set Pembayaran

3.2.7. Jurusan

Menu jurusan pada Gambar 9 ini menampilkan seluruh data jurusan yang sudah dimasukkan admin, di menu ini juga admin bisa *input*, edit, dan hapus data jurusan.

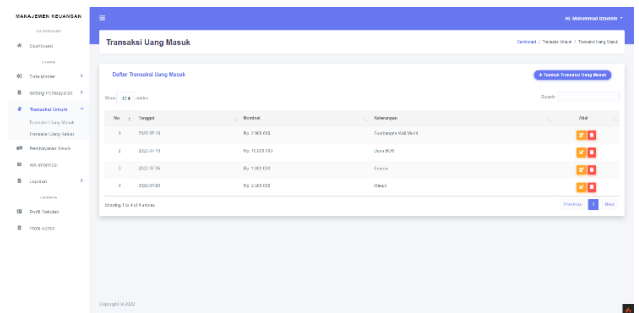


Gambar 9. Halaman Jurusan

3.2.9. Transaksi Umum

Pada menu ini berfungsi sebagai transaksi uang masuk dan transaksi uang keluar dan di dalam menu ini terdapat dua menu antara lain:

- 1) Transaksi uang masuk pada Gambar 12 berfungsi untuk pencatatan uang yang masuk ke sekolah seperti dana bos, dan lain-lain.



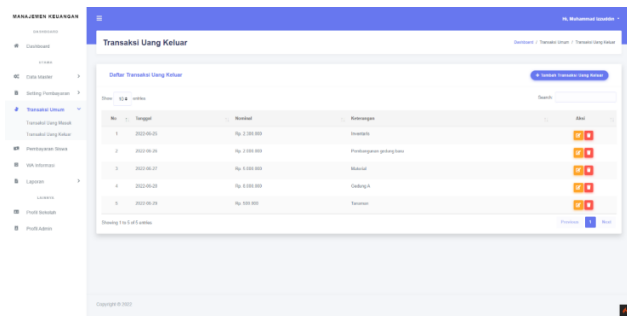
Gambar 12. Halaman Transaksi Uang Masuk

3.2.8. Setting Pembayaran

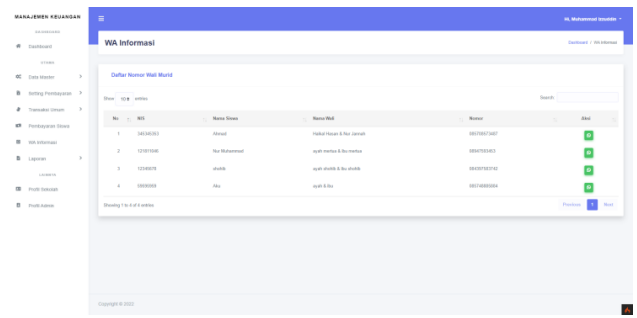
Menu ini dapat dilihat pada Gambar 10 berfungsi untuk mengelola *setting* pembayaran seperti nama pembayaran dan set pembayaran yang akan digunakan pada pembayaran siswa, di dalam menu ini terdapat dua menu antara lain; Nama Pembayaran dapat dilihat pada Gambar 10 dan Set Pembayaran pada Gambar 11.

Menu nama pembayaran ini menampilkan seluruh data nama pembayaran yang sudah dimasukkan oleh admin melalui *form input*, di menu ini juga admin bisa *input*, edit, dan hapus nama pembayaran.

- 2) Transaksi uang keluar pada Gambar 13 berfungsi untuk pencatatan uang yang keluar dari sekolah seperti pembelian inventaris sekolah, dan lain-lain.



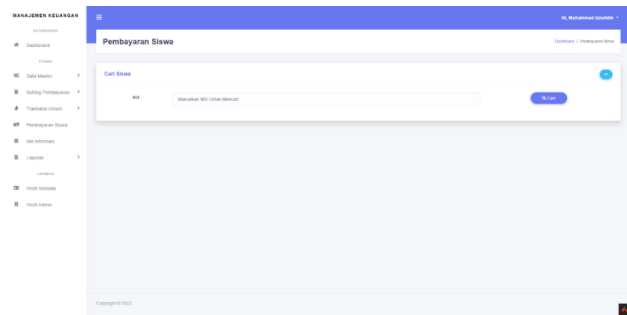
Gambar 13. Halaman Transaksi Uang Keluar



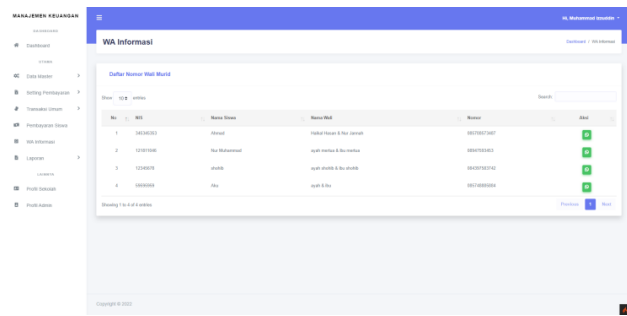
Gambar 15. Halaman WA Informasi

3.2.10. Pembayaran Siswa

Menu pembayaran siswa ini adalah menu utama dalam aplikasi ini yaitu menampilkan transaksi setiap siswa yang sudah terdaftar pada *form* data siswa, di menu pembayaran siswa ini juga tempat transaksi setiap siswa dan *send whatsapp* transaksi kepada nomor wali murid. Menu pembayaran siswa dan *send whatsapp* dapat dilihat pada Gambar 14 (a) Pembayaran Siswa dan (b) *send whatsapp*.



Gambar 14 (a). Pembayaran Siswa



Gambar 14 (b). Send Whatsapp

3.2.11. WA Informasi

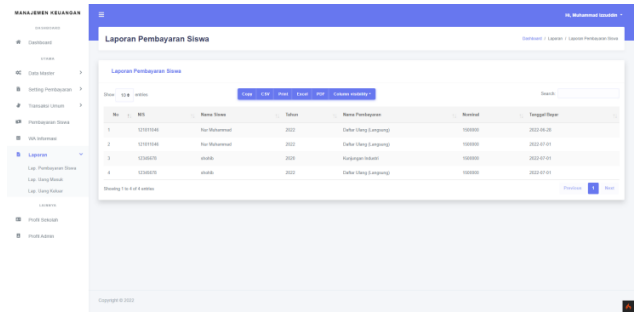
Menu pada Gambar 15 ini berfungsi untuk mengirimkan sebuah informasi mengenai pembayaran di sekolah SMK Mamba'ul Ihsan ini dengan mencari nomor yang ingin diinformasikan melalui *Whatsapp API*.

3.2.12. Laporan

Menu ini berfungsi untuk mencetak seluruh laporan yang ada di sistem ini diantaranya ada:

1) Laporan Pembayaran Siswa

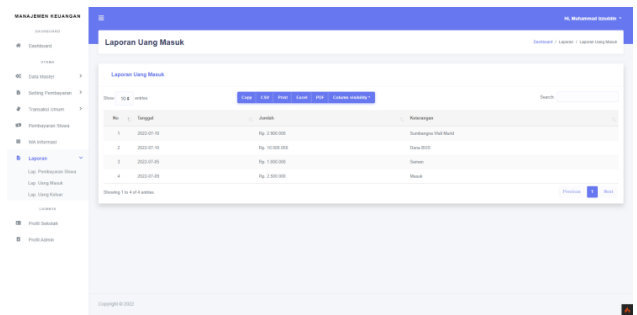
Laporan pembayaran siswa ini menampilkan seluruh data pembayaran siswa yang sudah melakukan transaksi di menu pembayaran siswa, laporan ini bisa di cetak maupun di *export*. Dapat dilihat pada Gambar 16.



Gambar 16. Halaman Laporan Pembayaran Siswa

2) Laporan Uang Masuk

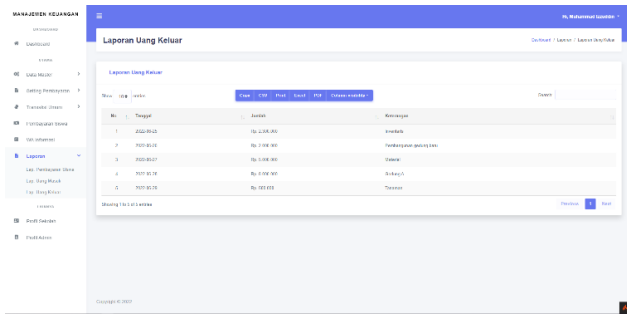
Laporan uang masuk pada Gambar 17 ini menampilkan seluruh data transaksi uang masuk yang sudah dilakukan di menu transaksi uang masuk, laporan ini bisa dicetak maupun di-*export*.



Gambar 17. Halaman Laporan Uang Masuk

3) Laporan Uang Keluar

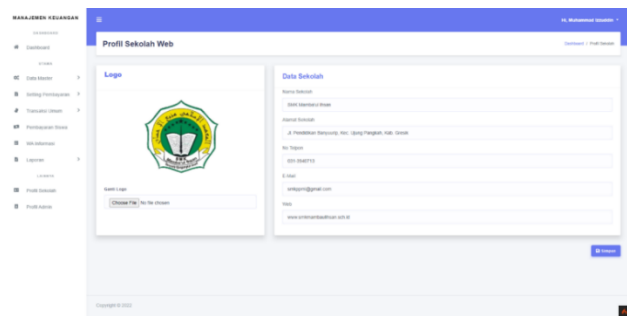
Laporan uang keluar pada Gambar 18 ini menampilkan seluruh data transaksi uang keluar yang sudah dilakukan di menu transaksi uang keluar, laporan ini bisa dicetak maupun di-*export*.



Gambar 18. Halaman Laporan Uang Keluar

3.2.13. Profil Sekolah

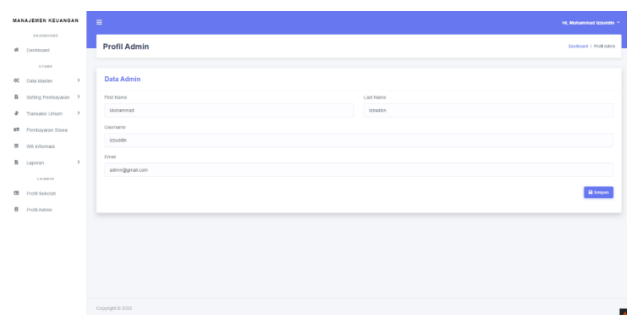
Menu profil sekolah pada Gambar 19 digunakan untuk mengedit informasi sekolah yang akan ditampilkan pada beberapa footer yang ada di sistem ini.



Gambar 19. Halaman Profil Sekolah

3.2.14. Profil Admin

Menu profil admin ini untuk mengedit informasi admin yang digunakan untuk masuk ke sistem manajemen keuangan ini. Menu profil admin dapat dilihat pada Gambar 20.



Gambar 20. Halaman Profil Admin

3.3. Pengujian Sistem

Pengujian dengan pihak staf TU Keuangan berdasarkan pengalaman penggunaan sistem informasi serta pengisian kuesioner. Pengisian kuesioner tersebut meliputi efektifitas pekerjaan, kecepatan dalam penyelesaian masalah atau pengambilan keputusan, dan kemudahan dalam pengoperasian. Hasil pengujian dengan menggunakan kuesioner tersebut dapat dilihat pada tabel 1 berikut.

Tabel 1. Hasil Kuisisioner Pengujian Sistem Informasi

No	Pertanyaan	Hasil %
Efektifitas		
1	Penggunaan sistem informasi berdampak terhadap keefektifan pekerjaan	100%
2	Sistem informasi meningkatkan kualitas pengambilan keputusan	
Efisiensi		
3	Adanya sistem informasi akan membantu mempercepat dalam menyelesaikan pekerjaan	100%
4	Sistem informasi bermanfaat dalam pekerjaan saya	
5	Saya menggunakan sistem informasi untuk memudahkan dalam pekerjaan	
Kemudahan		
6	Sistem informasi tersebut jelas dan dapat dimengerti dalam pengoperasiannya	96%
7	Penggunaan sistem informasi mudah dipahami	
8	Sistem informasi mudah digunakan	
9	Kemudahan sistem informasi cepat untuk disesuaikan	
10	Sistem informasi memudahkan saya menjadi terampil	

Secara keseluruhan hasil yang didapatkan sangat sesuai. Terkait dengan kemudahan, pengguna perlu untuk lebih banyak menggunakan sistem informasi. Sehingga lebih memahami alur kerja pengoperasiannya.

4. KESIMPULAN

Berdasarkan hasil pengujian, Sistem Informasi manajemen keuangan berbasis web dengan menggunakan *Whatsapp Gateway* ini dapat mempermudah bagian keuangan untuk mengelola data keuangan secara lebih efektif dan efisien. Aplikasi ini juga bisa mengirimkan notifikasi pembayaran siswa kepada wali murid dengan menggunakan *Whatsapp Gateway* tujuannya agar informasi pembayaran bisa diketahui oleh wali murid. Aplikasi ini juga bisa cetak bukti pembayaran dan rekap laporan keuangan sekolah agar bisa mengurangi kesalahan yang terjadi.

Ucapan Terima Kasih

Ucapan Terima Kasih kami ucapkan kepada Lembaga Penelitian dan Pengabdian Kepada Masyarakat (LPPM) Universitas Qomaruddin dan Pihak SMK Mamba'ul Ihsan yang bersedia menjadi tempat penelitian. Tidak lupa teman peneliti dan mahasiswa yang membantu dalam penyelesaian aplikasi.

DAFTAR PUSTAKA

[1] M. A. Novianto and S. Munir, "Jurnal Informatika Terpadu," *J. Inform. Terpadu*, vol. 8, no. 2, pp. 47–61, 2022, [Online]. Available: <https://journal.nurulfikri.ac.id/index.php/JIT>

[2] B. Wisnuadhi, A. Maspupah, S. R. Wulan, M. R. Sholahuddin, S. D. Setiarini, and L. H. Firdaus, "Studi Komparatif Fungsionalitas, Performance dan User experience pada Website Geopark," *J. Inf.*

- Syst. Res.*, vol. 3, no. 4, pp. 579–589, 2022, doi: 10.47065/josh.v3i4.1879.
- [3] L. Hakim, S. P. Kristanto, M. N. Shodiq, and E. Amaliyah, “Aplikasi Penerimaan dan Pengeluaran Kas Berbasis Web dan WhatsApp Gateway,” *J. Tekno Kompak*, vol. 15, no. 1, p. 13, 2021, doi: 10.33365/jtk.v15i1.900.
- [4] I. S. Ningsih, H. Mulyono, and F. Rini, “Sistem Informasi Pembayaran SPP Menggunakan Whatsapp Gateway di SMK Tamansiswa Padang,” *JURTEII J. Teknol. Inf.*, vol. 1, no. 2, pp. 28–32, 2022, doi: 10.22202/jurteii.2022.5715.
- [5] D. Utama, F. Sari, and D. Desyanti, “Aplikasi Manajemen Pembayaran Uang Kuliah Berbasis Whatsapp API Di Sekolah Tinggi Teknologi Dumai,” *JUTEKINF (Jurnal Teknol. Komput. dan Informasi)*, vol. 10, no. 2, pp. 54–61, 2022, doi: 10.52072/jutekinf.v10i2.432.
- [6] N. Abidin and M. Hariyadi, “Evaluasi Tingkat Keamanan Informasi Sebagai Upaya Peningkatan Keamanan Sistem Informasi Akademik Di Sekolah Tinggi Teknik Qomaruddin Gresik Menggunakan Indeks Keamanan Informasi,” *NJCA (Nusantara J. Comput. Its Appl.)*, vol. 5, no. 2, pp. 47–52, 2020.
- [7] S. Julianto and S. Setiawan, “Perancangan Sistem Informasi Pemesanan Tiket Bus Pada Po. Handoyo Berbasis Online,” *Simatupang, Julianto Sianturi, Setiawan*, vol. 3, no. 2, pp. 11–25, 2019, [Online]. Available: <https://journal.amikmahaputra.ac.id/index.php/JIT/article/view/56/48>
- [8] A. F. Sallaby and I. Kanedi, “Perancangan Sistem Informasi Jadwal Dokter Menggunakan Framework Codeigniter,” *J. Media Infotama*, vol. 16, no. 1, pp. 48–53, 2020, doi: 10.37676/jmi.v16i1.1121.
- [9] M. Muqorobin and N. A. Rozaq Rais, “Comparison of PHP Programming Language with Codeigniter Framework in Project CRUD,” *Int. J. Comput. Inf. Syst.*, vol. 3, no. 3, pp. 94–98, 2022, doi: 10.29040/ijcis.v3i3.77.
- [10] R. F. Awaludin, S. Bahri, and M. Muslih, “Penerapan Zachman Framework Dalam Perancangan Sistem Informasi Manajemen Keuangan Sekolah,” *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 6, no. 1, p. 78, 2021, doi: 10.24114/cess.v6i1.20433.
- [11] Abidatul Izzah, “Pengembangan Web Company Profile Terintegrasi Dengan Api Whatsapp (Studi Kasus: Agen Sembako Al-Barkah),” *INFOTECH J.*, pp. 40–44, 2021, doi: 10.31949/infotech.v7i1.1067.
- [12] T. S. Maulidda and S. M. Jaya, “Perancangan Sistem Informasi Berbasis Web Melalui Whatsapp Gateway Studi Kasus Sekolah Luar Biasa-Bc Nurani,” *J. Teknol. Inf. dan Komun.*, vol. 11, no. 1, pp. 38–44, 2021, doi: 10.56244/fiki.v11i1.421.
- [13] A. T. Wibowo and A. S. Wiguna, “Pemanfaatan Teknologi Single Page Application (SPA) dalam Pembuatan Aplikasi Feedback Dosen dari Mahasiswa Sebagai Bentuk Pengawasan Lembaga Terhadap Kinerja Dosen di Bidang Pengajaran,” *SMARTICS J.*, vol. 5, no. 1, pp. 34–43, 2019, doi: 10.21067/smartics.v5i1.3327.
- [14] M. Anshori, A. Widya, and P. Wahono, “Implementasi Sistem Notifikasi Akademik Berbasis WhatsApp Engine di SMK NU Al-Hidayah Ngimbang,” *Akad. Manaj. Inform. Komput.*, vol. 2, no. 1, pp. 1–8, 2022, [Online]. Available: <http://ejournal.amik.ac.id/index.php/computech>
- [15] A. Josi, “Implementasi Metode Waterfall dalam pembangunan Company Profile Website Akademi Komunitas Dharma Bhakti Bangka (AK DBB),” *JUTIM (Jurnal Tek. Inform. Musirawas)*, vol. 5, no. 2, pp. 133–140, 2020, doi: 10.32767/jutim.v5i2.1029.



UMMIBOT SEBAGAI MEDIA LAYANAN INFORMASI PENERIMAAN MAHASISWA BARU UNIVERSITAS MUHAMMADIYAH SUKABUMI

M. Rizky Suherlan¹, Asriyanik², Agung Pambudi³

^{1,2,3}Teknik Informatika, Universitas Muhammadiyah Sukabumi
Sukabumi, Jawa Barat, Indonesia 43113

rizkysuherlan07@gmail.com, asriyanik263@ummi.ac.id, agungpambd@ummi.ac.id

Abstract

The continuous advancement in information and communication technology has overcome geographical limitations and enhanced the speed of information and knowledge transfer. Information and communication technology has been applied in various aspects, including improving services, speeding up communication processes, facilitating access to information, and effectively sharing information. One significant technological innovation is the chatbot, an artificial intelligence technology that enables interactive communication. Implementing a chatbot at the Admissions Office of Muhammadiyah University Sukabumi (UMMI) aims to assist prospective students in obtaining information regarding new student admissions. The main challenge addressed is the limitation of office hours, which often hinders prospective applicants from accessing information quickly. This research employs the Natural Language Processing (NLP) method in chatbot development, intending to enable the chatbot to understand natural language and search for relevant information. The research objective is to create an efficient chatbot to respond to user inquiries outside of office hours. The stages of chatbot development using the NLP method encompass data collection, text processing, feature engineering, modeling, inference, deployment, and evaluation. The research results indicate that the chatbot, named UMMIBOT, successfully achieved a functional accuracy rate of 83.67%, addressing the issue of office hour limitations and providing responses that align with user input with high accuracy.

Keywords: Chatbot, Machine Learning, Natural Language Processing, New Student Admissions, UMMIBOT

Abstrak

Perkembangan teknologi informasi dan komunikasi yang terus maju telah mengatasi batasan jarak dan meningkatkan kecepatan transfer informasi dan pengetahuan. Teknologi informasi dan komunikasi telah diterapkan dalam berbagai aspek, termasuk meningkatkan pelayanan, mempercepat proses komunikasi, memudahkan akses informasi, dan efektif dalam berbagi informasi. Salah satu inovasi teknologi yang penting adalah chatbot, sebuah teknologi kecerdasan buatan yang memungkinkan komunikasi interaktif. Penerapan *chatbot* di PMB Universitas Muhammadiyah Sukabumi bertujuan untuk membantu calon mahasiswa dalam mendapatkan layanan informasi seputar pendaftaran mahasiswa baru. Kendala utama yang diatasi adalah keterbatasan jam kerja yang seringkali menghambat calon pendaftar dalam mendapatkan informasi dengan cepat. Penelitian ini menggunakan metode NLP (*Natural Language Processing*) dalam pengembangan *chatbot*, yang bertujuan untuk memungkinkan *chatbot* memahami bahasa alami dan mencari informasi yang diperlukan. Tujuan penelitian ini adalah menciptakan chatbot yang efektif dalam memberikan respon terhadap pertanyaan pengguna di luar jam kerja. Tahapan pengembangan *chatbot* dengan metode NLP meliputi *data collection*, *text processing*, *feature engineering*, *modeling*, *inference*, *deployment*, dan *evaluation*. Hasil penelitian menunjukkan bahwa *chatbot* yang dinamakan UMMIBOT berhasil mencapai tingkat akurasi fungsionalitas sebesar 83,67%, mengatasi masalah keterbatasan jam kerja, dan memberikan respon yang sesuai dengan input pengguna dengan tingkat akurasi yang tinggi.

Kata kunci: Chatbot, Pembelajaran Mesin, Pemrosesan Bahasa Alami, Penerimaan Mahasiswa Baru, UMMIBOT

1. PENDAHULUAN

Perkembangan pesat dalam Teknologi Informasi dan Komunikasi (TIK) telah mengatasi batasan geografis dan meningkatkan transfer informasi serta pengetahuan secara dramatis. Fenomena ini tercermin dalam adopsi luas oleh

masyarakat terhadap perangkat teknologi tinggi. Pemanfaatan TIK telah meresap ke berbagai sektor, termasuk dalam upaya meningkatkan pelayanan, mempercepat aliran informasi, memudahkan akses terhadap informasi, dan efektif dalam berbagi pengetahuan. Salah

satu tonggak penting dalam evolusi teknologi adalah *chatbot*, sebuah aplikasi kecerdasan buatan yang memungkinkan komunikasi interaktif. Terdapat beberapa pengertian tentang *chatbot*, yaitu teknologi kecerdasan buatan yang memungkinkan komunikasi interaktif [1]. *Chatbot*, yang juga dikenal sebagai *Chatterbot*, adalah suatu program yang memiliki kemampuan untuk memproses *input* yang diberikan oleh pengguna dan menghasilkan respon yang dikirim kembali ke pengguna [2]. *Chatbot* merupakan aplikasi yang berfungsi sebagai wadah percakapan (*chatting*) antara mesin dan manusia [3]. Dengan menerapkan NLP (*Natural language Processing*), *chatbot* memiliki kemampuan untuk memahami informasi dan secara cerdas mengumpulkan data spesifik hanya dengan menggunakan bahasa alami. NLP, yang merupakan bagian dari kecerdasan buatan, merupakan ilmu yang memfasilitasi interaksi antara mesin dan manusia melalui penggunaan bahasa alami. Dengan kata lain, NLP memungkinkan seseorang berinteraksi dengan mesin seolah-olah sedang berkomunikasi dengan manusia [4]. *Natural Language Processing* (NLP) adalah disiplin ilmu dalam bidang kecerdasan buatan yang memfasilitasi interaksi antara mesin dan manusia melalui bahasa alami [5].

Penelitian terdahulu telah mengeksplorasi penggunaan *chatbot* dalam berbagai konteks. Sebagai contoh, dalam sebuah penelitian yang dilakukan oleh Agung Siswanto Bayu Aji dengan judul "Membangun *Chatbot* Layanan *Helpdesk* Perpajakan KPP Pratama Jakarta Setiabudi Satu", *chatbot* ini telah diintegrasikan ke dalam aplikasi WhatsApp dan menggunakan algoritma *exact match* sebagai metode responnya. Algoritma *exact match* beroperasi dengan cara langsung mencocokkan input pengguna dengan pola yang telah ditentukan sebelumnya. Hasil dari penelitian ini menunjukkan bahwa *chatbot* mampu memberikan respon dengan cepat dan akurat ketika pengguna memasukkan pesan yang cocok dengan pola yang telah ada. Namun, jika pengguna memasukkan pesan yang tidak sesuai dengan pola yang ada, *chatbot* mungkin tidak memberikan respons atau merespon dengan pola yang sama berulang kali untuk mengarahkan pengguna agar memasukkan pesan sesuai dengan pola yang diminta [6]. Penelitian lain oleh Bestralaga Rusmarasy mencoba menerapkan *chatbot* dengan judul "Pengembangan *Chatbot* pada CoMa Untuk Memberikan Motivasi Kepada Pengguna Menggunakan AIML", *chatbot* dikembangkan menggunakan AIML (*Artificial Intelligence Markup Language*). AIML bekerja dengan prinsip *template matching*, di mana *chatbot* akan mencocokkan *input* dari pengguna dengan pola yang telah ditentukan sebelumnya. Penelitian ini menggunakan metode pengujian *blackbox* dan *whitebox*, yang menghasilkan tingkat akurasi sebesar 100% dan tingkat kegunaan (*usability*) sebesar 88,8%. Namun, terdapat kekurangan dalam pengembangan *chatbot* menggunakan AIML, yaitu jika terdapat *input* yang tidak sesuai dengan pola yang telah ditentukan, *chatbot* tidak dapat memberikan respon apapun

[2]. Dalam penelitian yang dilakukan oleh Eva Mursidah dengan judul "Implementasi *Chatbot* Layanan Informasi Pendaftaran Mahasiswa Baru Program Pascasarjana Departemen Teknik Informatika", *chatbot* digunakan sebagai media pendukung untuk memberikan layanan informasi pendaftaran program Pascasarjana Departemen Teknik Informatika di ITS. *Chatbot* ini dikembangkan dengan menggunakan metode *natural language processing* (NLP) dan alat bantu dialogflow yang diintegrasikan ke dalam platform WhatsApp. *Chatbot* yang dihasilkan mampu menerima *input* dari pengguna dengan berbagai cara, baik melalui penggunaan angka maupun mengetikkan kata kunci dari topik yang telah tersedia. Selain itu, *chatbot* juga dapat memberikan jawaban *default* jika *input* dari pengguna tidak sesuai dengan template pertanyaan yang telah ditentukan sebelumnya. Hasil pengujian menunjukkan tingkat akurasi sebesar 98,82% dari 85 kalimat pertanyaan terkait pendaftaran program Pascasarjana di ITS [7].

Di Universitas Muhammadiyah Sukabumi, *chatbot* dapat menjadi solusi untuk memberikan layanan informasi terkait penerimaan mahasiswa baru. Dimana terdapat kendala dalam hal keterbatasan waktu dalam memberikan informasi kepada calon mahasiswa baru di luar jam kerja. *Chatbot* yang dilengkapi dengan NLP dapat memberikan respons yang cepat dan akurat terhadap pertanyaan-pertanyaan ini. Dengan mengintegrasikan pengalaman dari penelitian sebelumnya, tujuan penelitian ini adalah untuk mengembangkan *chatbot* yang mampu mengatasi masalah ini dengan memberikan layanan informasi yang lebih baik dan responsif kepada calon mahasiswa baru di Universitas Muhammadiyah Sukabumi. Hasil penelitian ini diharapkan akan memberikan manfaat signifikan dalam memberikan informasi yang tepat waktu dan akurat kepada calon mahasiswa baru di Universitas Muhammadiyah Sukabumi. *Chatbot* ini akan menjadi alat yang efektif dalam memfasilitasi penerimaan mahasiswa baru dengan memberikan respon yang cepat dan relevan terhadap pertanyaan-pertanyaan mereka. Dengan demikian, penelitian ini akan membantu meningkatkan pengalaman calon mahasiswa baru dalam proses penerimaan, mengatasi kendala waktu yang seringkali menjadi hambatan dalam memperoleh informasi yang mereka butuhkan.

2. METODE PENELITIAN

2.1 Tahapan Pengumpulan Data

Untuk mengumpulkan data dalam penelitian ini, digunakan beberapa teknik yang telah dipilih, antara lain:

2.1.1 Studi Pustaka

Dalam penelitian ini, digunakan metode studi pustaka yang melibatkan pencarian, pengumpulan, dan analisis literatur serta penelitian yang telah dilakukan sebelumnya yang berasal dari jurnal, buku dan internet. Metode studi pustaka ini bertujuan untuk memperoleh pemahaman yang

menyeluruh mengenai teori, konsep, dan temuan yang berkaitan dengan topik penelitian.

2.1.2 Wawancara

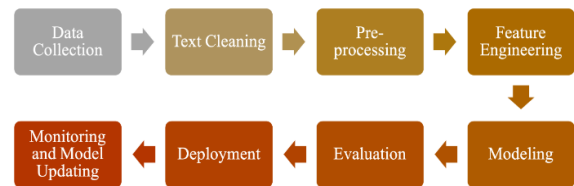
Pada tahap ini, dilakukan interaksi langsung dengan admin Hotline HPPMB UMMI yang melibatkan wawancara mendalam serta diskusi terkait permasalahan yang dihadapi serta solusi yang ditawarkan. Wawancara tersebut dilakukan untuk mendapatkan pemahaman yang lebih mendalam mengenai isu-isu yang relevan dengan pendaftaran mahasiswa baru di Universitas Muhammadiyah Sukabumi. Dalam proses wawancara, dilakukan pengumpulan data berupa pandangan, perspektif, dan pengalaman dari admin Hotline HPPMB UMMI. Informasi yang diperoleh dari wawancara ini menjadi sumber data yang sangat berharga dalam penelitian ini, karena dapat memberikan wawasan yang lebih komprehensif mengenai berbagai aspek yang terkait dengan proses pendaftaran mahasiswa baru. Selain itu, wawancara juga memberikan kesempatan untuk menggali informasi yang mungkin sulit ditemukan melalui sumber lain, sehingga dapat memberikan kontribusi yang signifikan dalam menemukan solusi yang efektif dan relevan dalam meningkatkan pelayanan informasi terkait pendaftaran mahasiswa baru di Universitas Muhammadiyah Sukabumi.

2.1.3 Observasi

Observasi dilakukan dengan tujuan untuk memperoleh pemahaman tentang situasi dan kondisi yang terjadi pada proses Penerimaan Mahasiswa Baru di Universitas Muhammadiyah Sukabumi. Pada tahap ini, observasi dilakukan untuk mengamati langsung dan memperoleh informasi mengenai berbagai aspek yang terkait dengan proses Penerimaan Mahasiswa Baru. Observasi dilakukan dengan tujuan untuk mendapatkan pemahaman tentang situasi aktual yang terjadi di lingkungan Universitas Muhammadiyah Sukabumi saat pelaksanaan penerimaan mahasiswa baru. Dalam observasi ini, berbagai aspek yang diamati meliputi proses pendaftaran, mekanisme seleksi, komunikasi antara pihak universitas dengan calon mahasiswa, serta keterlibatan berbagai pihak terkait dalam proses penerimaan. Observasi dilakukan dengan cermat dan sistematis untuk mendapatkan data dan informasi yang akurat mengenai situasi yang ada.

2.2 Tahapan Penelitian

Dalam pembuatan UMMIBOT metode pengembangan yang digunakan yaitu *Natural Language Processing Pipeline*. *Natural Language Processing Pipeline* merupakan serangkaian langkah berurutan yang digunakan untuk memproses dan menganalisis teks dalam konteks bahasa alami. *Pipeline* NLP terdiri dari beberapa tahapan utama, termasuk pra-pemrosesan, pemrosesan teks, dan analisis teks [8]. Berikut merupakan tahapan dari *NLP Pipeline* sebagaimana ditunjukkan pada Gambar 1 di bawah ini.



Gambar 1. Alur *Natural Language Processing Pipeline*

2.2.1 Data Collection

Data Collection merupakan proses yang melibatkan pengumpulan data teks yang akan digunakan dalam analisis NLP. Sumber data yang digunakan dapat bervariasi, seperti korpus teks, data pengguna, atau sumber data lain yang relevan dengan tujuan aplikasi NLP.

2.2.2 Text Processing

Untuk melakukan pemrosesan teks pada data yang telah diformat sebagai JSON, data tersebut akan dimuat dan dimasukkan ke dalam sebuah daftar yang disesuaikan berdasarkan *tag*. Pada tahap ini, langkah-langkah berikut dilakukan dalam memproses daftar *patterns* yang berisi pertanyaan:

- 1) Menghapus tanda baca dan mengubah huruf menjadi huruf kecil (*remove punctuation and case folding*), mengubah semua huruf kapital menjadi huruf kecil dan menghilangkan tanda baca yang tidak diperlukan [9].
- 2) Normalisasi kata (*word normalization*), kata-kata yang tidak baku atau tidak standar akan diubah menjadi bentuk yang standar menggunakan teknik normalisasi kata. Hal ini bertujuan untuk mencapai konsistensi dalam representasi kata-kata dalam teks [10].
- 3) Penghapusan *stopword* (*stopword removal*), kata-kata yang memiliki tingkat informasi yang rendah atau sering muncul dalam teks, seperti kata hubung atau kata penghubung, akan dihapus menggunakan *library* NLTK *stopword*. Tujuannya adalah untuk fokus pada kata-kata yang lebih relevan dalam analisis [9].
- 4) *Stemming*, kata-kata yang memiliki afiks atau imbuhan akan diubah menjadi bentuk dasarnya melalui proses *stemming* menggunakan *library* *sastrawi*. Dengan melakukan *stemming*, variasi bentuk kata dengan akar yang sama dapat dikurangi, sehingga memudahkan pemrosesan dan pemahaman teks [10].
- 5) Tokenisasi, data yang telah dibersihkan akan dipecah menjadi *token*, yaitu unit-unit kecil seperti kata atau frasa, menggunakan *tokenizer*. Setiap *token* akan diberi indeks dan kata yang sesuai dalam pembentukan kamus (*vocabulary*) [9].
- 6) *Sequence*, tahapan dimana teks diubah menjadi representasi numerik yang dapat dipahami oleh model *machine learning*. Setiap *token* dalam urutan tersebut mewakili kata atau unit penting dalam teks, dan urutan *token* tersebut membentuk informasi berurutan yang dapat dimanfaatkan oleh model [9].

7) *Padding*, sebelum data masuk ke lapisan *embedding*, dilakukan *padding* untuk memastikan panjang teks dalam urutan yang seragam. Hal ini penting agar data dapat diproses secara efisien dan konsisten oleh model jaringan saraf [9].

2.2.3 Feature Engineering

Persiapan data untuk *machine learning* seringkali melibatkan tugas utama yang disebut *Feature Engineering* [11]. Proses ini merupakan langkah dalam mengembangkan fitur-fitur yang relevan dari fitur-fitur yang ada, dengan tujuan untuk meningkatkan kinerja prediksi atau klasifikasi. *Feature engineering* merupakan proses ekstraksi fitur dari data mentah dan mengubahnya menjadi format yang sesuai dengan model *machine learning* [12].

2.2.4 Modeling

Pada tahap ini, dilakukan pelatihan model mesin yang telah di-vektorisasi menggunakan data latihan. Model akan mempelajari pola-pola yang ada dalam data latihan dan menyesuaikan parameter internalnya untuk meningkatkan performa. Dalam pengembangan *chatbot*, model yang digunakan adalah *Bidirectional Long Short-Term Memory*. Metode dalam *deep learning* yang dikenal sebagai *Bidirectional Long Short-Term Memory* (Bi-LSTM) terdiri dari dua *layer* LSTM yang beroperasi secara bertentangan. *Layer* pertama, yang disebut *forward layer*, memproses kata-kata dari awal hingga akhir. Sementara itu, *layer* kedua, yang disebut *backward layer*, memproses kata-kata dari akhir hingga awal. Dengan demikian, Bi-LSTM mampu memperoleh informasi kontekstual dari kedua arah, yang berguna dalam pemrosesan bahasa alami dan tugas-tugas lain dalam analisis teks [13].

Setelah proses pelatihan selesai, model siap digunakan untuk melakukan klasifikasi dan prediksi pada data baru. *Activation function* yang digunakan dalam model ini adalah *softmax*, yang menghasilkan probabilitas untuk setiap kategori atau *label*.

2.2.5 Inference

Tahap *Inference* melibatkan replikasi *chatbot* dari tahap menerima *input* hingga memberikan respon. Pada tahap ini, *chatbot* akan menganalisis *input* yang diterima dan menggunakan model atau algoritma yang telah dilatih sebelumnya untuk menghasilkan respon yang sesuai. Proses ini melibatkan pemrosesan *input*, pemrosesan bahasa alami, dan penerapan aturan atau logika yang telah ditentukan. Setelah *input* diproses dan informasi yang relevan diambil, *chatbot* akan menggunakan pengetahuan dan pemahaman yang dimilikinya untuk memberikan respon yang tepat. Respon yang dihasilkan dapat berupa teks, suara, atau visual, tergantung pada implementasi *chatbot* yang digunakan.

2.2.6 Deployment

Pada langkah ini, *chatbot* akan diintegrasikan ke dalam sebuah *website* dengan menggunakan kerangka kerja *Flask*. Proses ini melibatkan pengembangan dan konfigurasi *chatbot* agar dapat berinteraksi dengan pengguna melalui *website* yang telah dibuat menggunakan *Flask*. Kerangka kerja *Flask* memungkinkan pengembang untuk membuat titik akhir (*endpoint*) dan menghubungkan fungsi-fungsi *chatbot* dengan permintaan pengguna yang diterima melalui *website* tersebut [14].

2.2.7 Evaluation

Pada tahap ini, dilakukan evaluasi dan pengujian terhadap model untuk mengevaluasi kualitas dan performanya. Evaluasi dilakukan dengan menggunakan metode *blackbox testing* yang fokus pada pengujian fungsionalitas *chatbot*. *Blackbox testing* adalah suatu pendekatan dalam pengujian perangkat lunak yang berfokus pada pengujian fungsi-fungsi atau fitur-fitur aplikasi tanpa mempertimbangkan struktur *internal* atau cara kerja aplikasi tersebut. [15]. Evaluasi ini dilakukan oleh admin *hotline* UPT HPPMB melalui *website* untuk menguji kesesuaian *input* dan respons *chatbot*. *Input* yang diberikan berupa pertanyaan yang sering ditanyakan kepada admin *hotline*, termasuk proses pendaftaran, pembayaran, perpindahan jurusan, informasi mengenai *browsure*, dan beasiswa. Melalui evaluasi tersebut, dapat diidentifikasi kelebihan dan kekurangan *chatbot* dalam menjalankan tugasnya. Evaluasi secara menyeluruh memberikan wawasan yang berharga untuk meningkatkan kualitas dan performa *chatbot*.

3. HASIL DAN PEMBAHASAN

Penelitian ini menggambarkan pengembangan UMMIBOT, sebuah *chatbot* yang dibuat untuk membantu calon mahasiswa baru di Universitas Muhammadiyah Sukabumi dalam mendapatkan informasi seputar penerimaan mahasiswa baru. *Chatbot* ini dikembangkan dengan menggunakan metode Natural Language Processing (NLP) Pipeline, yang memungkinkan *chatbot* untuk memahami bahasa alami pengguna dan memberikan respons yang sesuai.

3.1. Data Collection

Dataset yang diperoleh melalui wawancara dengan admin *Hotline* UPT HPPMB UMMI serta pengumpulan data secara langsung melalui *website* PMB UMMI. Adapun data yang dikumpulkan yaitu informasi seputar Universitas Muhammadiyah Sukabumi dan riwayat pertanyaan dari calon mahasiswa pada admin *hotline* UPT HPPMB UMMI. Dalam proses pengumpulan data, diperoleh 101 kategori *dataset* dan 425 pertanyaan yang akan digunakan sebagai *dataset* dalam pembuatan UMMIBOT. Setelah memperoleh data, dilakukan analisis dan pemahaman terhadap data tersebut. Data yang telah diperoleh akan disusun dalam format JSON yang mencakup elemen-elemen seperti *intent*, *tag*, *patterns*, dan *response*. Elemen *tag* akan berfungsi

sebagai *label* atau kategori, *patterns* berisi daftar pertanyaan yang terkait dengan *tag* tersebut, dan *response* merupakan jawaban yang akan diberikan. Struktur JSON yang digunakan dapat ditemukan pada Gambar 2 seperti yang ditunjukkan di bawah ini.

```

1- {}
2- "intents": [
3-   {
4-     "tag": "salam",
5-     "patterns": [
6-       "Hai",
7-       "Hi",
8-       "Halo",
9-       "Apa Kabar",
10-      "Selamat Pagi!",
11-      "Selamat Siang!",
12-      "Selamat Malam!",
13-      "Salam",
14-      "Assalamualaikum wr.wb",
15-      "Hallo"
16-    ],
17-    "responses": [
18-      "Halo!",
19-      "Hai",
20-      "Halo, ada yang bisa saya bantu?",
21-      "Halo selamat datang",
22-      "Hai Kawan"
23-    ]
24-  },

```

Gambar 2. Implementasi Struktur JSON

3.2. Text Processing

Dataset yang terdapat dalam daftar *patterns* akan melalui proses pembersihan agar lebih terstruktur. Berikut merupakan tahapan dari *text processing*:

3.2.1 Remove Punctuation and Case Folding

Pada tahap ini tanda baca akan dihilangkan dan mengganti semua huruf kapital menjadi huruf kecil. Untuk implementasi *remove punctuation and case folding* dapat dilihat pada Gambar 3 berikut ini.

Sebelum	Sesudah
Informasi tentang perpustakaan ummi?	informasi tentang perpustakaan ummi
apakah di ummi menerima beasiswa KIP?	apakah di ummi menerima beasiswa kip
Bagaimana cara melakukan pengunduran diri dari... PTI	bagaimana cara melakukan pengunduran diri dari... pti
Agribisnis ummi FE	agribisnis ummi fe
logo ummi	logo ummi
Bagaimana sistem kehadiran di ummi? apa itu ILC?	bagaimana sistem kehadiran di ummi apa itu ilc?
jurusan yang tersedia?	jurusan yang tersedia

Gambar 3. Perbandingan Teks Sebelum dan Sesudah Proses *Remove Punctuation and Case Folding*

3.2.2 Word Normalization

Pada tahap ini, kata-kata yang tidak standar akan diubah menjadi kata-kata standar atau baku. Contohnya, kata "abis" akan diubah menjadi "habis", dan kata "info" akan diubah menjadi "informasi". Implementasi dari *word normalization* bisa dilihat pada Gambar 4 berikut ini.

```

key_norm = pd.read_csv('key_norm.csv')

def text_normalize(text):
    text = ' '.join([key_norm[key_norm['singkat'] == word]['hasil'].values[0]
                    if (key_norm['singkat'] == word).any()
                    else word for word in text.split()])
    text = str.lower(text)
    return text

print(text_normalize('info'))

```

Gambar 4. Contoh Implementasi *Word Normalization*

3.2.3 Stopword

Pada tahap ini kata yang tidak penting biasanya berupa kata hubung akan dihilangkan. Misalnya kata "ada", "dan", "di", dan lainnya. Implementasi *stopword* bisa dilihat pada Gambar 5 berikut ini.

	Sebelum	Sesudah
225	saya ingin bertanya saya ingin ganti pilihan j...	ganti pilihan jurusan
227	Saya dari kelas reguler ingin pindah ke non re...	kelas reguler pindah non reguler
106	biaya kuliah non reguler	biaya kuliah non reguler
183	visi misi upt humas dan pmb	visi misi upt humas pmb
328	tahapan pembayaran menggunakan bank Dana	tahapan pembayaran bank dana
324	cara pembayaran menggunakan bank Muamalat?	pembayaran bank muamalat
274	cara membuat SKL	skl
377	Administrasi Bisnis	administrasi bisnis
73	program pasca sarjana	program pasca sarjana
62	RPL	rpl

Gambar 5. Perbandingan Teks Sebelum dan Sesudah Proses *Stopword*

3.2.4 Stemming

Di tahap ini kata berimbuhan akan diubah dalam bentuk dasarnya, misalnya kata "pindahan" menjadi "pindah". Implementasi *stemming* dapat dilihat pada Gambar 6 berikut ini.

	Sebelum	Sesudah
328	tahapan pembayaran menggunakan bank Dana	tahap bayar bank dana
331	Fakultas Keguruan dan Ilmu Pendidikan ummi	fakultas guru ilmu didik ummi
277	cara membuat surat keterangan pengganti ijazah	surat terang ganti ijazah
146	fasilitas yang ada di ummi	fasilitas ummi
209	Bagaimana cara pembayaran perkuliahan mengguna...	bayar kuliah metode transfer bank ummi
156	Apa saja tugas dari Lembaga al islam dan kemuh...	tugas lembaga al islam kemuhammadiyah
355	Fakultas Ekonomi	fakultas ekonomi
405	Fakultas Hukum ummi	fakultas hukum ummi
235	pindah dari ummi	pindah ummi
416	arti logo ummi	arti logo ummi

Gambar 6. Perbandingan Teks Sebelum dan Sesudah Proses *Stemming*

3.2.5 Tokenizing and Fit on Text

Di tahap ini kalimat pada *dataset* akan dipecah menjadi *token*. Dan dijadikan kamus yang berisi kosa kata dalam *dataset*. Implementasi *tokenizing and fit on text* dapat dilihat pada Gambar 7 berikut ini.

```
{1: 'ummi',
 2: 'bayar',
 3: 'daftar',
 4: 'program',
 5: 'bank',
 6: 'fakultas',
 7: 'kuliah',
 8: 'pindah',
 9: 'beasiswa',
10: 'kampus',
11: 'studi',
12: 'tahap',
```

Gambar 7. DaftarToken yang Dihasilkan Setelah Proses *Tokenizing and Fit on Text*

3.2.6 Sequence

Token yang dipecah akan diubah menjadi *integer* dimana nilai *integer* akan bergantung pada urutan kata dalam kamus yang telah dibuat. Berikut contoh implementasi *sequence* sebagaimana ditunjukkan pada Gambar 8 berikut ini.

```
print("Sequences = ", train[4])
print("Artinya = ", ['daftar[61]', ', ', 'daftar[194],'])
```

Sequences = [61, 194]
Artinya = [selamat , pagi]

Gambar 8. Pembentukan *Sequence* pada *Dataset*

3.2.7 Padding

Di tahap ini akan dilakukan proses penyamarataan panjang dari *sequence*. Berikut implementasi dari *padding* sebagaimana ditunjukkan pada Gambar 9 berikut ini.

```
print(x_train)
```

```
[[190  0  0 ...  0  0  0]
 [191  0  0 ...  0  0  0]
 [192  0  0 ...  0  0  0]
 ...
 [309  1  0 ...  0  0  0]
 [189  1  0 ...  0  0  0]
 [189 21  0 ...  0  0  0]]
```

Gambar 9. Contoh Implementasi *Padding* pada *Dataset*

3.3. Feature Engineering

Setelah data melalui tahapan *text processing*, selanjutnya data akan diubah menjadi vektor numerik berdimensi rendah dimana vektor-vektor tersebut memiliki hubungan semantik antar kata.

3.4. Modeling

Data yang sudah divektorisasi akan dimasukan dalam model untuk ditrain. Model yang digunakan dalam penelitian ini yaitu *Bidirectional Long Short-Term Memory*. Berikut merupakan struktur model yang digunakan sebagaimana ditunjukkan pada Gambar 10 berikut ini.

```
model.summary()
```

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 7, 10)	3100
bidirectional (Bidirectional)	(None, 20)	1680
flatten (Flatten)	(None, 20)	0
dense (Dense)	(None, 101)	2121

Total params: 6,901
Trainable params: 6,901
Non-trainable params: 0

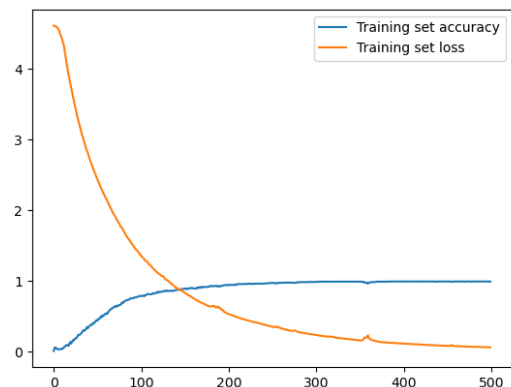
Gambar 10. Struktur Model yang Digunakan

Berikut merupakan hasil *training model* dari 500 iterasi dengan jumlah *learning rate* sebesar 0,01 yang ditunjukkan pada Gambar 11 berikut ini.

```
...
Epoch 499/500
14/14 [=====] - 0s 14ms/step - loss: 0.0624 - accuracy: 0.9906
Epoch 500/500
14/14 [=====] - 0s 18ms/step - loss: 0.0619 - accuracy: 0.9930
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

Gambar 11. Hasil *Training Model*

Grafik dari hasil *training model* dapat dilihat pada Gambar 12 berikut ini.



Gambar 12. Grafik Hasil *Train*

3.5. Inference

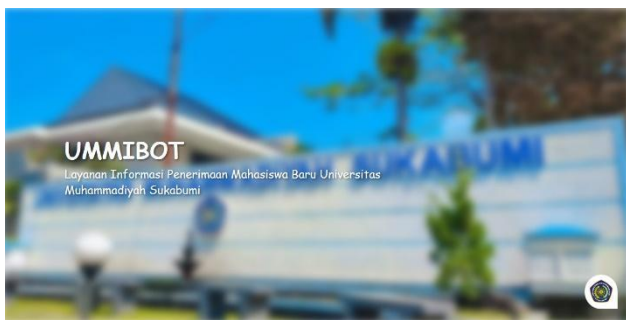
Pada tahap ini akan dilakukan simulasi percakapan dengan model yang telah dibuat. Model akan menganalisis *input* dan memberikan respon. Implementasi *inference* bisa dilihat pada Gambar 13 berikut ini.

```
input : makasih
text_p ['terimakasih']
p1 [[121]]
p2 [121]
p3 [[121 0 0 0 0 0 0]]
1/1 [=====] - 0s 53ms/step
1/1 [=====] - 0s 49ms/step
out 88
bye
answer : Kalau ada masalah, hubungi aku lagi ya
```

Gambar 13. Inference

3.6. Deployment

Model yang telah dibuat akan diintegrasikan dalam *website* menggunakan *flask*. Adapun tampilan *website* dapat dilihat pada Gambar 14 yang menunjukkan tampilan awal *website* sebelum *bubble* di klik dan Gambar 15 yang menunjukkan tampilan *chatbot* setelah *bubble* di klik.



Gambar 14. Tampilan Awal Chatbot



Gambar 15. Tampilan Chatbox pada Chatbot

3.7. Evaluation

Pengujian menggunakan *blackbox testing* yang berfokus pada fungsionalitas dan kualitas respon yang dihasilkan. Berikut merupakan hasil pengujian dari *chatbot* sebagaimana ditunjukkan pada Tabel 1 berikut ini.

Tabel 1. Uji Fungsionalitas Chatbot

Komponen yang Diuji	Skenartion Pengujian	Keterangan
Input tidak ada dalam dataset	Menguji percakapan dimana <i>input user</i> tidak ada dalam <i>dataset</i> .	Sesuai
Input ada dalam dataset namun susunan acak	Menguji percakapan dimana <i>input user</i> berada dalam <i>dataset</i> tetapi dengan susunan kata yang acak	Sesuai

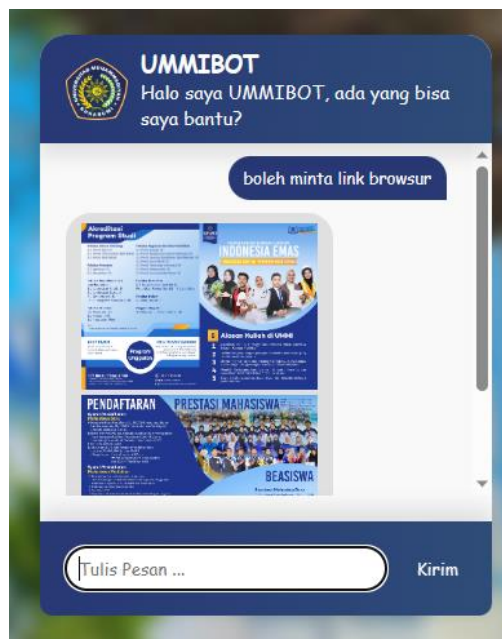
Komponen yang Diuji	Skenartion Pengujian	Keterangan
Input dengan dataset	Menguji percakapan dimana <i>input</i> sesuai dengan <i>dataset</i>	Sesuai
Input dengan makna ganda	Menguji percakapan dimana <i>input</i> mengandung dua makna yang berbeda	Tidak Sesuai

Pada Gambar 16 dapat dilihat bahwa jika *input* tidak tersedia atau terdeteksi dalam kamus maka respon yang diberikan adalah “maaf saya tidak mengerti”.



Gambar 16. Pengujian Input Tidak Ada dalam Dataset

Gambar 17 menunjukkan bahwa jika *user* memberikan *input* sesuai dengan *patterns* pada *dataset* maka respon yang diberikan akan sebagai berikut.



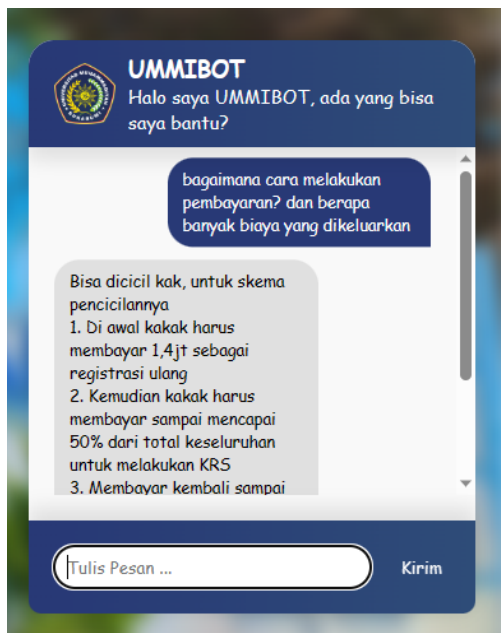
Gambar 17. Pengujian Input Tersedia dalam Dataset

Jika *user* memberikan *input* dengan susunan acak maka respon yang diberikan akan seperti pada Gambar 18 berikut ini.



Gambar 18. Pengujian *Chatbot* setelah Susunan *Input* Diacak

Gambar 19 dibawah ini menunjukkan interaksi dimana jika *input user* mengandung dua makna yang berbeda maka respon yang diberikan akan menjadi *miss* klasifikasi.



Gambar 19. *Input User* Mengandung Dua Makna yang Berbeda

Untuk memperoleh akurasi, *chatbot* akan diuji dengan memberikan pertanyaan dari kategori-kategori yang sering ditanyakan oleh calon mahasiswa kepada admin *Hotline UPT HPPMB UMMI*. Hasil pengujian dari setiap kategori pertanyaan yang ditunjukkan pada Tabel 2 berikut ini.

Tabel 2. Uji Akurasi Berdasarkan Kesesuaian Pertanyaan

No	Pertanyaan	Keterangan
1.	Bagaimana proses pendaftaran untuk kuliah di universitas ini?	Sesuai
2.	Apa persyaratan yang harus dipenuhi untuk mendaftar sebagai mahasiswa baru?	Sesuai
3.	Bagaimana proses pendaftaran kuliah di ummi?	Sesuai
4.	Bagaimana cara mendapatkan informasi lebih lanjut tentang proses pendaftaran kuliah di ummi?	Sesuai
5.	Apakah ada batas waktu pendaftaran untuk kuliah di ummi?	Sesuai
6.	Bagaimana cara melakukan pembayaran perkuliahan di UMMI?	Sesuai
7.	Apa saja metode pembayaran yang tersedia untuk pembayaran perkuliahan di UMMI?	Sesuai
8.	Apakah bisa melakukan pembayaran melalui bank BSI? Bagaimana caranya?	Sesuai
9.	Bagaimana cara pembayaran perkuliahan menggunakan metode transfer bank di UMMI?	Sesuai
10.	Apakah di UMMI menyediakan pembayaran angsuran untuk perkuliahan?	Sesuai
11.	Apa saja persyaratan untuk melakukan perpindahan prodi di UMMI?	Sesuai
12.	Bagaimana tahapan untuk melakukan perpindahan program studi?	Sesuai
13.	Selain pindah program studi apakah juga bisa pindah fakultas?	Sesuai
14.	Apakah ada batasan waktu atau semester tertentu untuk melakukan perpindahan prodi di UMMI?	Tidak Sesuai
15.	Syarat apa saja yang diperlukan agar bisa pindah kampus?	Sesuai
16.	Bagaimana cara mengajukan beasiswa di UMMI?	Sesuai
17.	Apa saja jenis beasiswa yang tersedia di UMMI?	Sesuai
18.	Bagaimana cara mengajukan Beasiswa KIP di UMMI dan apa persyaratannya?	Sesuai
19.	Apakah ada beasiswa khusus untuk mahasiswa yang berasal dari latar belakang ekonomi kurang mampu, seperti Beasiswa KIP (Kartu Indonesia Pintar)?	Tidak Sesuai
20.	Kalo dapet beasiswa dapet potongannya berapa?	Sesuai
21.	Kak kalau misalnya udah ditahap menunggu ujian masih bisa ganti prodi gak?	Tidak Sesuai
22.	Saya dapat nilai ujian terbesar di sekolah, apakah bisa mengajukan beasiswa?	Sesuai
23.	Bagaimana cara melakukan pembayaran lewat BCA	Sesuai
24.	Bagaimana cara melakukan pembayaran?	Sesuai
25.	Bagaimana cara melakukan pendaftaran?	Sesuai
26.	Penyetaraan jurusannya apa saja?	Tidak Sesuai
27.	Bagaimana cara melakukan transfer bank BNI?	Sesuai
28.	Kan di ummi ada beasiswa non akademik kriterianya apa saja?	Sesuai
29.	Kak pembayaran melalui bank BCA gimna caranya?	Sesuai
30.	Kalo beasiswa dapet potongan berapa?	Sesuai
31.	Untuk pendaftaran online gimana caranya ya kak?	Sesuai

No	Pertanyaan	Keterangan
32.	Apakah masih ada beasiswa untuk alumni muhammadiyah?	Sesuai
33.	Di ummi menerima beasiswa KIP gak? kalo menerima bagaimana langkah-langkahnya?	Sesuai
34.	Tahapan pendaftaran beasiswa KIP bagaimana?	Sesuai
35.	Kalo pembayaran lewat dana gimana caranya?	Tidak Sesuai
36.	Biaya pendaftaran untuk mahasiswa baru berapa ya kak?	Tidak Sesuai
37.	Biaya pendaftaran secara keseluruhan berapa?	Tidak Sesuai
38.	Saya ingin ganti pilihan jurusan saya, bisa ganti gak?	Sesuai
39.	Boleh saya tau biaya kuliah non reguler kak?	Sesuai
40.	Aku mau tanya seputar pendaftaram, gimana caranya?	Sesuai
41.	Bagaimana tahapan untuk perpindahan program studi?	Sesuai
42.	Mau tanya boleh? kalo uang semester 1 bisa dicicil atau harus langsung semua?	Sesuai
43.	Kalo kelas non reguler itu bagaimana?	Sesuai
44.	Kalo biaya pendidikan itu bayarnya bisa bertahap gak?	Sesuai
45.	Apakah sekarang tersedia beasiswa peringkat kelas?	Sesuai
46.	Boleh minta info rincian biaya persemetermya?	Sesuai
47.	Izin bertanya mengenai informasi penggunaan kip di ummi?	Tidak Sesuai
48.	Kalo di ummi bisa menggunakan KIP kuliah?	Sesuai
49.	Untuk biaya perkuliahan itu bagaimana?	Sesuai

Berdasarkan tabel di atas terdapat satu respon yang tidak sesuai. Hal ini dikarenakan ketika *input* diproses dan diprediksi terdapat 2 atau lebih tag yang memiliki data yang sama seperti *input*. Kata “batasan waktu” terdapat pada tag “jadwal pendaftaran” dan “perpindahan prodi” berada pada tag “perpindahan prodi dan fakultas” sehingga respon menjadi *miss predict*. Dengan demikian, akurasi dari *chatbot* dapat dihitung dengan menggunakan rumus:

$$\frac{\text{Jawaban Benar}}{\text{Total Pertanyaan}} \times 100\%$$

Pada tabel 2 di atas dapat dilihat pengujian respon *chatbot* terhadap inputan dari pengguna. Hasil yang didapatkan yaitu terdapat 8 respon yang tidak sesuai dari 49 pertanyaan. Sehingga dapat dihitung akurasi dari sistem yaitu sebesar 83,67 %.

4. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan pada pembuatan *chatbot* sebagai media untuk memberikan layanan informasi terkait pendaftaran mahasiswa baru di Universitas Muhammadiyah Sukabumi dengan menggunakan *natural language processing* ditemukan bahwa *chatbot* yang telah dibuat dan diberi nama UMMIBOT berhasil mencapai tingkat akurasi

fungsionalitas sebesar 83,67%. Hal ini menunjukkan bahwa UMMIBOT mampu memberikan respons yang sesuai dengan input yang diberikan oleh pengguna dengan tingkat keakuratan yang tinggi. Namun, selama pengujian terdapat beberapa kelemahan yang perlu diperhatikan dalam UMMIBOT. Salah satu kelemahan yang teridentifikasi adalah ketidakmampuan untuk mengenali kalimat-kalimat dengan makna ganda. Misalnya, ketika pengguna memberikan input yang mengandung kata-kata dengan makna ganda, UMMIBOT tidak mampu membedakan konteks yang dimaksud oleh pengguna, akibatnya respons yang diberikan tidak sesuai dengan harapan pengguna. Secara keseluruhan, pengembangan UMMIBOT adalah langkah yang positif dalam memanfaatkan teknologi untuk meningkatkan pelayanan informasi di bidang penerimaan mahasiswa baru. Dengan perbaikan yang tepat, UMMIBOT memiliki potensi untuk menjadi alat yang sangat berguna dalam membantu calon mahasiswa baru dalam memahami proses pendaftaran dan mendapatkan informasi yang dibutuhkan dengan cepat dan akurat.

DAFTAR PUSTAKA

- [1] Z. N. Adesfiana, I. Astuti, and E. Enawaty, “Pengembangan Chatbot Berbasis Web Menggunakan Model Addie,” *J. Khatulistiwa Inform.*, vol. 10, no. 2, pp. 147–152, 2022, [Online]. Available: <https://ejournal.bsi.ac.id/ejurnal/index.php/khatulistiwa/article/view/14050>
- [2] B. Rusmarasy, B. Priyambadha, and F. Pradana, “Pengembangan Chat Bot pada CoMa untuk Memberikan Motivasi Kepada Pengguna Menggunakan AIML,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 5, pp. 4484–4490, 2019.
- [3] A. A. Chandra, V. Nathaniel, and F. R. Satura, “Pengembangan Chatbot Informasi Mahasiswa Berbasis Telegram dengan Metode Natural Language Processing,” vol. 3, no. 1, pp. 20–27, 2022.
- [4] H. E. Rosyadi, F. Amrullah, R. D. Marcus, and R. R. Affandi, “Rancang Bangun Chatbot Informasi Lowongan Pekerjaan Berbasis Whatsapp dengan Metode NLP (Natural Language Processing),” *BRILIANT J. Ris. dan Konseptual*, vol. 5, no. 1, pp. 619–626, 2020.
- [5] S. Wijanarko, “Analisis Kesesuaian Komentar Mahasiswa Pada Sistem Akademi Online Angket Penilaian Dosen Menggunakan Supervised Model,” *J. Tek. Inform.*, vol. 8, no. 2, pp. 74–84, 2022, doi: 10.51998/jti.v8i2.503.
- [6] A. S. B. Aji, “Membangun Chatbot Layanan Helpdesk Perpajakan Kpp Pratama Jakarta Setiabudi Satu,” *Sebatik*, vol. 26, no. 1, pp. 194–201, 2022, doi: 10.46984/sebatik.v26i1.1916.
- [7] E. Mursidah, L. Ambarwati, A. Karima, K. Kunci, : Chatbot, and M. Baru, “Implementasi Chatbot Layanan Informasi Pendaftaran Mahasiswa Baru Program Pascasarjana Departemen Teknik Informatika Its Implementation of Chatbot Information Services for New Student Registration

- Postgraduate Program Its Information Engineering Departmenten,” *J. Ilm. NERO*, vol. 7, no. 1, p. 2022, 2022.
- [8] A. Chen, “NLP Pipeline,” *Python Notes For Linguistics*, 2020. <https://alvinntnu.github.io/python-notes/nlp/nlp-pipeline.html#a-general-nlp-pipeline> (accessed Jul. 03, 2023).
- [9] P. B. Wintoro, H. Hermawan, M. A. Muda, and Y. Mulyani, “Implementasi Long Short-Term Memory pada Chatbot Informasi Akademik Teknik Informatika Unila,” *Expert J. Manaj. Sist. Inf. dan Teknol.*, vol. 12, no. 1, p. 68, 2022, doi: 10.36448/expert.v12i1.2593.
- [10] A. R. W. Rapsanjani and E. Junianto, “Implementasi Probabilistic Neural Network Dan Word Embedding Untuk Analisis Sentimen Vaksin Sinovac,” *J. Responsif Ris. Sains dan Inform.*, vol. 3, no. 2, pp. 233–242, 2021, doi: 10.51977/jti.v3i2.588.
- [12] C. M. Sitorus, A. Rizal, and M. Jajuli, “Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine,” *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 2, pp. 254–265, 2020, doi: 10.28932/jutisi.v6i2.2672.
- [13] E. Subowo, F. Adi Artanto, I. Putri, and W. Umaedi, “BLTSM untuk analisis sentimen berbasis aspek pada aplikasi belanja online dengan cicilan,” *J. Fasilkom*, vol. 12, no. 2, pp. 132–140, 2022.
- [14] R. Somya, “Perancangan Aplikasi Chatting Berbasis Web di PT. Pura Barutama Kudus menggunakan Socket.IO dan Framework Foundation,” *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 4, no. 1, pp. 8–15, 2018, doi: 10.23917/khif.v4i1.5979.
- [15] A. P. Putra, F. Andriyanto, K. Karisman, T. D. M. Harti, and W. P. Sari, “Pengujian Aplikasi Point of Sale Menggunakan Blackbox Testing,” *J. Bina Komput.*, vol. 2, no. 1, pp. 74–78, 2020, doi: 10.33557/binakomputer.v2i1.757.



PENGEMBANGAN *REST API* DENGAN MENGGUNAKAN *EXPRESS JS* UNTUK Mencari Mentor Pribadi

Nasrul¹, Achmad Izhar²

^{1,2}Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri
Jakarta Selatan, DKI Jakarta, Indonesia 12640
nasrul@nurulfikri.ac.id, achm19068ti@student.nurulfikri.ac.id

Abstract

Currently there is a problem faced by many students, namely in understanding and mastering programming. These problems occur due to limitations in understanding program code, limited access to learning media, and limited learning time. To answer this challenge, the design and development of the Rest API for finding personal mentors is proposed as a solution that can improve the quality and effectiveness of learning program code. In this research, the Express JS framework is used to implement the Rest API, which interacts with third party APIs such as Midtrans and Redis. The MVC development approach and the Scrum method are applied to manage the application development process. The aim of this research is to design and build a Rest API that allows students to find a personal mentor that suits their needs in learning code. It is hoped that the use of Express JS and the Scrum method in the development of this application will facilitate the creation of effective and efficient private coding applications.

Keywords: Express JS, Mentor, MVC, Rest API, Scrum

Abstrak

Saat ini ada sebuah permasalahan yang dihadapi oleh banyak mahasiswa, yaitu dalam memahami dan menguasai pemrograman. Permasalahan tersebut terjadi akibat keterbatasan dalam memahami kode program, keterbatasan akses ke media pembelajaran, dan keterbatasan waktu pembelajaran. Untuk menjawab tantangan ini, perancangan dan pengembangan Rest API untuk pencarian mentor pribadi diusulkan sebagai solusi yang dapat meningkatkan kualitas dan efektivitas pembelajaran kode program. Dalam penelitian ini, digunakan *Framework Express JS* untuk mengimplementasikan *Rest API*, yang berinteraksi dengan *API* pihak ketiga seperti *Midtrans* dan *Redis*. Pendekatan pengembangan *MVC* dan metode *Scrum* diterapkan untuk mengatur proses pengembangan aplikasi. Tujuan dari penelitian ini adalah untuk merancang dan membangun sebuah *Rest API* yang memungkinkan mahasiswa mencari mentor pribadi yang sesuai dengan kebutuhan mereka dalam pembelajaran kode program. Diharapkan bahwa penggunaan *Express JS* dan metode *Scrum* dalam pengembangan aplikasi ini akan memfasilitasi penciptaan aplikasi *private coding* yang efektif dan efisien.

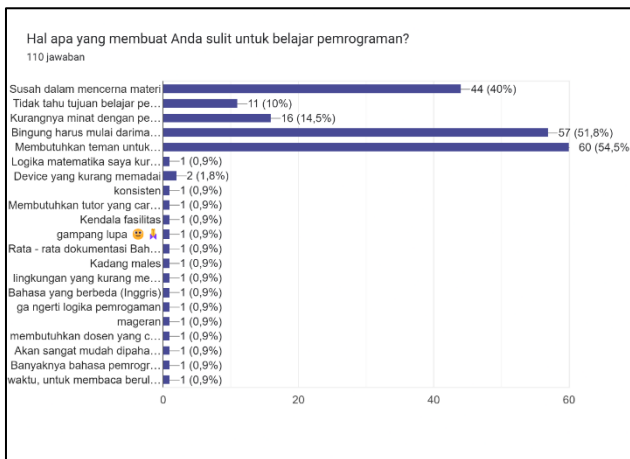
Kata kunci: Express JS, Mentor, MVC, Rest API, Scrum

1. PENDAHULUAN

Hampir semua kegiatan yang terjadi saat ini dapat dengan mudah dilakukan dengan bantuan teknologi. Ada banyak cara memajukan teknologi di Indonesia, salah satunya dengan bersekolah atau berkuliah di jurusan teknik, seperti teknik informatika, teknologi informasi, sistem informasi, sistem informatika, manajemen informatika atau jurusan lainnya. Menurut penelusuran yang dilakukan, setiap tahunnya lulusan TI di Indonesia tidak kurang dari 400 ribu orang, harapannya lulusan ini menjadi fondasi perkembangan teknologi di Indonesia yang tertinggal dari beberapa negara lain yang ada di dunia saat ini.

Akan tetapi mayoritas lulusan TI Indonesia saat ini masih belum memenuhi sumber daya manusia yang dibutuhkan oleh industri pada saat ini, walaupun SDM mahasiswa dalam ranah teknologi di Indonesia diperkirakan dapat mencapai 200 ribu orang. dan jumlah tersebut akan terus bertambah, seiring dengan perkembangan teknologi informasi yang begitu naik pesat [1]. Ada berbagai penyebab hal tersebut dapat terjadi, berdasarkan identifikasi Narenda selaku CEO Dicoding, saat ini ada 3 masalah yang menyebabkan permasalahan terjadi, pertama faktor kurikulum pembelajaran, kedua karena kualitas pengajar TI belum merata, dan yang terakhir adalah dari kualitas sumber daya manusia belum memadai [2].

Setelah mengetahui permasalahan pada saat ini, peneliti mengadakan survei lanjutan kepada 110 mahasiswa jurusan Teknik Informatika (TI) dimana hasil survei lanjutan tersebut menyimpulkan bahwa terdapat 46 mahasiswa jurusan TI merasa susah dalam mencerna materi pemrograman. Dari 12 mahasiswa menyatakan bahwa mereka tidak tahu tujuan belajar pemrograman, 16 mahasiswa kurang minat dengan pemrograman, 61 mahasiswa bingung memulai belajar pemrograman dari mana dan 61 mahasiswa membutuhkan teman untuk dapat mempelajari pemrograman bersama dan sisanya menjawab yang lainnya. Adapun survei lanjutan terhadap 110 mahasiswa tersebut seperti halnya yang terlampir pada gambar 1.



Gambar 1. Jawaban Survei Lanjutan Terhadap 110 Mahasiswa

Dari hasil survei lanjutan yang dilakukan, maka telah diperoleh 98,5% dari 110 mahasiswa yang melakukan pengisian survei menyatakan bahwa membutuhkan mentor pribadi untuk belajar pemrograman dengan mudah dan berkualitas dengan fitur pencarian dan fitur reservasi mentor yang mudah. Hasil survei lanjutan dapat dilihat pada gambar 2.



Gambar 2. Survei Membutuhkan Mentor untuk Belajar Pemrograman

Berdasarkan permasalahan saat ini dan survei yang telah dilakukan, maka peneliti melakukan rancangan dan pembangunan *Rest API* menggunakan *express js* untuk mencari mentor pribadi. *Rest API* yang akan dikembangkan

diharapkan dapat memberikan kontribusi terbaik pencarian mentor sesuai dengan kebutuhan dan preferensi pengguna.

Dengan mentoring, pendampingan dan pendekatan tertentu lebih luas diharapkan dapat membangun hubungan antara mentor dan *mentee* dalam waktu yang panjang [3], sehingga dapat memperluas dan meningkatkan aksesibilitas pembelajaran individu yang mungkin terbatas oleh faktor geografis atau keterbatasan pertukaran informasi, diskusi, dan penjelasan konsep dan praktik pemrograman. Selain itu, diharapkan implementasi aplikasi *private coding* dengan *express js* memberikan manfaat signifikan bagi mahasiswa yang ingin belajar pemrograman dan mencari mentor pribadi.

2. METODE PENELITIAN

Penelitian ini memiliki rancangan penelitian seperti metode analisis, metode dalam pengumpulan data, dan metode untuk pengembangan aplikasi *private coding*. Metode analisis ini menggunakan metode kualitatif dengan kuesioner yang berisi pertanyaan relevan untuk mengetahui hal-hal yang dibutuhkan pengguna aplikasi dalam menggunakan aplikasi *private coding* dan apa yang dibutuhkan oleh peneliti dalam mengembangkan penelitian ini menjadi lebih baik dan bermanfaat dalam jangka waktu panjang untuk masyarakat.

2.1 Metode Pengumpulan Data

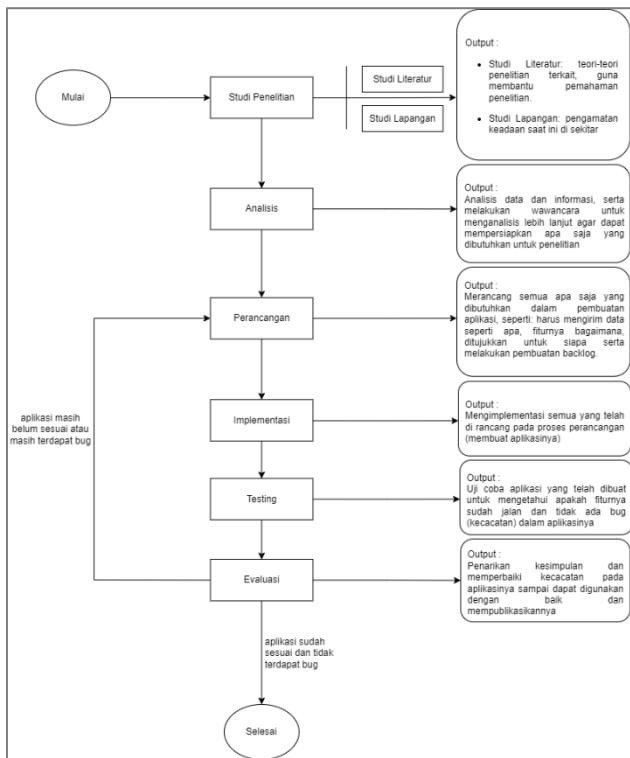
Metode pengumpulan data dilakukan dengan kondisi alami untuk mempermudah kebutuhan penelitian dengan sumber data utama dan teknik pengumpulan data sebagai berikut:

1. Wawancara calon pengguna aplikasi, mentor, atau individu yang terlibat dalam proses pembelajaran agar memberikan pemahaman mendalam tentang kebutuhan, harapan, pengalaman pengguna serta perspektif *private coding* dan pencarian mentor.
2. Observasi dan pengamatan langsung pengguna aplikasi dan melibatkan mentor pembelajaran.
3. Kuesioner mahasiswa ranah teknologi (jurusan teknik informatika, sistem informasi, teknologi informasi, dan lainnya) terkait pemrograman dan kuesioner pengguna serta mentor pemrograman untuk mendapatkan umpan balik aplikasi *private coding*.

Selain metode analisis dan pengumpulan data, penelitian ini menggunakan *scrum* untuk pengembangan aplikasi. Metode *Scrum* adalah kerangka kerja yang sifat pekerjaannya kompleks dan cepat berubah. *Scrum* ini dapat menjawab permasalahan yang adaptif kompleks dan menghasilkan kreativitas dan inovatif [4]. Setiap acara di dalam *Scrum* memiliki batasan waktu, yang artinya selalu memiliki durasi maksimum. Pada saat *sprint* dimulai, durasinya tetap dan tidak dapat diperpendek maupun diperpanjang. Acara-acara lainnya dapat diakhiri saat tujuan dari acara tersebut telah tercapai [5].

2.2 Tahapan Penelitian

Tahapan penelitian menggunakan metode *Scrum* dapat dilihat pada gambar 3 berikut.



Gambar 3. Tahapan Penelitian

Tahapan penelitian pada gambar 3 dapat dijelaskan sebagai berikut:

1. Studi Literatur

Studi literatur mencari referensi artikel, jurnal penelitian sebelumnya dan pembuatan *prototype* aplikasi berbasis *web*. *Prototype* adalah versi tahap awal dari sistem perangkat lunak yang digunakan untuk memberikan gambaran tentang sebuah ide, menguji desain, menemukan sebanyak mungkin masalah, dan menemukan solusi untuk masalah tersebut [6]. Hasil studi literatur dalam penelitian ini berupa pembuatan model penelitian untuk menjadi acuan dan hal apa saja yang dibutuhkan dan diperlukan agar mencapai tujuan penelitian.

2. Analisis

Pada tahap analisis, peneliti mengidentifikasi permasalahan yang ada terkait proses pembuatan *Rest API* aplikasi *private coding*, yang nantinya dikembangkan untuk memecahkan permasalahan dalam penelitian ini. Dengan mengidentifikasi permasalahan tersebut, peneliti kemudian menganalisis hasil wawancara untuk membuat *user stories*. *User story* menjelaskan siapa saja yang akan menjadi pengguna sistem, tujuannya dan fitur-fitur yang ada di dalam aplikasi [7].

3. Pembuatan *Product Backlog*

Pada tahapan pembuatan *product backlog* ini, peneliti membuat berdasarkan *user stories* dari analisis yang

dilakukan dan membuat rangkaian alur *Rest API*. *Product backlog* berkembang seiring dengan berkembangnya produk dan lingkungan produk dikembangkan [8].

4. Implementasi

Pada tahap implementasi yaitu pembuatan *Rest API* aplikasi *private coding* berbasis *web* berdasarkan *product backlog* yang sudah dibuat sebelumnya. Pengembangan *Rest API* aplikasi *private coding* menggunakan pendekatan arsitektur *MVC*, yaitu dengan memisahkan lapisan data (*model*), lapisan tampilan (*view*) dan lapisan penghubung antara data dan tampilan (*controller*). Dengan penerapan arsitektur ini diharapkan pengembangan aplikasi dapat bersifat *modular* dan memberikan kemudahan dalam proses *maintenance* kode program serta mendapatkan performa aplikasi yang baik [9].

5. Testing

Pada penelitian ini akan dilakukan dua pengujian yaitu *Black Box Testing*, *User Acceptance Test (UAT)* dan kuisioner pengguna dengan analisis menggunakan pengukuran *Skala Likert (Likert's Summated Rating)*. *Black Box Testing* sebagai metode untuk menguji fungsionalitas aplikasi. Sedangkan *UAT* sebagai metode pengujian kesesuaian antara hasil pengembangan aplikasi dengan kebutuhan pengguna [10]. *Skala Likert* adalah skala yang dapat digunakan untuk membandingkan skor sikap, pendapat dan persepsi seseorang atau sekelompok orang untuk melihat perkembangan atau perubahan sikap sebelum dan sesudah eksperimen [11].

6. Kesimpulan

Pada tahap ini, maka akan ditarik kesimpulan berdasarkan tahapan-tahapan yang telah dilakukan pada tahapan sebelumnya dan peneliti akan memberikan saran untuk dikembangkan lagi oleh peneliti selanjutnya.

3. HASIL DAN PEMBAHASAN

3.1 Analisis dan Perancangan

Berikut ini hasil dan pembahasan rancang bangun *rest API* menggunakan *Express JS* guna mencari mentor pribadi untuk pembelajaran:

1. *User Stories*

Setelah melakukan tahapan wawancara berupa kuisioner kepada masyarakat yang berpotensi menjadi pengguna aplikasi *private coding*, peneliti menganalisis data yang telah dikumpulkan untuk kemudian dibuat beberapa *user stories* yang terdapat di dalam tabel 1.

Tabel 1. *User Stories*

No	Sebagai	Saya Ingin	Sehingga
1	User	Ada verifikasi <i>email</i>	<i>Email</i> saya tidak digunakan oleh orang lain.
2	Admin	Melihat daftar semua mentor yang telah terdaftar di aplikasi	Saya dapat mengelola dan memverifikasi profil mentor dengan lebih mudah.
3	Admin	Melihat daftar	Saya dapat mengelola data

No	Sebagai	Saya Ingin	Sehingga
		semua siswa yang telah terdaftar di aplikasi	siswa dan memastikan validitas informasi yang diberikan.
4	Mentor	Membuka kursus	Saya bisa menawarkan keahlian dan pengetahuan saya kepada para siswa.
5	Admin	Memverifikasi kursus yang akan dibuat mentor	Mentor tidak dapat sembarangan membuka kursus
6	Mentor	Mengubah informasi kursus yang telah saya buat	Saya dapat memberikan detail yang akurat dan terkini kepada para siswa tanpa harus membuat ulang kursus lagi
7	Mentor	Melihat informasi kursus yang telah saya buat	Saya bisa mendapatkan informasi terbaru kursus saya
8	Mentor	Mengatur jadwal ketersediaan mengajar saya	Saya dapat menghindari tabrakan jadwal dan memungkinkan siswa untuk memesan sesi kursus dengan mudah.
9	Mentor	Melihat daftar siswa yang telah memesan kursus saya	Saya dapat mempersiapkan diri dan materi yang sesuai sebelum sesi kursus dimulai.
10	Siswa, Tamu	Melihat daftar mentor dan kursus yang tersedia	Saya dapat memilih mentor yang cocok dan kursus yang sesuai dengan kebutuhan belajar saya.
11	Siswa	Melakukan pemesanan kursus dari mentor pilihan saya	Saya bisa mendapatkan akses untuk belajar dari mentor yang kompeten dan berkualitas.
12	Siswa	Melakukan pembayaran kursus dengan berbagai metode pembayaran yang berbeda	Saya dapat memilih opsi pembayaran yang paling nyaman dan sesuai dengan preferensi saya.
13	Siswa	Menerima invoice transaksi setelah memesan kursus	Saya dapat melihat informasi penting seperti nama mentor, detail kursus, dan tanggal kursus berlangsung.
14	Admin	Memproses pembayaran kursus yang telah dipesan oleh siswa	Saya dapat memastikan transaksi pembayaran berjalan lancar dan memberikan akses kepada siswa untuk memulai kursus.
15	Siswa	Membatalkan pemesanan kursus dengan syarat dan ketentuan tertentu	Jika ada perubahan rencana mendadak, saya bisa membatalkan kursus dengan cara yang mudah dan jelas.
16	Siswa	Memberikan rating dan komentar setelah menyelesaikan kursus	Pengalaman belajar saya dapat membantu mentor dalam meningkatkan kualitas pengajaran dan membantu calon siswa lain untuk memilih mentor yang tepat.
17	Siswa	Melihat pesan dan notifikasi dari mentor atau admin terkait kursus atau transaksi saya	Saya tetap mendapatkan informasi terkini dan dapat merespons pesan dengan tepat waktu.
18	Mentor	Melihat rating dan komentar yang diberikan oleh siswa	Saya dapat mengevaluasi dan meningkatkan kualitas pengajaran saya untuk memberikan pengalaman

No	Sebagai	Saya Ingin	Sehingga
		terhadap kursus saya	belajar yang lebih baik bagi siswa.
19	Mentor	Menerima notifikasi ketika ada pemesanan kursus baru	Saya dapat segera menyiapkan diri dan menyusun materi sesuai dengan kebutuhan siswa yang telah memesan kursus.
20	Siswa	Melihat daftar kursus yang sedang berlangsung	Saya dapat bergabung dalam sesi kursus yang sedang berjalan jika sesuai dengan jadwal dan kebutuhan belajar saya.
21	Siswa	Melihat rekomendasi mentor berdasarkan minat dan keterampilan saya	Saya bisa lebih mudah menemukan mentor yang tepat dan sesuai dengan bidang yang saya inginkan.
22	Siswa	Menerima notifikasi saat kursus yang saya pesan akan segera dimulai	Saya dapat mempersiapkan diri dan mengingatkan diri saya tentang jadwal kursus yang telah saya pesan.
23	Siswa	Melihat riwayat kursus yang telah saya selesaikan	Saya bisa melacak perkembangan belajar saya dan mengingat kembali materi yang telah dipelajari di masa lalu.
24	Mentor	Melihat daftar siswa yang telah menyelesaikan kursus saya	Saya dapat memberikan rekomendasi atau testimonial kepada siswa-siswa tersebut berdasarkan performa dan prestasi belajar mereka.

2. Product Backlog

Berdasarkan gambaran *user stories*, peneliti merangkum fitur aplikasi *private coding* berisi penjelasan *task* atau tugas yang dilakukan dalam pengembangan aplikasi, prioritas fitur dan estimasi *Story Points* sebagai ukuran kompleksitas atau waktu pengerjaan setiap *task*. Dalam penjelasannya, setiap *Task* diberi penilaian prioritas sesuai kepentingannya. Untuk mengukur kompleksitas setiap *Task*, digunakan skala *Story Points*. 1 nilai *Story Points* memerlukan estimasi waktu pengerjaan sekitar 2-8 jam, 2 nilai *Story Points* memerlukan estimasi waktu pengerjaan antara 10-20 jam dan 3 nilai *Story Points* memerlukan estimasi waktu pengerjaan sekitar 21-40 jam. Deskripsi aplikasi disertakan untuk memberikan gambaran tentang tujuan dan fungsi aplikasi. Dengan informasi yang lengkap ini, diharapkan pengembangan aplikasi *private coding* dapat dilakukan dengan terstruktur dan efisien yang dapat dilihat pada tabel 2.

Tabel 2. Product Backlog

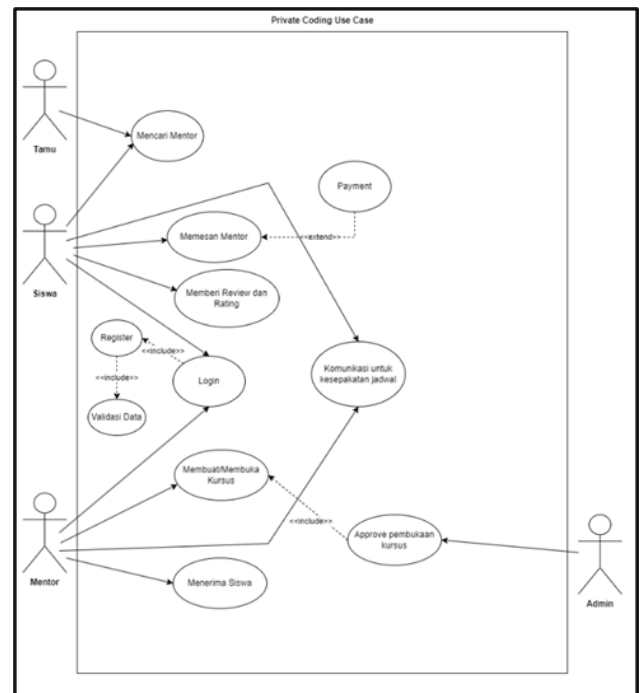
No	Task	Priority	Story Points	Deskripsi
1	Pembuatan <i>User Requirement</i>	Tinggi	1	Menyusun dan mendokumentasikan kebutuhan pengguna secara rinci untuk mengidentifikasi fitur-fitur utama aplikasi
2	Pembuatan <i>Use Case</i>	Tinggi	1	Menggambarkan interaksi antara aktor dan sistem untuk setiap fitur yang

No	Task	Priority	Story Points	Deskripsi
				teridentifikasi dalam <i>user requirement</i> .
3	Pembuatan <i>Class Diagram</i>	Tinggi	1	Mendesain struktur dan hubungan antara kelas-kelas dalam aplikasi untuk memodelkan logika dan relasi data
4	Pembuatan <i>Activity Diagram</i>	Tinggi	1	Menggambarkan alur kerja dari berbagai proses dalam aplikasi untuk memahami bagaimana pengguna berinteraksi dengan sistem.
5	Pembuatan <i>Schema Database</i>	Tinggi	1	Merancang skema <i>database</i> untuk menyimpan informasi/data aplikasi <i>private coding</i> .
6	<i>Migration Table Database</i>	Tinggi	1	Melaksanakan migrasi skema basis data ke dalam sistem basis data untuk memastikan struktur basis data sesuai desain.
7	<i>Setup Redis</i>	Tinggi	1	Mengonfigurasi dan mengimplementasikan Redis sebagai penyimpanan cache untuk meningkatkan kinerja aplikasi.
8	<i>Seeder for dummy data</i>	Rendah	1	Membuat data <i>dummy</i> untuk keperluan pengujian dan pengembangan aplikasi.
9	<i>API Login dan Logout</i>	Tinggi	1	Menerapkan <i>API</i> untuk mengelola proses <i>login</i> dan <i>logout</i> pengguna dengan menggunakan token JWT.
10	<i>API Register</i>	Tinggi	2	Mengimplementasikan <i>API</i> untuk mengizinkan pengguna mendaftar dan membuat akun baru.
11	<i>Verifikasi Email menggunakan OTP</i>	Tinggi	2	Menerapkan mekanisme verifikasi email dengan menggunakan <i>one-time password</i> (OTP).
12	<i>Create Token dan Refresh Token</i>	Tinggi	2	Membuat dan mengimplementasikan token dan <i>refresh</i> token untuk mengelola otentikasi dan otorisasi pengguna.
13	<i>API Profile</i>	Sedang	2	Mengimplementasikan <i>API</i> untuk mengelola profil pengguna.
14	<i>API Course</i>	Tinggi	3	Menerapkan <i>API</i> untuk menampilkan informasi tentang kursus yang tersedia.
15	<i>API buy course</i>	Tinggi	3	Membuat <i>API</i> untuk memproses pembelian kursus oleh siswa.
16	<i>API transactions</i>	Tinggi	2	Terapkan <i>API</i> untuk tampilkan informasi transaksi siswa.

No	Task	Priority	Story Points	Deskripsi
17	<i>API notifications</i>	Sedang	1	Mengimplementasikan <i>API</i> untuk kelola notifikasi yang dikirim oleh aplikasi.
18	Integrasi dengan <i>Payment gateway (midtrans)</i>	Sedang	2	Mengintegrasikan sistem pembayaran dengan midtrans atau <i>gateway</i> pembayaran lainnya.
19	<i>API review course</i>	rendah	2	Menerapkan <i>API</i> untuk memungkinkan siswa memberikan ulasan dan penilaian terhadap kursus yang telah diikuti.

3. Use Case Diagram

Pada tahapan analisis sistem yang akan digunakan untuk mengetahui kebutuhan sistem, akan dibuat sebuah *use case diagram*. *Use case diagram* merupakan sebuah *tools* yang digunakan untuk membuat pemodelan interaksi *user* dengan sistem [12]. *Use case diagram* bisa mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat [13].



Gambar 4. Use Case Diagram Rest API Private Coding

Pada gambar 4 merupakan *use case diagram* rancang bangun *Rest API* untuk mencari mentor pribadi dengan penjelasan sebagai berikut:

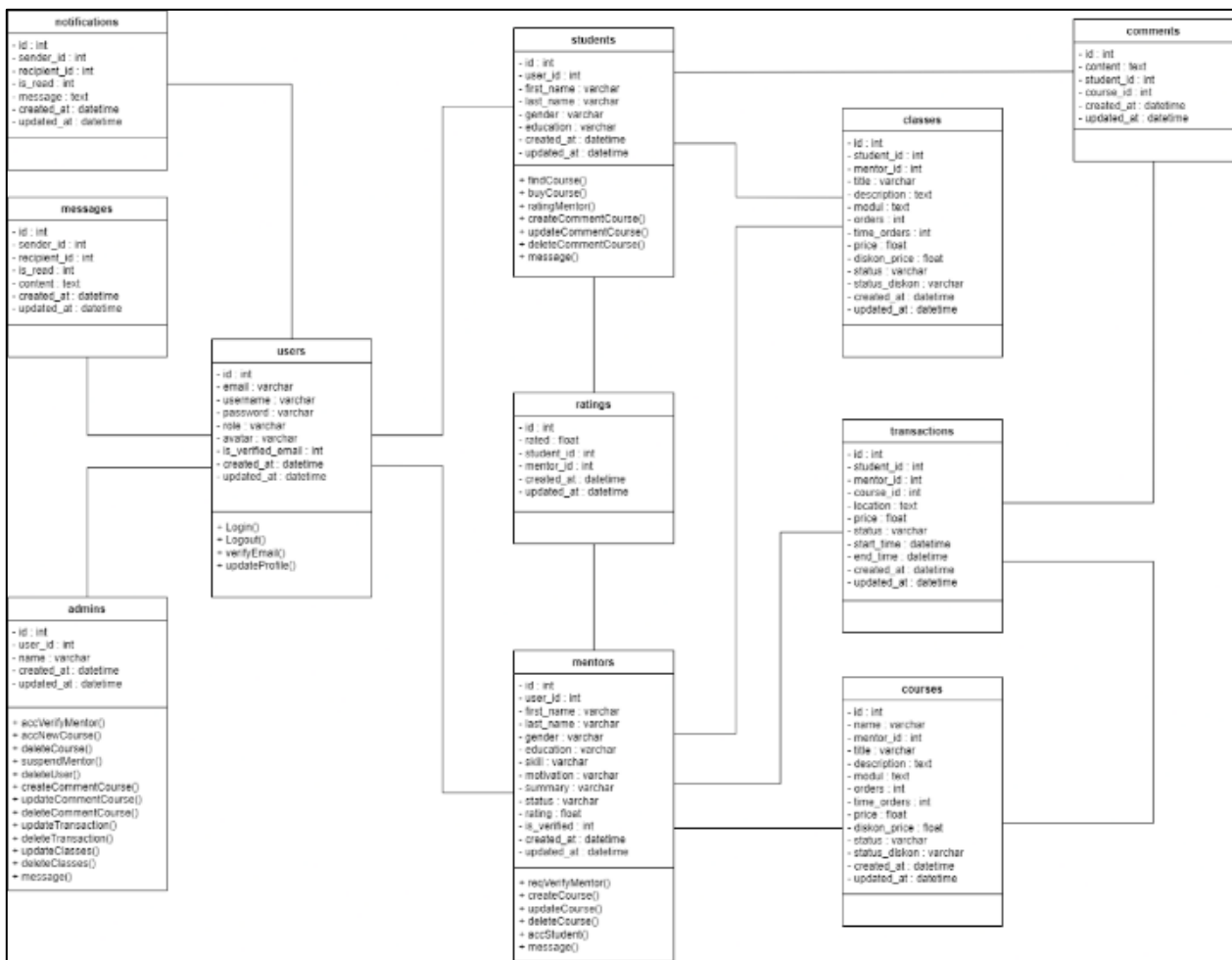
- a. Siswa dapat mencari mentor, berkomunikasi untuk membentuk kesepakatan jadwal bersama dengan mentor, kemudian siswa melakukan pemesanan mentor pemrograman yang dibutuhkan dimana ketika siswa melakukan pemesanan mentor siswa harus melakukan *payment* sesuai ketentuan dan memberikan *review* mentor pemrograman pada aplikasi. Tidak

hanya itu, siswa melakukan *login* tetapi sebelum *login* aplikasi, siswa melakukan registrasi akun dahulu dan validasi data agar akun yang didaftarkan oleh siswa dapat digunakan baik.

- b. Adapun mentor melakukan *login* pada aplikasi *private coding*, tetapi mentor harus registrasi akun dan validasi data terlebih dahulu. Kemudian mentor dapat membuat atau membuka kursus yang sesuai dengan keahliannya dan melakukan komunikasi untuk kesepakatan jadwal bersama siswa. Setelah siswa memesan mentor dan melakukan pembayaran, maka mentor menerima siswa yang membutuhkan bimbingannya untuk mempelajari pemrograman.

4. Class Diagram

Class diagram adalah salah satu jenis diagram dalam pemodelan sistem yang digunakan untuk menggambarkan struktur dan hubungan antar kelas dalam suatu sistem. *Class diagram* memberikan gambaran tingkat tinggi tentang komponen-komponen yang penting dalam sistem dan bagaimana mereka saling berhubungan [14]. *Class diagram* pada aplikasi *private coding* menggambarkan relasi hubungan di antara kelas-kelas, objek, atribut dan operasi pada aplikasi. Gambar 5 berikut ini adalah *class diagram* pada rancang bangun *Rest API* untuk mencari mentor pribadi.



Gambar 5. Class Diagram Rest API Private Coding

Pada gambar 5 terdapat 10 *class* yang memiliki atribut guna menjelaskan bagaimana kualitas yang baik dari suatu kelas tersebut, dan terdapat operasi yang terjadi pada masing-masing *class*. Adapun relasi *class* tersebut digambarkan dengan notasi *multiplicity*. Pada relasi antar *class* digambarkan dengan notasi *multiplicity*, *class students* berhubungan dengan *class* yang disebut sebagai *one-to-many*, yaitu *class students* dapat memiliki banyak relasi yang bersifat *dependent* atau memiliki ketergantungan satu dengan yang lainnya terhadap *class* yang ada. Hal ini

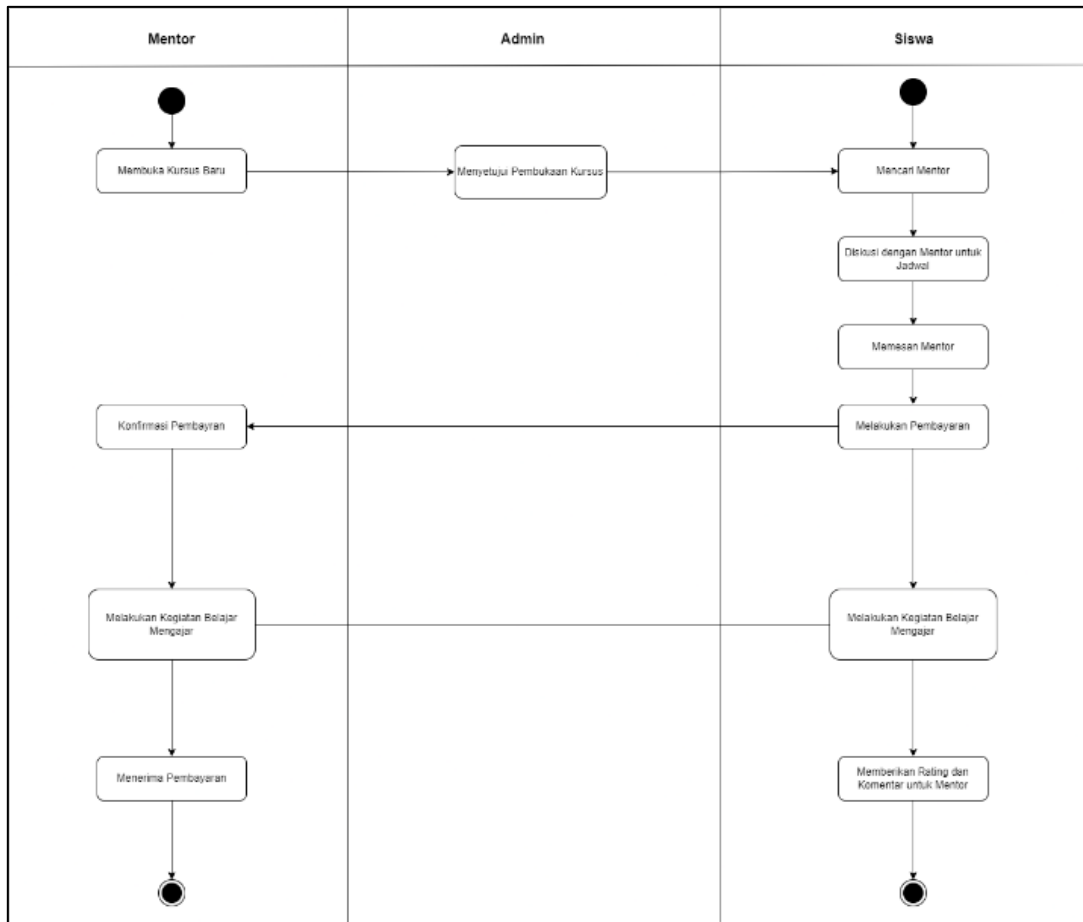
disebabkan karena *class* tersebut begitu membutuhkan data dari *class* yang tersedia untuk mengatur jadwal pembelajaran. Selanjutnya untuk *class* yang tidak memiliki relasi disebut *independent class* yang tidak tergantung pada *class* lain untuk operasi.

5. Activity Diagram

Activity diagram digunakan untuk menggambarkan tentang aktivitas yang terjadi pada sistem. *Activity diagram* menunjukkan langkah-langkah dalam proses kerja dari

sebuah sistem dari pertama sampai akhir [15]. *Activity diagram* juga akan menggambarkan konsep aliran data/kontrol, aksi terstruktur serta dirancang dengan baik

dalam suatu sistem [16]. Berikut ini *activity diagram* rancang bangun *Rest API* menggunakan *Express JS* untuk mencari mentor pribadi pada aplikasi *private coding*.



Gambar 6. Activity Diagram Rest API Private Coding

Pada gambar 6 dapat disimpulkan bahwa terdapat 3 alur proses yang terjadi dalam perancangan *Rest API aplikasi private coding* yaitu mentor, admin dan siswa. Alur aplikasi *private coding* ini, terjadi ketika mentor membuka kursus baru pada aplikasi, setelah itu admin melakukan persetujuan ke kursus yang ingin dibuka dan di saat bersamaan siswa dapat melakukan proses pencarian mentor. Selanjutnya, siswa melakukan diskusi pada aplikasi *private coding* bersama mentor untuk menentukan jadwal pembelajaran dan jika kedua pihak tepat maka siswa akan melakukan pemesanan mentor dan melakukan pembayaran. Setelah pembayaran berhasil, mentor melakukan konfirmasi pembayaran dan memulai kegiatan belajar dan mengajar bersama siswa sesuai dengan waktu yang ditentukan. Jika waktu pembelajaran berjalan, maka mentor akan menerima pembayaran dari siswa dan siswa dapat memberikan *rating* (penilaian) dan komentar sesuai dari pembelajaran yang telah diberikan oleh mentor.

3.2 Implementasi

Pada bagian ini akan membahas implementasi rancang bangun *Rest API* menggunakan *Express JS* guna mencari mentor pribadi dari aplikasi *private coding*. Dengan adanya *Rest API*, pengguna lain tidak dapat secara langsung mengakses *database*, melainkan harus *request* melalui internet, lalu setelah itu dapat mengakses *Rest API* dan kemudian *server Rest API* meresponsnya [17]. Sedangkan *Express JS* adalah satu *web framework* paling populer dan dokumentasinya yang lengkap dan penggunaannya yang cukup mudah, serta dapat membuat dan mengembangkan berbagai produk seperti aplikasi web ataupun *REST API* [18].

Berikut ini implementasi fitur-fitur *Rest API* pada aplikasi *private coding* diantaranya:

1. Fungsi Register Siswa

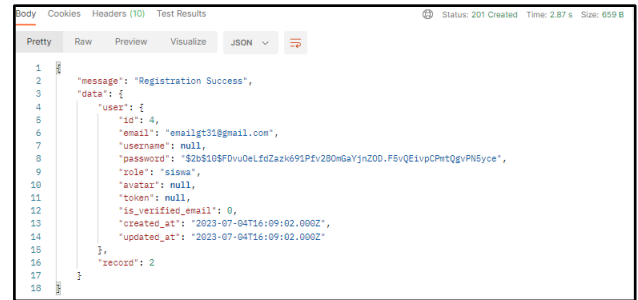
Dalam penerapan fungsi *Rest API* adapun tahapan register siswa yang dibuat untuk registrasi akun siswa. Berikut adalah tampilan register siswa yang dibuat:



Gambar 7. Registrasi Siswa

Pada gambar 7 dapat dijelaskan cara mendaftarkan dengan *role* siswa, *user* harus *merequest* ke `http://localhost:3000/register` dengan *method POST*. Lalu di dalam *body request*nya harus berisi suatu `{“email”, “password”, “role”, “first_name”, “last_name”, “education”, “gender”}`. Lalu jika berhasil akan

menghasilkan *response* seperti tampak pada gambar 8 berikut.



Gambar 8. Registrasi Sukses

Setelah itu, maka di dalam *database table users* akan terisi data-data yang di-*input*, seperti tampak pada gambar 9 dan di dalam *database table students* akan berisi data seperti tampak pada gambar 10 .

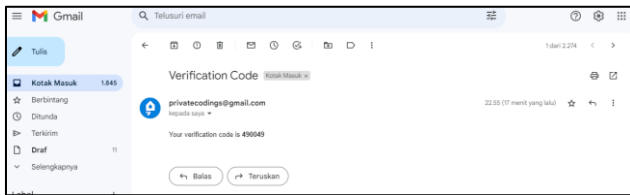
	id	email	username	password	role	avatar	token	is_verified_email	created_at	updated_at
	1	admin@gmail.com	admin	\$2b\$10\$im48pUUIQLSU5s0i2lYqu8G35lv9E56zgV5svdKLW...	admin	NULL	NULL	1	2023-07-04 23:07:32	2023-07-04 23:07:32
	2	siswa@gmail.com	alwan	\$2b\$10\$9MCq8L2toCj5bHhCn8eH5eIMbPW5SnyJ6PRWVidbmaB...	siswa	NULL	NULL	1	2023-07-04 23:07:32	2023-07-04 23:07:32
	3	mentor@gmail.com	zharonk	\$2b\$10\$HdZ6LUioWWYaN58UgDWaujU0Jum2iC4EgSMcXYXm6...	mentor	NULL	NULL	1	2023-07-04 23:07:32	2023-07-04 23:07:32
	4	emailgt31@gmail.com	NULL	\$2b\$10\$FDvuoEoLfdZazk691Pfv28OmGaYjnZOD.F5vQElvpCPm...	siswa	NULL	NULL	0	2023-07-04 23:09:02	2023-07-04 23:09:02

Gambar 9. Data Database Table Users

	id	user_id	first_name	last_name	gender	education	created_at	updated_at
	1	2	Alwan	Dini	Perempuan	Perguruan Tinggi	2023-07-04 23:07:32	2023-07-04 23:07:32
	2	4	john	doe	pria	SD	2023-07-04 23:09:02	2023-07-04 23:09:02

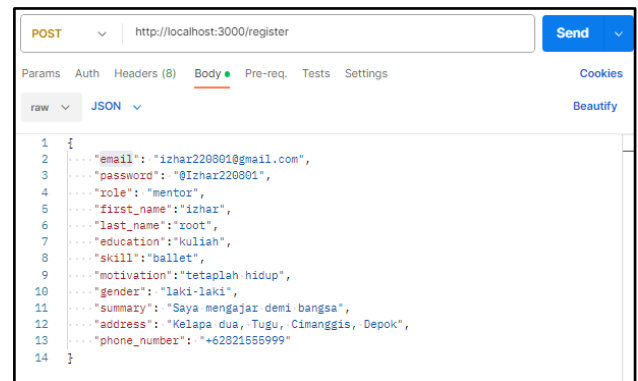
Gambar 10. Data Tabel Students

Kemudian, dari data tersebut akan dikirimkan 6 digit kode *One Time Password (OTP)* pada email yang telah terdaftar pada aplikasi *private coding*, 6 digit kode OTP ini untuk memverifikasikan email akun yang telah terdaftar, agar akun yang terdaftar pada aplikasi aktif dan digunakan, seperti tampak pada gambar 11.



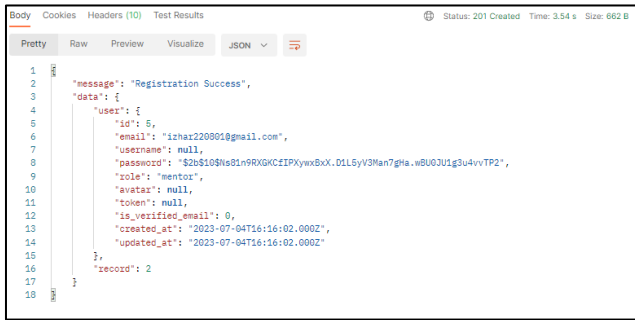
Gambar 11. Kode OTP via E-Mail

2. Fungsi Register Mentor



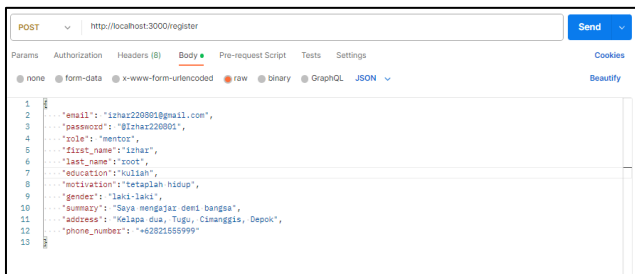
Gambar 12. JSON Register

Pada gambar 12, fungsi tersebut dapat berfungsi untuk mendaftarkan dengan *role* yang tersedia pada aplikasi seperti *role* mentor dan *user* harus melakukan *request* ke `http://localhost:3000/register` dengan *method POST*. Lalu di dalam *body request*nya harus berisi informasi terkait `{“email”, “password”, “role”, “first_name”, “last_name”, “education”, “gender”, “skill”, “motivation”, “summary”, “address”, “phone_number”}` agar informasi yang dihasilkan berhasil. Dan apabila berhasil maka akan menghasilkan *response* berhasil seperti tampak pada gambar 13 berikut ini.



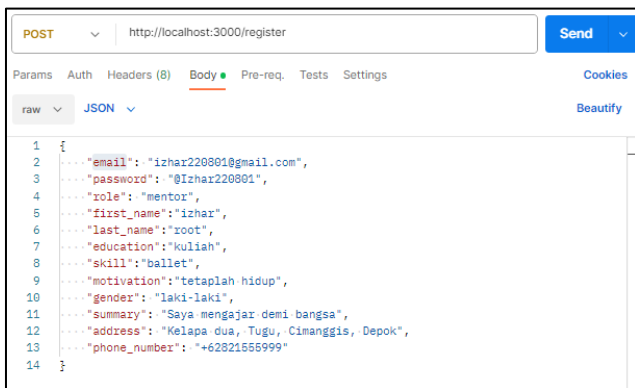
Gambar 13. Response Berhasil

Untuk dapat mendaftarkan dengan *role* mentor, *user* harus *me-request* ke `http://localhost:3000/register` dengan *method* *POST*. Lalu di dalam *body request*nya harus berisi informasi: {"email", "password", "role", "first_name", "last_name", "education", "gender", "skill", "motivation", "summary", "address", "phone_number"}. Pada gambar 14 merupakan tampilan *invalid register* jika *user* ada yang tidak valid dalam melakukan registrasi:



Gambar 14. Invalid Register

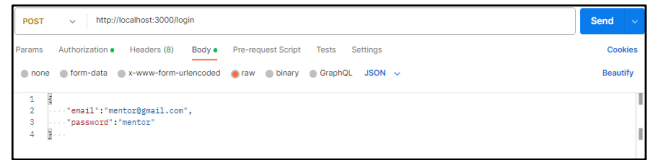
User juga dapat melakukan registrasi dengan *email* yang sudah terdaftar dengan hasil seperti tampak pada gambar 15.



Gambar 15. Registrasi dengan Email

3. Fungsi Login

Untuk dapat *login*, *user* harus *me-request* ke `http://localhost:3000/login` dengan *method* *POST*. Lalu di dalam *body request*nya harus berisi: {"email", "password"}, seperti tampak di gambar 16.



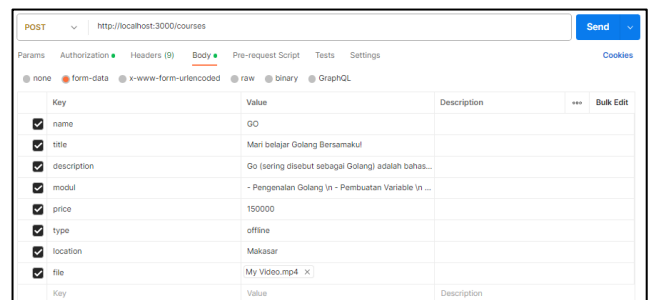
Gambar 16. Login

Setelah *login user* akan mendapatkan response {"token", "refreshToken"}. *Token* disini yang nantinya akan dimasukan ke dalam *Bearer Token* setiap kali *user* ingin *me-request* ke *API* tertentu, supaya aplikasi mengenali bahwa yang meminta adalah *user* dan *token* yang aktif selama 900detik/15menit.

Lalu untuk *refresh token*, *user* diminta kembali *token*nya. Saat *token*nya masa aktif *token* sudah habis dan kembali *refresh token*, maka masa aktifnya akan ditambah selama 86400detik/24jam.

4. Verifikasi Email Akun

Melakukan *request* dengan akun yang belum ter-*verifikasi*. Seperti yang sudah dibahas pada fungsi *register*, bahwa akun yang mendaftar harus untuk *memverifikasi* email akunnya, supaya akunnya dapat aktif.



Gambar 17. Request dengan Akun

Pada gambar 17 menggambarkan bahwa *user* mencoba untuk melakukan *request* akses ke `http://localhost:3000/courses` dengan email yang belum *terverifikasi* pada aplikasi. Oleh karena itu, *user* tersebut akan mendapatkan *response request akun* seperti gambar 18.

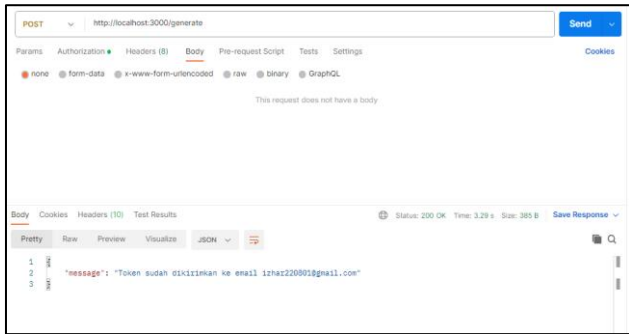


Gambar 18. Response Request Akun

Pada gambar 18 berisi informasi *response request* akun pada aplikasi yang telah dilakukan, maka kemudian akan menampilkan *message* sudah atau belum *terverifikasi*.

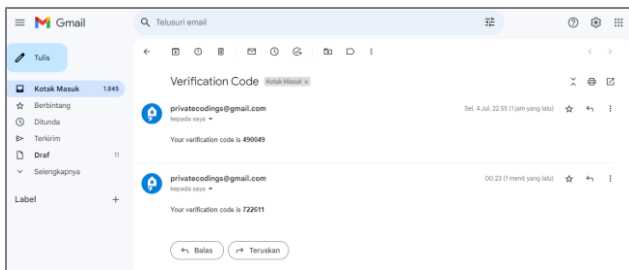
Untuk melakukan *verifikasi* email akun, *user* harus *me-request* ke `http://localhost:3000/verif` dengan *method* *POST*. Lalu di dalam *body request*-nya harus berisi : {"otp"} Untuk *OTP* akan dikirimkan ke *email* dan bisa didapatkan saat *user* berhasil mendaftarkan akun atau *user*

bisa *merequest* ke <http://localhost:3000/generate> dengan *method POST*. Pada gambar 19 merupakan contoh *user* dengan email “*izhar220801@gmail.com*” *me-request* ke <http://localhost:3000/generate> untuk mendapatkan OTP:



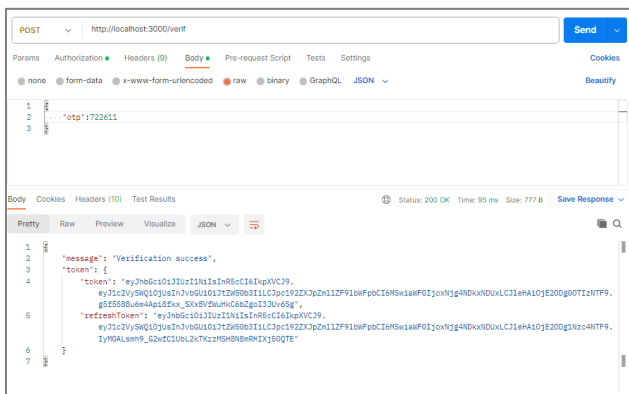
Gambar 19. Verifikasi Email

Setelah itu OTP baru akan dikirimkan ke *email* yang sudah dimasukkan sebelumnya seperti tampak pada gambar 20.



Gambar 20. OTP menerima Email

Setelah itu *user* dapat memasukkan OTP ke dalam *body request* {"otp"}. Jika OTP benar maka akan ditampilkan informasi seperti gambar 21 berikut ini.



Gambar 21. Informasi Verifikasi OTP

3.3 Hasil Pengujian

1. Blackbox Testing

Pengujian *blackbox* memiliki 27 pengujian yang dilakukan oleh *developer*. Jumlah pengujian yang berhasil didapatkan sebanyak 27 keberhasilan, sehingga persentase keberhasilannya yaitu:

- Persentase keberhasilan = (Jumlah skenario pengujian berhasil / Jumlah total skenario pengujian) x 100%
- Persentase keberhasilan = (27 / 27) x 100% = 100%
- Jadi, dapat disimpulkan bahwa sistem telah berhasil menyelesaikan skenario pengujian.

2. Kuesioner

Berikut ini adalah hasil kuesioner yang didapatkan dari responden, dengan hasil kuesioner (*skala likert*) berikut ini:

- Jawaban Sangat Setuju (SS) = 20 x 4 = 80
- Jawaban Setuju (S) = 16 x 3 = 48
- Jawaban Tidak Setuju (TS) = 0 x 2 = 0
- Jawaban Sangat Tidak Setuju (STS) = 0 x 1 = 0
- Total Skor = 128

Rumus nilai maksimum = (Jumlah responden * Nilai tertinggi *likert* * Total pernyataan) Nilai maksimum = 4 * 4 * 9 = 144
 Penyelesaian Akhir Indeks (%) = (total skor / skor max) * 100 indeks (%) = (128 / 144) * 100 = 88,8%.

Hasil tes *skala likert* yang mencapai 88,8% , artinya masuk dalam interval 80-100 dan menunjukkan bahwa responden sangat setuju terhadap topik yang diuji.

4. KESIMPULAN

Berdasarkan hasil pengujian rancang bangun *Rest API* menggunakan *Express JS* guna mencari mentor pribadi untuk pembelajaran maka didapatkan kesimpulan berikut:

- Rancangan dan pengembangan *Rest API private coding* berjalan dengan baik dan efisien, karena terfokus pada penggunaan teknologi *Express JS* sebagai *backend* aplikasi *private coding*.
- Implementasi fungsi pencarian mentor pribadi pada aplikasi *private coding* telah berfungsi dengan baik sehingga pengguna yang ingin mencari mentor pemrograman menjadi lebih mudah. Hal tersebut sesuai dengan rumus nilai maksimum perhitungan *skala likert* yang telah dilakukan oleh peneliti terhadap fungsionalitas aplikasi. Hasil pengujian yang dilakukan untuk fungsi aplikasi memperoleh persentase hasil keberhasilan mencapai 88,8% dengan interval 80-100 yang menunjukkan bahwa responden sangat setuju terhadap topik penelitian. Dan melalui hasil analisis pengujian *skala likert* yang dilakukan, mayoritas pengguna menyatakan bahwa aplikasi *private coding* dapat mempermudah pencarian mentor pemrograman.

Adapun rancang bangun *Rest API* menggunakan *Express JS* guna mencari mentor pribadi untuk pembelajaran masih memiliki beberapa kekurangan, sehingga perlu pengembangan lanjutan. Berikut adalah masukan terkait dengan pengembangan aplikasi *private coding* untuk penelitian yang akan datang, diantaranya:

- Saat ini aplikasi *private coding* masih dalam tahap pengembangan sehingga fungsi keamanan aplikasi belum sepenuhnya baik. Atas dasar tersebut, peneliti

berharap tim dapat memperbaiki fitur dan layanan aplikasi termasuk dengan keamanannya.

2. Saat ini aplikasi *private coding* masih belum memiliki fitur yang dapat merekomendasikan mentor dengan melihat peminatan siswa yang mendaftar. Sehingga dengan demikian peneliti berharap dengan adanya fitur tersebut bisa membantu siswa dalam mendapatkan mentor yang sesuai.

Ucapan Terima Kasih

Ucapan terima kasih diberikan kepada saudara Achmad Izhar, mahasiswa STT Terpadu Nurul Fikri yang telah merancang dan membangun *Rest API* aplikasi *private coding* guna mencari mentor pribadi. Penghargaan yang setinggi-tingginya juga diberikan kepada tim LPPM STT Terpadu Nurul Fikri yang telah memfasilitasi penerbitan jurnal ini.

DAFTAR PUSTAKA

- [1] B. Arif, "Miris, 400 Ribu Sarjana TI Per Tahun Tak Memenuhi Kualifikasi Industri." <https://edukasi.sindonews.com/read/387390/211/miris-400-ribu-sarjana-ti-per-tahun-tak-memenuhi-kualifikasi-industri-1617620642?showpage=all> (accessed Sep. 14, 2022).
- [2] Narenda, "Miris, 400 Ribu Sarjana TI Per Tahun Tak Memenuhi Kualifikasi Industri," *CEO Dicoding*, 2021. <https://edukasi.sindonews.com/read/387390/211/miris-400-ribu-sarjana-ti-per-tahun-tak-memenuhi-kualifikasi-industri-1617620642?showpage=all> (accessed Sep. 14, 2022).
- [3] M. Gilang, "Aplikasi Berbasis Web dan Keunggulannya yang Bisa Anda Ketahui," 2021. <https://www.qubisa.com/article/pengertian-mentoring-mentor-dan-mentee> (accessed Sep. 25, 2022).
- [4] P. Agriza, N. Azizah, M. Asfi, and I. Syafrinal, "Implementasi Model Scrum Pada Sistem Informasi Pembelajaran Diluar Kampus Untuk Skema Wirausaha Kampus Merdeka," *Syntax J. Inf.*, vol. 10, n, pp. 1–12, 2021.
- [5] Z. Imaduddin, H. Saptono, S. T. Fauziah, H. A. Tawakal, and D. Hamzah, "Aplikasi Monitoring Perkembangan Janin (Antenatal Care) dengan Metode Scrum Berbasis Perangkat Mobile," *J. Teknol. Terpadu*, vol. Vol. 5, No, 2019.
- [6] R. W. Imam Haromain, Sirojul Munir, "Sistem Informasi Geografis Pencarian Layanan Vaksin dan PCR Covid-19 Menggunakan Google Maps API dan Jalur Terpendek," *J. Teknol. Terpadu*, vol. Vol. 8, No, pp. 86–93, 2022.
- [7] F. Nadhira, M. I. Wahyuddin, and R. T. K. Sari, "Penerapan Metode Agile Scrum Pada Rancangan SisIAM4," *J. Media Inform. Budidarma*, vol. 6, no. 1, p. 560, 2022, doi: 10.30865/mib.v6i1.3525.
- [8] S. N. Rafianto, Dimas, "Penerapan Metode Scrum Pada Pembuatan User Experience Landing Page Sistem Informasi Lentera," *J. Sist. Inf. dan Sains Teknol*, vol. 3, no, pp. 1–14, 2021.
- [9] R. W. Sirojul Munir, Ahmad Rio Adriansyah, Radon Dhelika, "Implementasi Arsitektur Aplikasi MVC pada Perancangan Aplikasi Bank Sampah Berbasis Web," *J. Teknol. Terpadu*, vol. Vol. 4, No, 2018.
- [10] D. A. P. Ammar Musthofa Firdaus, "Aplikasi Pencari Tempat Magang Berbasis Android Menggunakan Metode Agile Scrum," *J. Inform. UPGRIS*, vol. Vo. 8, No., 2022.
- [11] S. T. F. Sirojul Munir, Anifatul Aufah, Kuati Septiani, "Analisis dan Rancang Bangun Prototype Web Market Place UMKM Juara," *J. Teknol. Terpadu*, vol. Vol. 6, No, pp. 66–71, 2020.
- [12] L. Setiyani, "Desain Sistem: Use Case Diagram Pendahuluan," in *Prosiding Seminar Nasional: Inovasi & Adopsi Teknologi*, 2021, pp. 246–260.
- [13] B. A. Rahmatul Zakia, Tifanny Nabarian, "Rancang Bangun Antarmuka Berbasis Website Design Method(Wdm) untuk Toko Baju Online," *J. Inform.*, vol. Vol. 9No., pp. 24–33, 2023.
- [14] B. A. H. Muchamad Hasan Basri, "Perancangan Sistem Informasi Kepegawaian Berbasis Web Pada Suara Merdeka Network," 2023.
- [15] N. W. A.-H. Abdul Aziz Kamil, "SISTEM INFORMASI AKADEMIK SMPN 3 TELUK KUANTAN," *J. Perencanaan, Sains, Teknol. dan Komput.*, vol. Vol. 6, No, pp. 79–87, 2023.
- [16] M. W. Tia Arianti, Amal Fa'izi, Safri Adam, "Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram UML (Unified Modelling Language)," *J. Ilm. Komput. Terap. dan Inf.*, vol. Vol. 1, No, 2022.
- [17] Baharudin, "Implementasi Web Service dengan Metode Rest API untuk Integrasi Data Covid 19 di Sulawesi Selatan," *J. Sintaks Log.*, vol. Vol. 2, No, 2022.
- [18] A. Firdaus, "Rancang Bangun Sistem Informasi Perpustakaan menggunakan Web Service pada Jurusan Teknik Komputer Polstri," *J. Informanika*, vol. Vol.5, No., 2019.



PENYUSUNAN DOKUMENTASI SISTEM PADA PERANCANGAN WEB BACKOFFICE APLIKASI MOBILE TELEMEDICINE

Laily Maghfira¹, Esi Putri Silmina²

^{1,2} Teknologi Informasi, Universitas 'Aisyiyah Yogyakarta
Kabupaten Sleman, Daerah Istimewa Yogyakarta. Indonesia 55592

lailymaghfira@gmail.com, esiputrisilmina@unisayogya.ac.id

Abstract

Documentation is a means of conveying information about software. A computer program cannot be considered software without documentation of the software. The compiled documentation can provide many benefits for developers. Proper documentation can shorten the time required to design, create, test, and utilize the software. Designing the back-office web for this mobile telemedicine application requires a formal and structured system design process and meets standards regarding software design. This can be achieved by compiling system documentation as a first step in the ongoing system development. Based on these problems, system documentation is needed in the backoffice web design process for mobile telemedicine applications from the beginning to the end of the design process. The research method used in this research will focus on preparing documentation for designing the back-office web system for mobile telemedicine applications. The application development model used is the waterfall model. The stages of preparing this system documentation are explaining the project background, user persona, user flow, wireframe, prototype, testing, and developing design. The testing process uses the black box testing method. The document preparation process adapts to the project being undertaken. This research results in system documentation, which the developer can later use on duty for further development purposes.

Keywords: Backoffice, Drafting, Software System Documentation, Telemedicine, Web Design

Abstrak

Dokumentasi adalah sarana untuk menyampaikan informasi tentang sebuah perangkat lunak. Suatu program komputer belum dapat dianggap sebagai perangkat lunak tanpa adanya dokumentasi dari perangkat lunak tersebut. Dokumentasi yang disusun dapat memberikan banyak manfaat bagi para *developer*. Penyusunan dokumentasi yang tepat dapat mempersingkat waktu yang diperlukan untuk melakukan perancangan, pembuatan, pengujian dan pemanfaatan sebuah perangkat lunak. Perancangan *web backoffice* aplikasi *mobile telemedicine* ini memerlukan proses perancangan sistem secara formal dan terstruktur dan memenuhi standar mengenai perancangan perangkat lunak (*software*). Hal ini dapat dicapai dengan menyusun dokumentasi sistem sebagai langkah awal dari pengembangan sistem yang akan berlangsung. Berdasarkan masalah tersebut, maka dibutuhkan sebuah dokumentasi sistem dalam proses perancangan *web backoffice* aplikasi *mobile telemedicine* sejak awal hingga akhir proses perancangan. Metode penelitian yang dilakukan adalah penelitian ini akan terfokus dalam penyusunan dokumentasi perancangan sistem *web backoffice* aplikasi *mobile telemedicine*. Model pengembangan aplikasi yang digunakan adalah model *waterfall*. Tahapan dari penyusunan dokumentasi sistem ini, yaitu dengan menjelaskan *background project*, *user persona*, *user flow*, *wireframe*, *prototype*, *testing* dan *develop design*. Proses *testing* menggunakan metode *black box testing*. Proses penyusunan dokumen menyesuaikan dengan proyek yang dikerjakan. Hasil dari penelitian ini adalah sebuah dokumentasi sistem yang nantinya dapat digunakan oleh *developer* yang bertugas untuk keperluan pengembangan selanjutnya.

Kata kunci: Backoffice, Dokumentasi Sistem Perangkat Lunak, Penyusunan, Perancangan Web, Telemedicine

1. PENDAHULUAN

Teknologi informasi saat ini berkembang sangat pesat sesuai dengan perubahan karakteristik masyarakat *modern* yang memiliki mobilitas yang tinggi. Hal ini dapat terlihat dari penggunaan teknologi informasi secara menyeluruh

dalam segala aspek kehidupan. Salah satu aspek yang terkena dampak yaitu dalam hal akses layanan kesehatan. Penggunaan teknologi informasi dalam di bidang kesehatan yang sudah cukup sering ditemui yaitu aplikasi *telemedicine* sebagai media pelayanan medis jarak jauh.

Menurut survei Katadata Insight Center (KIC), layanan *telemedicine* & fasilitas kesehatan (faskes) yang paling banyak digunakan di Indonesia sejak pandemi COVID-19, yaitu Halodoc dengan persentase sebanyak 46,5%, *Telemedicine* RS/Klinik sebanyak 41,8%, Alodokter sebanyak 35,7%, Konsul *online* dokter sebanyak 20,3%, KlikDokter sebanyak 15,5%, isoman.kemkes.go.id sebanyak 10,2%, Good Doctor sebanyak 5,4%, LinkSehat sebanyak 4,4%, Lekasehat sebanyak 2% dan Lainnya sebanyak 1,1% [1].

Christina Rawis, Stanley D. S. Karouw, Sherwin R. U. A. Sompie dari Universitas Sam Ratulangi, Manado, Indonesia melakukan penelitian yang berfokus pada pembuatan dokumentasi SRS berdasarkan sistem informasi akademik yang sudah ada dengan menggunakan gabungan dari 3 model *template* SRS (IEEE, ACM, dan KOMINFO). Hasil penelitian ini adalah sebuah dokumentasi *Software Requirement Specification* Sistem Informasi Akademik UNSRAT dengan menggunakan gabungan 3 *template* [2].

Richi Andrew Serafino dan Meme Susilowati dari Universitas Machung, Malang, Indonesia melakukan penelitian yang berfokus pada pembuatan dokumen *Software Requirement Specification* (SRS) Perancangan Sistem Informasi *Marketing* dengan standar ISO/IEC/IEEE 29148:2018. Metode yang digunakan dalam penelitian ini adalah 3 *Major Phases* untuk melakukan tahapan analisis dan desain. Hasil dari penelitian ini berupa dokumen SRS yang berisi tentang gambaran kebutuhan sistem pada perusahaan tersebut [3].

Penelitian ini berbeda dengan penelitian sebelumnya. Hal ini dapat terlihat pada lokasi penelitian dan hasil penelitian yang dilakukan berupa struktur penulisan dokumentasi sistem pada *web backoffice*. Institusi XYZ telah mengembangkan sebuah aplikasi *mobile telemedicine* yang bertujuan untuk membantu mahasiswa kedokteran tingkat akhir dalam proses pembelajaran sehingga mampu beradaptasi dengan teknologi informasi khususnya dibidang kesehatan. Aplikasi *mobile* tersebut belum memiliki aplikasi *backoffice* untuk melakukan monitoring data, sehingga diperlukannya sebuah perancangan aplikasi *backoffice* dalam bentuk *web*. Proses perancangan *web backoffice* aplikasi *mobile telemedicine* yang sedang berlangsung ini belum terdapat dokumentasi sistem yang menggambarkan tahapan dari proses perancangan.

Berdasarkan latar belakang yang telah dijelaskan, penulisan penelitian ini bertujuan untuk menyusun sebuah dokumentasi sistem berdasarkan pada proyek yang sedang berlangsung. Dokumentasi tersebut akan menyajikan informasi-informasi sejak awal hingga akhir proses perancangan. Rumusan masalah penelitian berupa bagaimana menyajikan informasi terkait proses perancangan dalam bentuk dokumen yang terstruktur agar dapat membantu *developer* untuk pengembangan selanjutnya. Prosedur penelitian dilakukan melalui beberapa

tahap dengan menjelaskan setiap tahapannya (*background project, user persona, user flow, wireframe, prototype, testing dan develop design*). Berdasarkan penjelasan sebelumnya, maka penelitian ini diberi judul “Penyusunan Dokumentasi Sistem Pada Perancangan *Web Backoffice* Aplikasi *Mobile Telemedicine*”

Sistem Informasi

Maydianto dan Ridho berpendapat bahwa sistem informasi merupakan kerangka terorganisir yang mencakup perangkat lunak, perangkat keras, prosedur, data, dan orang-orang yang bekerja bersama untuk mengumpulkan, menyimpan, mengelola, mengolah, dan menyampaikan informasi yang relevan dan bermanfaat kepada pengguna dalam sebuah organisasi [4].

Telemedicine

Waller dan Stotler berpendapat bahwa definisi *telemedicine* merupakan perpaduan antara kata *medicine* dan *tele*, yang berarti seni dan ilmu pemeliharaan kesehatan, serta pencegahan dan penyembuhan penyakit yang dilakukan melalui jarak jauh [5].

Aplikasi *Backoffice*

Menurut Saputra, Prasetyo, dan Khairudin, Aplikasi *backoffice* merupakan salah satu bagian penting dari sebuah bisnis ataupun produk. Aplikasi *backoffice* tidak berjalan sendiri, biasanya aplikasi ini memiliki suatu objek untuk di monitor. *User* yang akan menggunakan *backoffice* nantinya adalah operator/admin yang sudah ditunjuk oleh perusahaan/pihak berwenang untuk memonitoring maupun membuat laporan dari hasil kinerja sebuah produk/jasa. Pembuatan aplikasi *backoffice* bertujuan untuk membantu memonitoring kinerja dari aplikasi utamanya (aplikasi *mobile*) dimana aplikasi utamanya digunakan oleh *end user* [6].

Dokumentasi Perangkat Lunak

Dokumentasi adalah sebuah artefak yang disusun dengan tujuan untuk menyampaikan informasi lengkap dan akurat tentang sistem perangkat lunak yang berjalan. Penyusunan dokumentasi yang baik dapat dilihat dari hasil penulisan, apakah dokumentasi yang disusun telah lengkap dan akurat atau belum. Hasil dokumentasi yang bagus dapat memberikan manfaat bagi para *developer*. Sebaliknya, dokumentasi yang buruk sejak awal proses perancangan perangkat lunak (*software*) bisa jadi akan mempengaruhi baik atau tidaknya kualitas *software* yang dibuat.

Perancangan

Menurut N. Rizqya, perancangan merupakan pemilihan proses maupun data dalam proses pembuatan sistem yang baru. Tujuan perancangan sistem ialah untuk menjelaskan gambaran rancangan atau *blue print* secara lengkap guna menjadi bahan panduan pengembangan dan *developer* yang

tentunya telah disesuaikan dengan perangkat lunak yang ada [7].

Pengujian Sistem

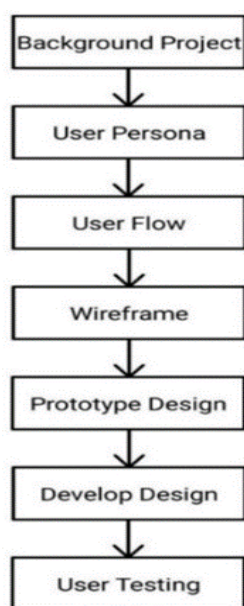
Pengujian sistem dilakukan bertujuan untuk melakukan evaluasi terhadap *software* serta menentukan apakah *software* yang dirancang atau dikembangkan sudah sesuai dengan fungsionalitas yang diharapkan [8].

2. METODE PENELITIAN

Penelitian ini akan terfokus dalam penyusunan dokumentasi perancangan sistem *web backoffice* aplikasi *mobile telemedicine*. Prosedur penyusunan dokumentasi sistem akan menyesuaikan dengan proses perancangan.

2.1 Tahapan Penelitian

Terdapat 7 tahapan dalam proses penelitian diantaranya: (1) tahapan *background project* yang dilakukan dengan menjelaskan tujuan perancangan *web backoffice* aplikasi *mobile telemedicine*; (2) tahapan *user persona* yang dilakukan dengan menjelaskan siapa pengguna akhir web; (3) tahapan *user flow* yang dilakukan dengan menjelaskan alur pengguna saat akan menggunakan produk (*web*) yang terbagi menjadi beberapa bagian; (4) tahapan *wireframe* yang dilakukan dengan menyajikan konsep desain dan fungsi dasar; (5) tahapan *prototype design* yang dilakukan dengan menyajikan tampilan sistem yang telah diubah kedalam sistem yang bekerja (*working system*) dan memungkinkan adanya interaksi antara pengembang sistem dengan pengguna sistem; (6) tahapan *user testing* yang dilakukan dengan menyajikan informasi pengujian sistem dengan menggunakan metode *black box testing*; (7) *develop design* yang dilakukan dengan menyajikan informasi terkait *environment* dalam perancangan *web backoffice* aplikasi *mobile telemedicine*. Detail dari tahapan yang telah dijelaskan dapat terlihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

2.2 Model Pengembangan Aplikasi

Model pengembangan yang digunakan pada proses perancangan *web backoffice* adalah *waterfall*. Model *waterfall* memiliki 5 tahapan diantaranya: (1) tahapan *requirement analysis and definition* dilakukan dengan menganalisis kebutuhan *user* serta masalah yang dihadapi; (2) tahapan *system and software design* dilakukan dengan implementasi spesifikasi sistem yang telah diperoleh pada tahap 1 ke dalam bentuk desain *interface* pengembangan sistem; (3) tahapan *implementation and unit testing* dilakukan dengan mengimplementasikan hasil rancangan sistem ke dalam rangkaian program dengan bahasa pemrograman yang sudah ditentukan; (4) tahapan *integration and unit testing* dilakukan dengan memeriksa dan menguji sistem untuk mengetahui apakah terdapat *error* pada sistem tersebut [9].

2.3 Metode Pengujian

Pada fase *user testing* metode pengujian yang digunakan untuk *web backoffice* aplikasi *mobile telemedicine* institusi XYZ adalah *Black Box Testing*. *Black Box Testing* merupakan sebuah metode yang digunakan bertujuan untuk menguji sebuah *software* tanpa harus memperhatikan detail *software* tersebut [10].

Pengujian *Black Box* ialah kondisi pengujian perangkat lunak (*software*) yang berfokus pada spesifikasi fungsionalnya, penguji akan mendefinisikan kondisi dari input dan spesifikasi fungsional perangkat lunak [11].

2.4 Dokumentasi Requirement Engineering

Requirements engineering merupakan cabang dari *software engineering* yang berkaitan dengan tujuan tujuan yang diinginkan, fungsi, dan kendala pada sistem perangkat lunak [12].

Requirements engineering adalah serangkaian aktivitas terstruktur yang membantu *developer* memahami masalah yang coba dipecahkan serta mendokumentasikan spesifikasi sistem untuk para *stakeholders* dan *engineers* yang terlibat dalam pengembangan sistem. Beberapa aktivitas yang merupakn dasar dari semua proses RE (*requirements engineering*) adalah (1) elisitasi; (2) analisis; (3) validasi; (4) negosiasi; (5) dokumentasi dan (6) manajemen. Hasil dari proses RE adalah sebuah dokumen persyaratan yang mendefinisikan apa yang akan diimplementasikan [13].

3. HASIL DAN PEMBAHASAN

3.1 Background Project

Perancangan *web backoffice* ini bertujuan untuk mengatasi masalah berupa dibutuhkannya sebuah *software* yang dapat membantu kinerja dari aplikasi *mobile telemedicine* yang sudah dibuat/dirancang sebelumnya. *Web* ini memiliki fitur-fitur seperti: *login*, *register*, *dashboard*, *pasien*, *konsultasi*, *dokter*, *spesialisasi*, *activity log* dan *logout*. Harapannya *web* ini nanti dapat digunakan agar dapat mempermudah

admin dalam hal mengelola data admin, pasien, dokter dan data lainnya.

3.2 User Persona

Menurut Szerovay, *user persona* dibuat bertujuan untuk mewakili sekelompok pengguna penting yang mencerminkan pola berdasarkan perilaku, tujuan, dan sikap [14].

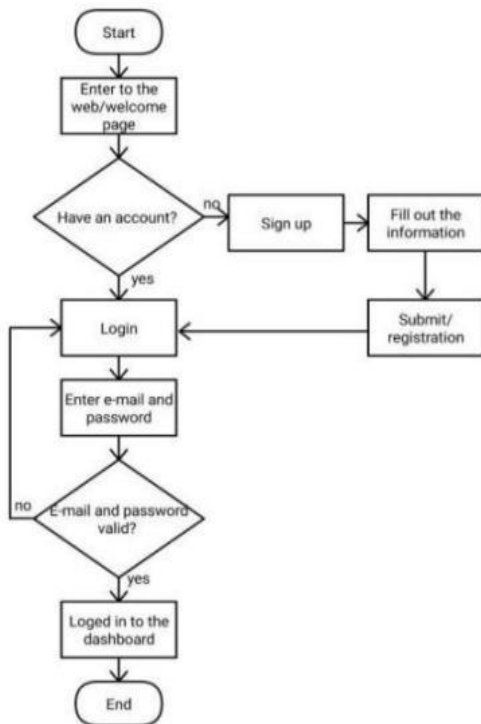
Tabel 1 menjelaskan tentang *user persona* bertujuan menggambarkan *user type* dan *user expectation* dari pengguna *web*.

Tabel 1. Tabel *User Persona*

<i>User type</i>	<i>User Expectation</i>
Admin	<i>User</i> admin dapat memiliki hak akses untuk mengelola seluruh data yang ada seperti: melihat data konsultasi pasien, spesialisasi dokter, <i>log activity</i> , pasien, mengubah data pasien, serta melihat/ menambah/ mengubah/menghapus data dokter.

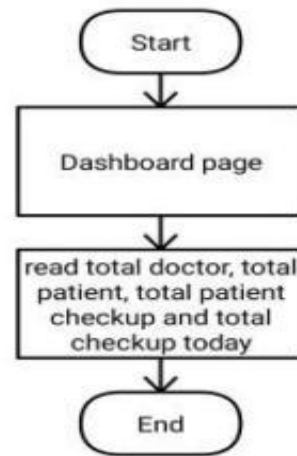
3.3 User flow

Menurut Maulana, *user flow* merupakan urutan penyelesaian tugas yang dilakukan oleh pengguna pada sebuah aplikasi. merupakan alur yang dilalui oleh pengguna [15]. Alur pengguna (*user flow*) biasanya divisualisasikan dalam bentuk diagram *flow chart* guna mempermudah setiap proses yang dilalui oleh *user* saat berinteraksi dengan sistem [16]. Berikut beberapa *user flow* dari *web backoffice* yang ditunjukkan pada Gambar 2 hingga Gambar 9.



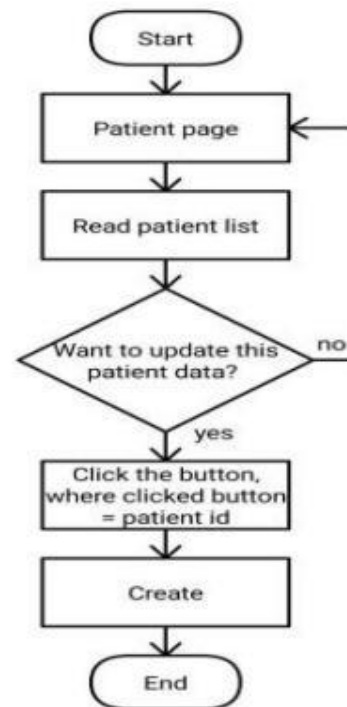
Gambar 2. *User Flow Login dan Register*

Gambar 2 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* akan melakukan *login* dan *register*.



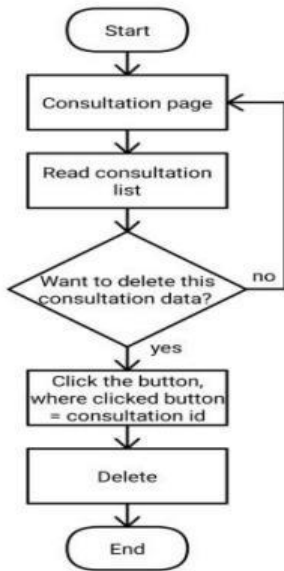
Gambar 3. *User Flow Dashboard*

Gambar 3 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* berhasil melakukan *login* dan *register* yang kemudian *user* akan diarahkan menuju ke halaman *dashboard*.



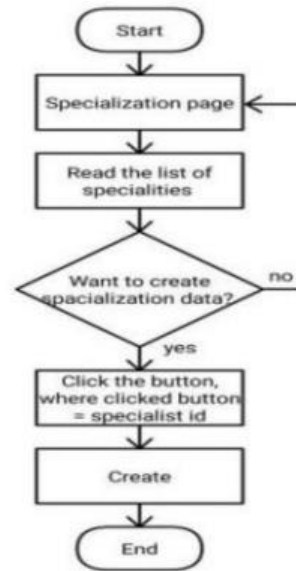
Gambar 4. *User Flow Pasien*

Gambar 4 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* ingin beralih menuju menu/halaman pasien serta melakukan *update* status pasien (*active* atau *suspend*).



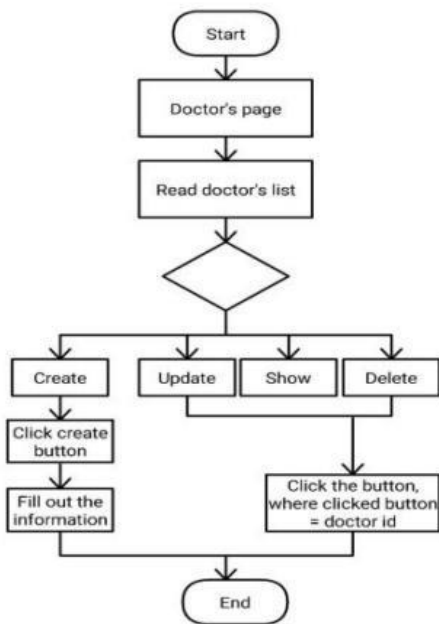
Gambar 5. User Flow Konsultasi Pasien

Gambar 5 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* ingin beralih menuju menu/halaman konsultasi pasien serta melakukan hapus data konsultasi yang ingin dihapus.



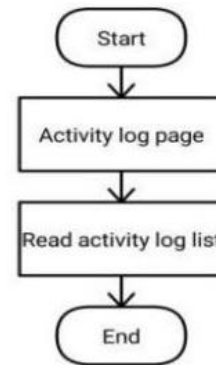
Gambar 7. User Flow Spesialisasi

Gambar 7 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* ingin beralih menuju menu/halaman spesialisasi serta melakukan tambah data spesialisasi dokter.



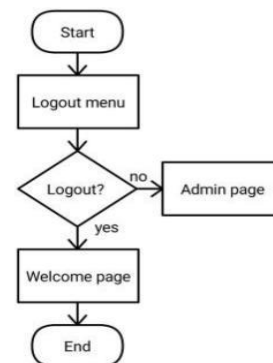
Gambar 6. User Flow Dokter

Gambar 6 yang diperlihatkan di atas menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* ingin beralih menuju menu/halaman dokter serta melakukan tambah, *update*, lihat dan hapus data dokter yang diinginkan.



Gambar 8. User Flow Log Activity

Gambar 8 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* ingin beralih menuju menu/halaman *Log Activity* untuk melihat aktivitas apa saja yang telah dilakukan.



Gambar 9. User Flow Logout

Gambar 9 menjelaskan bagaimana alur pengguna (*user flow*) dari proses ketika *user* ingin keluar dari *web* yang kemudian beralih kembali menuju halaman *welcome*.

3.4 Wireframe

Wireframe merupakan suatu konsep awal dalam pengembangan desain *website* maupun aplikasi *mobile* yang lebih banyak menentukan tata letak yang tepat untuk posisi tombol, fitur, dan konten dari sebuah aplikasi [17].

Menurut Segara, *wireframing* merupakan langkah penting dalam proses desain atau perancangan sebuah media digital (*screen design process*). *Wireframe* diterapkan dengan tujuan mempermudah penyusunan sebuah konten dan *user experience* yang akan diterapkan [18]. Berikut merupakan tampilan *wireframe* dari *web backoffice* yang ditujukan pada Gambar 10 hingga Gambar 25.



Gambar 10. Halaman Welcome

Gambar 10 memvisualisasikan rancangan *wireframe* halaman *welcome*.



Gambar 11. Halaman Login

Gambar 11 memvisualisasikan rancangan *wireframe* halaman *login*.



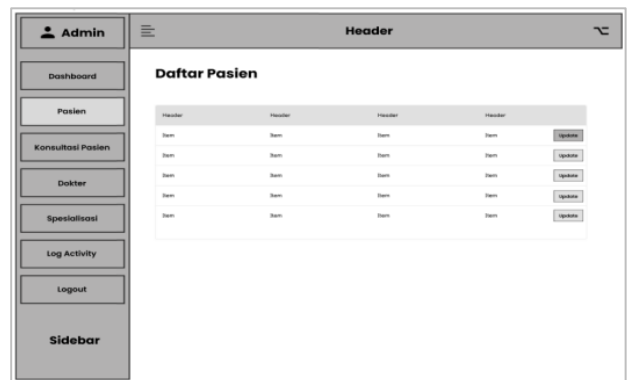
Gambar 12. Halaman Register

Gambar 12 yang diperlihatkan di atas memvisualisasikan rancangan *wireframe* halaman *register*.



Gambar 13. Halaman Dashboard

Gambar 13 memvisualisasikan rancangan *wireframe* halaman *dashboard*.



Gambar 14. Halaman Pasien

Gambar 14 memvisualisasikan rancangan *wireframe* halaman pasien.



Gambar 15. Form update Pasien

Gambar 15 yang diperlihatkan di atas memvisualisasikan rancangan *wireframe* halaman pasien pada bagian *form update* status pasien.



Gambar 16. Halaman Konsultasi Pasien

Gambar 16 memvisualisasikan rancangan *wireframe* halaman konsultasi pasien.



Gambar 17. Form Hapus Konsultasi Pasien

Gambar 17 memvisualisasikan rancangan *wireframe* halaman konsultasi pasien pada bagian *form* hapus data konsultasi.



Gambar 18. Halaman Dokter

Gambar 18 memvisualisasikan rancangan *wireframe* halaman dokter.



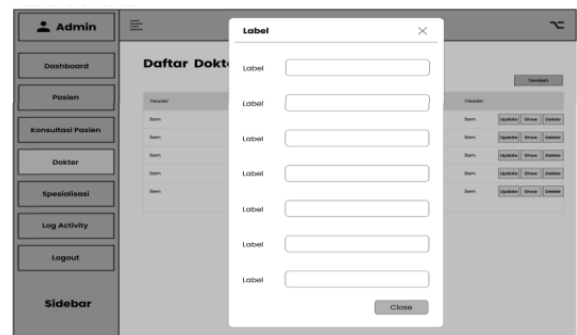
Gambar 19. Form Tambah Dokter

Gambar 19 memvisualisasikan rancangan *wireframe* halaman dokter pada bagian *form* tambah dokter.



Gambar 20. Form Update Dokter

Gambar 20 memvisualisasikan rancangan *wireframe* halaman dokter pada bagian *form update* dokter.



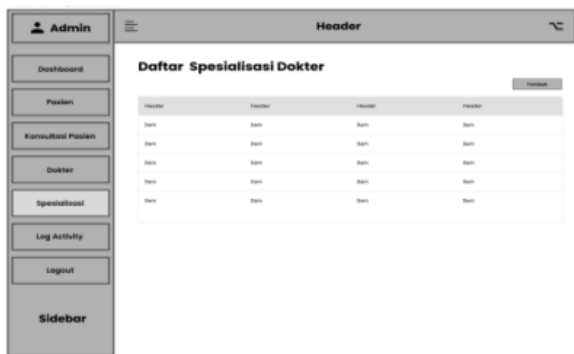
Gambar 21. Form Lihat Dokter

Gambar 21 memvisualisasikan rancangan *wireframe* halaman *welcome*.



Gambar 22. Form Hapus Dokter

Gambar 22 memvisualisasikan rancangan *wireframe* halaman dokter pada bagian *form* hapus dokter.



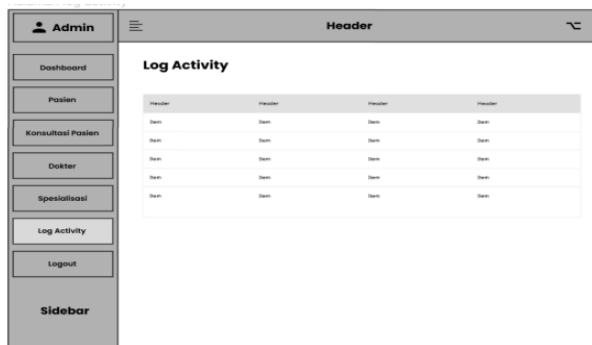
Gambar 23. Halaman Spesialisasi

Gambar 23 yang diperlihatkan diatas memvisualisasikan rancangan *wireframe* halaman spesialisasi.



Gambar 24. Form Tambah Spesialisasi

Gambar 24 memvisualisasikan rancangan *wireframe* halaman spesialisasi pada bagian *form* tambah spesialisasi.



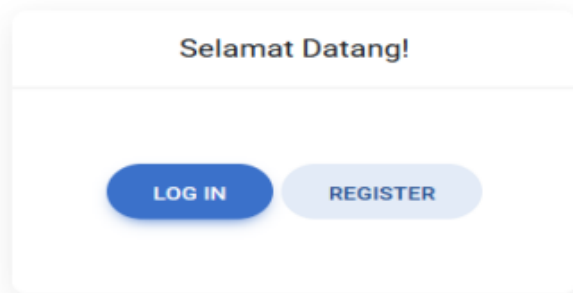
Gambar 25. Halaman Log Activity

Gambar 25 memvisualisasikan rancangan *wireframe* halaman *log activity*.

3.5 Prototype Design

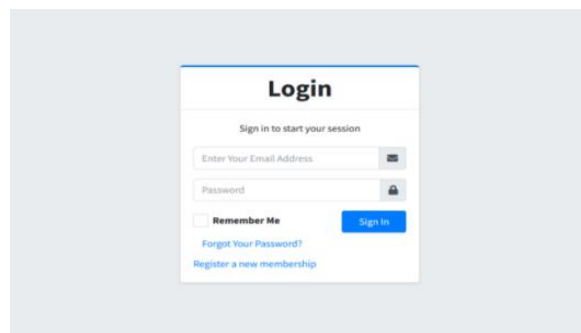
Proses menghasilkan sebuah *prototype* dikenal sebagai *prototyping*. *Prototyping* merupakan proses perancangan *prototype* yang merupakan model dari sebuah *prototype* produk yang mungkin belum terdapat semua fitur produk yang sebenarnya, tetapi sudah memiliki beberapa fitur utama dari produk sebenarnya dan biasa digunakan untuk keperluan pengujian atau bahan uji coba sebelum melanjutkan ke fase pembuatan produk asli [19]. Adanya *prototyping* ini, *developer* maupun *user* dapat saling

berinteraksi selama proses perancangan atau pembuatan perangkat lunak. Berikut merupakan tampilan *prototype* dari *web backoffice* yang ditujukan pada Gambar 26 hingga Gambar 41.



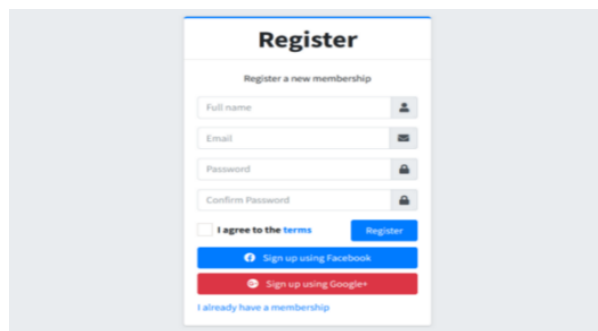
Gambar 26. Halaman Welcome

Gambar 26 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman *welcome*.



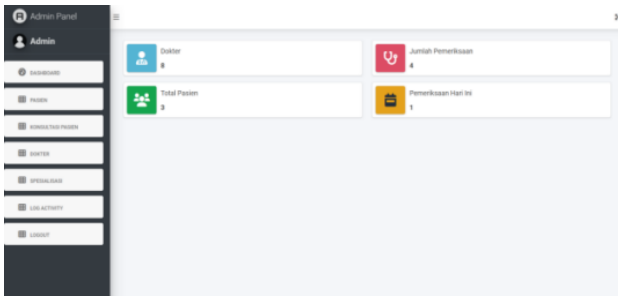
Gambar 27. Halaman Login

Gambar 27 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman *login*.



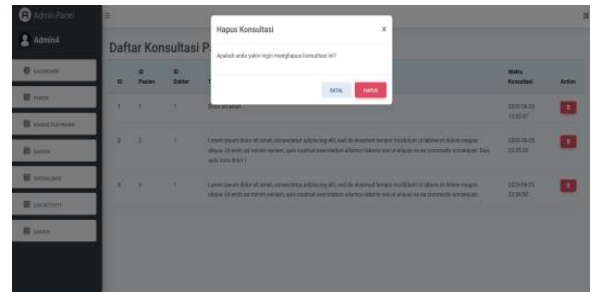
Gambar 28. Halaman Register

Gambar 28 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman *register*.



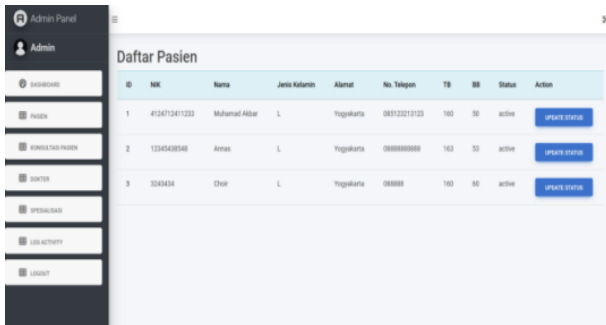
Gambar 29. Halaman Dashboard

Gambar 29 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman *dashboard*.



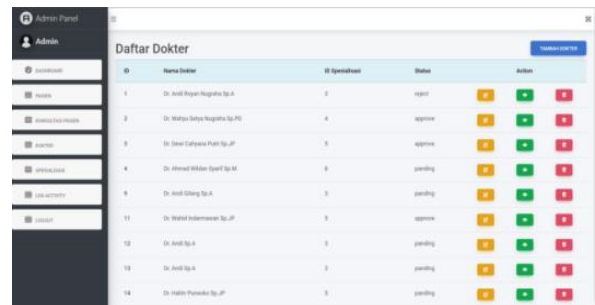
Gambar 33. Form Hapus Konsultasi Pasien

Gambar 33 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman konsultasi pasien pada bagian *form* hapus konsultasi.



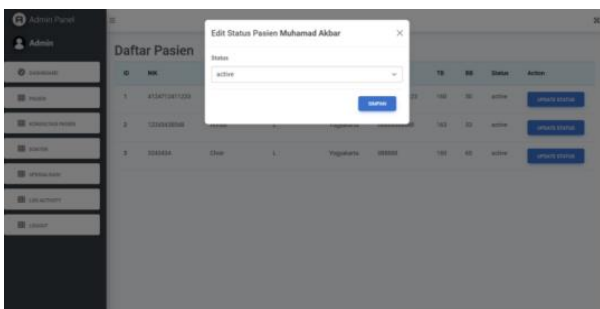
Gambar 30. Halaman Pasien

Gambar 30 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman pasien.



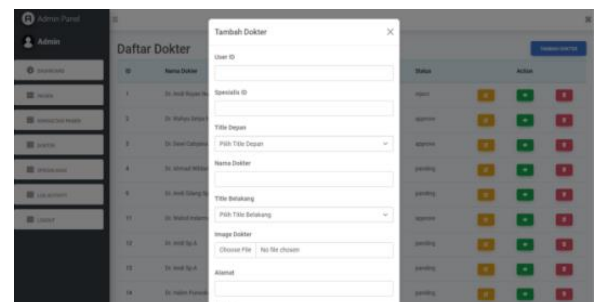
Gambar 34. Halaman Dokter

Gambar 34 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman dokter.



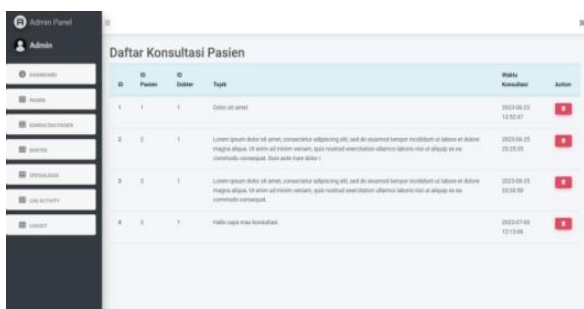
Gambar 31. Form Update Pasien

Gambar 31 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman pasien pada bagian *form* update pasien.



Gambar 35. Form Tambah Dokter

Gambar 35 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman dokter pada bagian *form* tambah dokter.



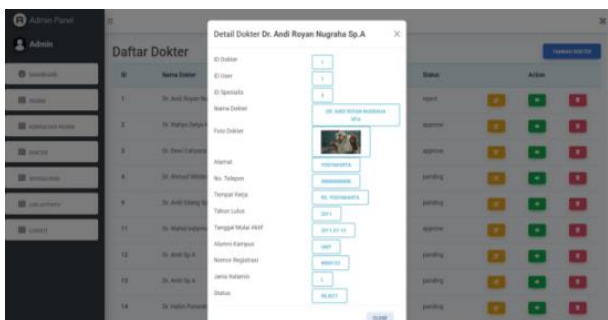
Gambar 32. Halaman Konsultasi Pasien

Gambar 32 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman konsultasi pasien.



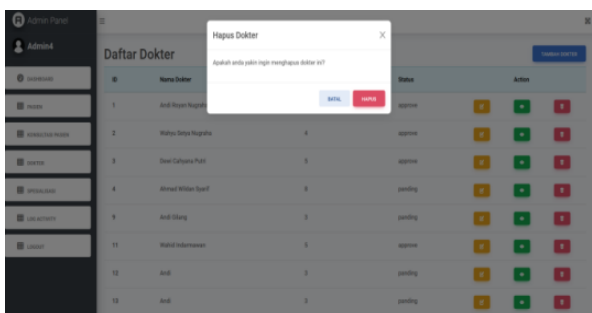
Gambar 36. Form Update Dokter

Gambar 36 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman dokter pada bagian *form* update dokter.



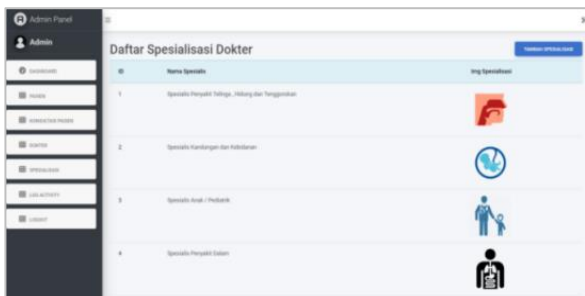
Gambar 37. Form Lihat Dokter

Gambar 37 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman dokter pada bagian *form* lihat dokter.



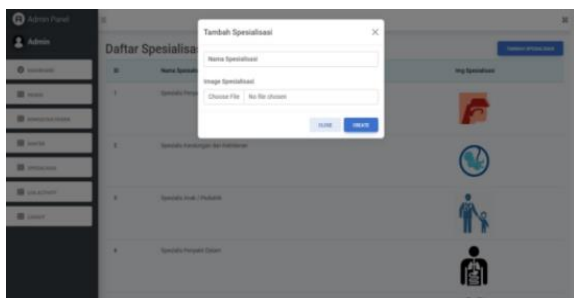
Gambar 38. Form Hapus Dokter

Gambar 38 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman dokter pada bagian *form* hapus dokter.



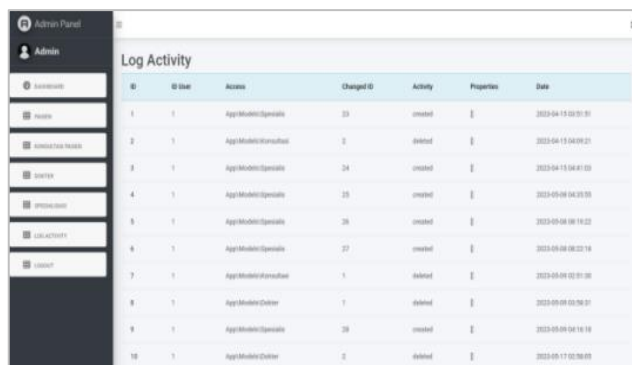
Gambar 39. Halaman Spesialisasi

Gambar 39 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman spesialisasi.



Gambar 40. Form Tambah Spesialisasi

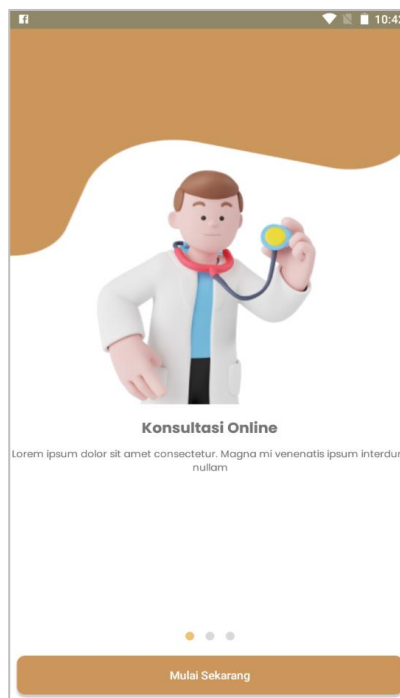
Gambar 40 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman spesialisasi pada bagian *form* tambah spesialisasi.



Gambar 41. Halaman Log Activity

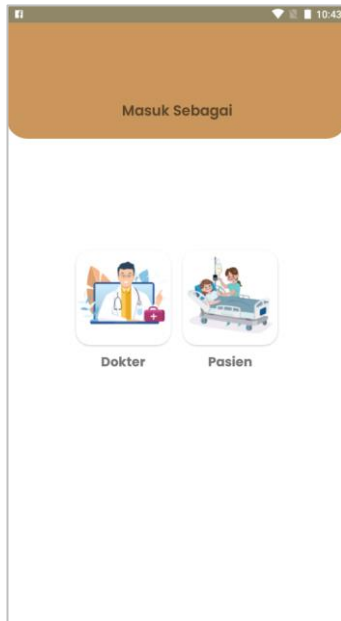
Gambar 41 memvisualisasikan hasil implementasi dari rancangan *wireframe* halaman log activity.

Tampilan *prototype* dari aplikasi *mobile telemedicine* yang ditujukan pada Gambar 42 hingga Gambar 55.



Gambar 42. Halaman Welcome

Gambar 42 memvisualisasikan halaman *welcome* aplikasi *mobile telemedicine*.



Gambar 43. Halaman *Login*

Gambar 43 memvisualisasikan halaman *login* aplikasi *mobile telemedicine*.



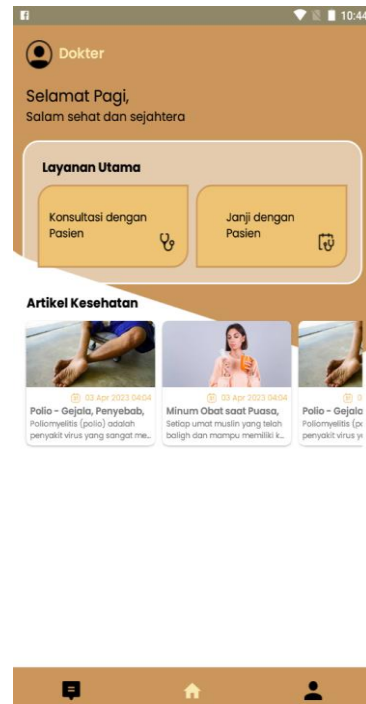
Gambar 45. Halaman *Login* Pasien

Gambar 45 memvisualisasikan halaman *login* pasien aplikasi *mobile telemedicine*.



Gambar 44. Halaman *Login* Dokter

Gambar 44 memvisualisasikan halaman *login* dokter aplikasi *mobile telemedicine*.



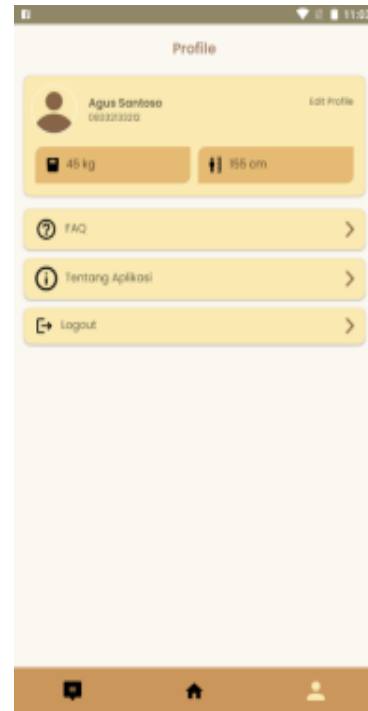
Gambar 46. Halaman Utama Dokter

Gambar 46 memvisualisasikan halaman utama dokter aplikasi *mobile telemedicine*.



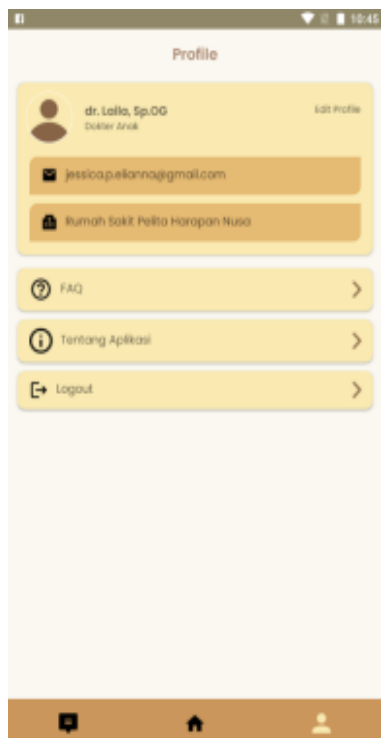
Gambar 47. Halaman Utama Pasien

Gambar 47 memvisualisasikan halaman utama pasien aplikasi *mobile telemedicine*.



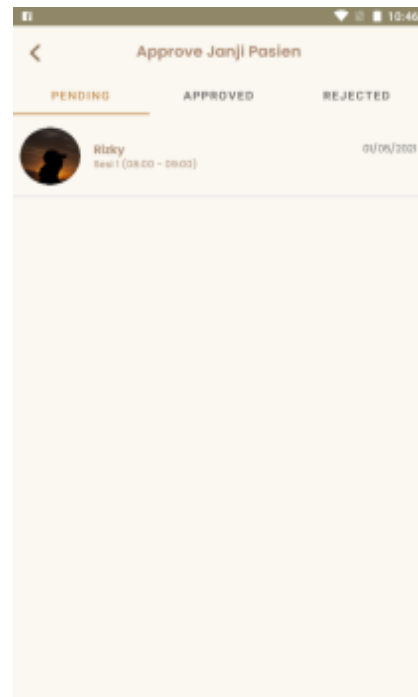
Gambar 49. Halaman Profil Pasien

Gambar 49 memvisualisasikan halaman profil pasien aplikasi *mobile telemedicine*.



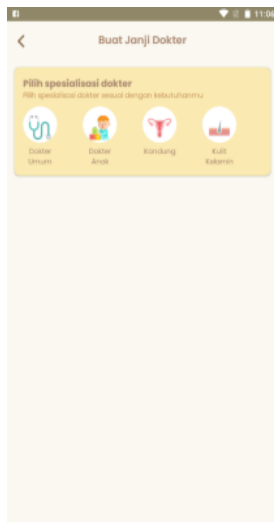
Gambar 48. Halaman Profil Dokter

Gambar 48 memvisualisasikan halaman profil dokter aplikasi *mobile telemedicine*.



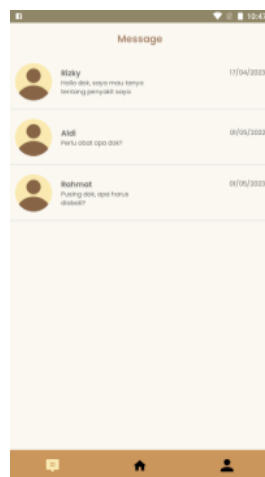
Gambar 50. Halaman Approve Janji Pasien

Gambar 50 memvisualisasikan halaman *approve janji* pasien pada menu dokter aplikasi *mobile telemedicine*.



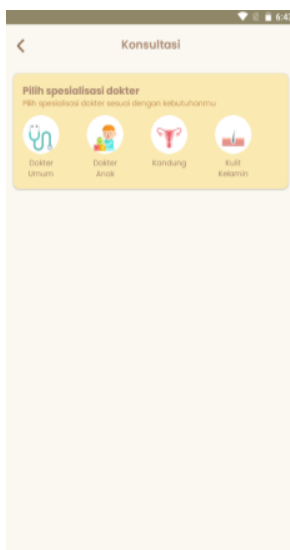
Gambar 51. Halaman Buat Janji Dokter

Gambar 51 memvisualisasikan halaman buat janji dokter pada menu pasien aplikasi *mobile telemedicine*.



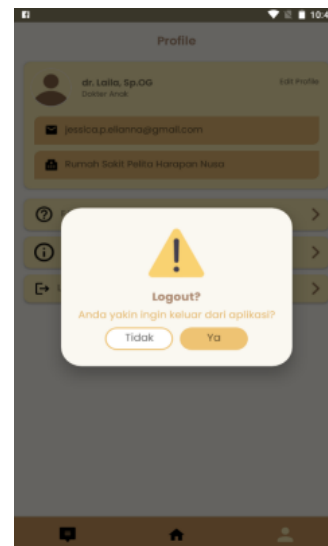
Gambar 52. Halaman Konsultasi Dokter

Gambar 52 memvisualisasikan halaman konsultasi pada menu dokter aplikasi *mobile telemedicine*.



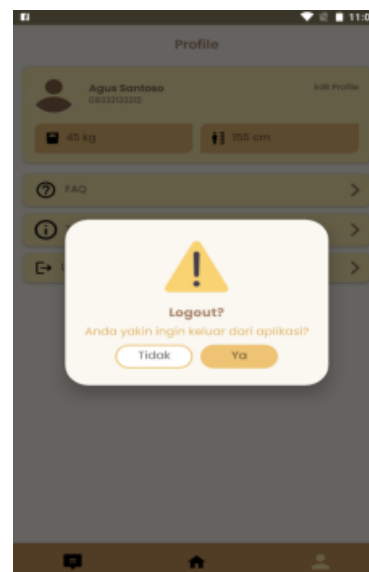
Gambar 53. Halaman Konsultasi Pasien

Gambar 53 memvisualisasikan halaman konsultasi pada menu Pasien aplikasi *mobile telemedicine*.



Gambar 54. Halaman Logout Dokter

Gambar 54 memvisualisasikan halaman *logout* dokter aplikasi *mobile telemedicine*.



Gambar 55. Halaman Logout Pasien

Gambar 55 memvisualisasikan halaman *logout* Pasien aplikasi *mobile telemedicine*.

3.6 User Testing

Bagian ini menjelaskan tentang hasil yang didapatkan dari pengujian *web backoffice* aplikasi *mobile telemedicine* menggunakan metode *black box testing* berfokus pada uji fungsionalitas. Berdasarkan pengujian fitur-fitur *web* yang sudah dilakukan, didapatkan bahwa semua fitur berjalan dengan baik sesuai yang diharapkan. Visualisasi detail hasil pengujian akan tampak pada Tabel 2 hingga Tabel 11.

Hasil *Test Scenario*: Verifikasi Halaman *Welcome* dapat dilihat pada Tabel 2.

Tabel 2. Test Scenario: Verifikasi Halaman Welcome

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol login	Beralih menuju halaman login	Beralih menuju halaman login	Pass
2	Menekan tombol register	Beralih menuju halaman register	Beralih menuju halaman register	Pass

Hasil Test Scenario: Verifikasi Halaman Login dapat dilihat pada Tabel 3.

Tabel 3. Test Scenario: Verifikasi Halaman Login

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	E-mail: admin1@gmail.com Password: Admin1	Berhasil login dan menampilkan halaman dashboard	Berhasil login dan menampilkan halaman dashboard	Pass
2	E-mail: admin@gmail.com Password: Admin1	Berhasil login dan menampilkan halaman dashboard	Tidak berhasil login dan menampilkan halaman dashboard	Pass

Hasil Test Scenario: Verifikasi Halaman Register dapat dilihat pada Tabel 4.

Tabel 4. Test Scenario: Verifikasi Halaman Register

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Fullname: Admin E-mail: admin1@gmail.com Password: Admin1 Confirm Password: Admin1	Berhasil login dan menampilkan halaman dashboard	Berhasil login dan menampilkan halaman dashboard	Pass

Hasil Test Scenario: Verifikasi Halaman Dashboard dapat dilihat pada Tabel 5.

Tabel 5. Test Scenario: Verifikasi Halaman Dashboard

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Ketika berhasil login atau register, akan diarahkan menuju halaman dashboard	Dapat menampilkan data total dokter, pasien, dan jumlah pemeriksaan hari ini	Dapat menampilkan data total dokter, pasien, dan jumlah pemeriksaan hari ini	Pass

Hasil Test Scenario: Verifikasi Halaman Pasien dapat dilihat pada Tabel 6.

Tabel 6. Test Scenario: Verifikasi Halaman Pasien

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol menu pasien	Beralih menuju halaman pasien dan menampilkan seluruh data pasien	Beralih menuju halaman pasien dan menampilkan seluruh data pasien	Pass
2	Menekan tombol update pada menu pasien	Dapat melakukan update status data pasien	Dapat melakukan update status data pasien	Pass

Hasil Test Scenario: Verifikasi Halaman Konsultasi Pasien dapat dilihat pada Tabel 7.

Tabel 7. Test Scenario: Verifikasi Halaman Konsultasi Pasien

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol menu konsultasi pasien	Beralih menuju halaman konsultasi pasien dan menampilkan seluruh data konsultasi	Beralih menuju halaman pasien dan menampilkan seluruh data pasien	Pass
2	Menekan tombol delete pada menu konsultasi pasien	Dapat melakukan delete data konsultasi	Dapat melakukan delete data konsultasi	Pass

Hasil Test Scenario: Verifikasi Halaman Dokter dapat dilihat pada Tabel 8.

Tabel 8. Test Scenario: Verifikasi Halaman Dokter

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol menu dokter	Beralih menuju halaman dokter dan menampilkan seluruh data dokter	Beralih menuju halaman dokter dan menampilkan seluruh data dokter	Pass
2	Menekan tombol tambah pada menu dokter	Dapat menambah data dokter	Dapat menambah data dokter	Pass
3	Menekan tombol update pada menu dokter	Dapat melakukan update data dokter	Dapat melakukan update data dokter	Pass
4	Menekan tombol show pada	Dapat melihat data dokter	Dapat melihat data dokter	Pass

Test Case	Input	Expected Output	Actual Output	Pass/fail?
5	menu dokter Menekan tombol <i>delete</i> pada menu dokter	Dapat melakukan <i>delete</i> data dokter	Dapat melakukan <i>delete</i> data dokter	Pass

Hasil *Test Scenario*: Verifikasi Halaman Spesialisasi dapat dilihat pada Tabel 9.

Tabel 9. *Test Scenario*: Verifikasi Halaman Spesialisasi

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol menu spesialisasi	Beralih menuju halaman spesialisasi dan menampilkan seluruh data spesialisasi dokter	Beralih menuju halaman spesialisasi dan menampilkan seluruh data spesialisasi dokter	Pass
2	Menekan tombol tambah pada menu spesialisasi	Dapat menambah data spesialisasi dokter	Dapat menambah data spesialisasi dokter	Pass

Hasil *Test Scenario*: Verifikasi Halaman *Log Activity* dapat dilihat pada Tabel 10.

Tabel 10. *Test Scenario*: Verifikasi Halaman *Log Activity*

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol menu <i>log activity</i>	Beralih menuju halaman <i>log activity</i> dan menampilkan seluruh data <i>log activity</i>	Beralih menuju halaman <i>log activity</i> dan menampilkan seluruh data <i>log activity</i>	Pass

Hasil *Test Scenario*: Verifikasi Halaman *Logout* dapat dilihat pada Tabel 11.

Tabel 11. *Test Scenario*: Verifikasi Halaman *Logout*

Test Case	Input	Expected Output	Actual Output	Pass/fail?
1	Menekan tombol <i>logout</i>	Berhasil <i>logout</i> dan kembali menuju halaman <i>welcome</i>	Berhasil <i>logout</i> dan kembali menuju halaman <i>welcome</i>	Pass

3.7 Develop Design

Bagian ini menjelaskan tentang *environment* yang digunakan pada perancangan *web backoffice* aplikasi *mobile telemedicine*. Perancangan *web* menggunakan *Framework* Laravel dengan pendekatan arsitektur MVC dan penerapan PHP MVC *Framework* berbasis *web* pada perancangan.

Penjelasan detail *environment API* dapat dilihat pada Tabel 12.

Tabel 12. *API*

Language	Framework	Architecture
PHP	Laravel	MVC (Model - View - Controller)

Penjelasan detail *environment Front-end dan Back-end* dapat dilihat pada Tabel 13.

Tabel 13. Tabel *Front-end* dan *Back-end*

Language	Framework	Architecture	Database
PHP 8.2.0	Laravel 10.13.2	MVC (Model - View - Controller)	MySQL

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, maka dapat ditarik kesimpulan:

- Tersusunnya sebuah dokumentasi sistem perancangan *web backoffice* aplikasi *mobile telemedicine* pada institusi XYZ sesuai dengan yang diharapkan.
- Berdasarkan pengujian dengan *black box testing* yang telah dilakukan, diketahui bahwa fungsional *web backoffice* aplikasi *mobile telemedicine* pada institusi XYZ berjalan sesuai kebutuhan pengguna dengan persentase keberhasilan sebesar 100%.

Harapannya dokumen yang telah tersusun dapat menjadi panduan *developer* ketika akan melakukan pengembangan selanjutnya.

DAFTAR PUSTAKA

- C. M. Annur, "Layanan Telemedicine yang Paling Banyak Digunakan di Indonesia, Apa Saja?." databoks, 2022. <https://databoks.katadata.co.id/> (accessed Jul. 25, 2023).
- C. Rawis, S. D. S. Karouw, and Sherwin R. U. A. Sompie, "Software Requirement Specification Academic Information System of Sam Ratulangi University," J. Tek. Elektro dan Komput., vol. 10, no. 2, pp. 107–118, 2021, [Online]. Available: <https://ejournal.unsrat.ac.id/index.php/elekdankom/article/download/34108/32441>.
- R. A. Serafintino, M. Susilowati, S. Informasi, and U. Machung, "Dokumen Software Requirement

- Spesification (SRS) Sistem Informasi Pemasaran Usaha Jasa Percetakan Dan Iklan,” Kurawal J. Teknol. Inf. dan Ind., vol. 5, no. 2, pp. 117–128, 2022.
- [4] Maydianto and M. R. Ridho, “Rancang Bangun Sistem Informasi Point of Sale Dengan Framework Codeigniter Pada Cv Powershop,” J. Comasie, vol. 04, no. 02, pp. 50–59, 2021.
- [5] M. Waller and C. Stotler, “Telemedicine: a Primer,” *Curr. Allergy Asthma Rep.*, vol. 18, no. 10, p. 54, 2018, doi: 10.1007/s11882-018-0808-4.
- [6] S. Saputra and A. B. Prasetyo, “Scientia Sacra : Jurnal Sains , Teknologi dan Masyarakat Pembuatan dan Implementasi Aplikasi Backoffice untuk Menopang Kebutuhan Monitoring dan Reporting dari Aplikasi PPOB Mobile,” *J. Sci. Sacra J. Sains, Teknol. dan Masy.*, vol. 2, no. 3, pp. 263–271, 2022.
- [7] N. Rizqya, “Perancangan Sistem Informasi Akuntansi Laporan Posisi Keuangan Pada UMKM Berbasis WEB (Studi Kasus UMKM Home Catering),” *J. Ilm. Komputasi*, vol. 19, no. 3, pp. 383–390, 2020, doi: 10.32409/jikstik.19.3.65.
- [8] M. Sholeh, I. Gisfas, Cahiman, and M. A. Fauzi, “Black Box Testing on ukmbantul.com Page with Boundary Value Analysis and Equivalence Partitioning Methods,” *J. Phys. Conf. Ser.*, vol. 1823, no. 1, 2021, doi: 10.1088/1742-6596/1823/1/012029.
- [9] A. Saputra, C. F. I. Safitri, F. Fitriyani, Y. Gulo, and T. Desyani, “Pengembangan Aplikasi Kasir Menggunakan Model Waterfall,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 4, no. 2, p. 86, 2021, doi: 10.32493/jtsi.v4i2.10167.
- [10] N. M. D. Febriyanti, A. A. K. O. Sudana, and I. N. Piarsa, “Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen,” *J. Ilm. Teknol. dan Komput.*, vol. 2, no. 3, pp. 535–544, 2021.
- [11] T. Hidayat and M. Muttaqin, “Pengujian sistem informasi pendaftaran dan pembayaran wisuda online menggunakan black box testing dengan metode equivalence partitioning dan boundary value analysis,” *J. Tek. Inform. UNIS*, vol. 6, no. 1, pp. 2252–5351, 2018. [9]
- [12] M. K. Phillip A. Laplante, *Requirements Engineering for Software and Systems*, 4th ed. New York: Auerbach Publications, 2022. doi: <https://doi.org/10.1201/9781003129509>.
- [13] I. Sommerville, “Integrated Requirements Engineering: A tutorial,” *IEEE Softw.*, pp. 16–23, 2005, doi: 10.1109/MS.2005.13.
- [14] K. Szerovay, “Creating Personas — Part 1,” *UX Knowledge Base Sketch*, 2017. <https://uxknowledgebase.com/> (accessed Aug. 25, 2023).
- [15] R. T. Maulana, “Perancangan User Interface User Experience Dengan Metode User Centered Design Pada Aplikasi Mobile Auctentik,” *Informatics Eng.*, p. 60, 2020, [Online]. Available: <https://dspace.uui.ac.id/123456789/28891>.
- [16] R. P. Sutanto, “Analisis User Flow pada Website Pendidikan: Studi Kasus Website DKV UK Petra,” *J. Desain Komun. Vis. Nirmana*, vol. 22, no. 1, pp. 41–51, 2022, doi: 10.9744/nirmana.22.1.41-51.
- [17] A. Agung, A. Wiryya, R. I. Rokhmawati, and H. M. Az-zahra, “Perancangan dan Evaluasi User Experience Aplikasi Microteaching Pendukung Standar Kompetensi Mahasiswa Calon Guru berbasis Mobile dengan Pendekatan Human Centered Design dan Remote Usability Testing,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 1, pp. 402–410, 2022.
- [18] A. Segara, “Penerapan Pola Tata Letak (Layout Pattern) pada Wireframing Halaman Situs Web,” *J. Magenta, STMK Trisakti*, vol. 3, no. 1, pp. 452–464, 2019.
- [19] F. N. Hasanah and R. S. Untari, *Buku Ajar Rekayasa Perangkat Lunak*, 1st ed. Sidoarjo: UMSIDA Press, 2020



RANCANG BANGUN SISTEM INFORMASI *BACK OFFICE* APLIKASI *TELEMEDICINE* SEBAGAI MEDIA PEMBELAJARAN

Anggit Widcaksono¹, Esi Putri Silmina²

^{1,2} Teknologi Informasi, Universitas 'Aisyiyah Yogyakarta
Sleman, Daerah Istimewa Yogyakarta, Indonesia 55592
anggitwww3@gmail.com, esiputrisilmina@unisayogya.ac.id

Abstract

Back Office applications have a vital role in a product, including in the context of administration. This application will be used by operators or admins whom the company has trusted to monitor and generate reports on the performance of a product or service. It should be emphasized that the Back Office application does not operate independently but has a function to monitor particular objects. This study aims to design and implement a Telemedicine-based Back Office information system application in the context of education. His background is in developing Telemedicine and the importance of efficient management in health administration. Through the type 1 prototype method approach, this research identifies user needs, builds prototypes, evaluates, and uses the system. This Back Office application is integrated with the main web-based application, enabling operators/company admins to monitor and report the performance of related products/services. The result is a system that can increase the effectiveness of education administration through information technology. This study underscores the importance of efficient Back Office management in supporting Telemedicine. Thus, this application can positively impact the delivery of education and health services through the Telemedicine approach.

Keywords: *Back Office Management, Education, Health Administration, Information Systems, Telemedicine*

Abstrak

Aplikasi *Back Office* memiliki peran yang sangat penting dalam sebuah produk, termasuk dalam konteks administrasi. Aplikasi ini akan digunakan oleh operator atau admin yang telah dipercayai oleh perusahaan untuk memantau dan menghasilkan laporan tentang kinerja suatu produk atau layanan. Perlu ditekankan bahwa aplikasi *Back Office* tidak beroperasi secara independen, melainkan memiliki fungsi untuk memantau objek-objek tertentu. Studi ini bertujuan untuk merancang dan mengimplementasikan aplikasi sistem informasi *Back Office* berbasis *Telemedicine* dalam konteks pendidikan. Latar belakangnya adalah pengembangan *Telemedicine* dan pentingnya manajemen yang efisien dalam administrasi kesehatan. Melalui pendekatan metode *prototype* tipe 1, penelitian ini mengidentifikasi kebutuhan pengguna, membangun *prototype*, mengevaluasi, dan menggunakan sistem tersebut. Aplikasi *Back Office* ini terintegrasi dengan aplikasi berbasis web utama, memungkinkan operator/admin perusahaan untuk memantau dan melaporkan kinerja produk/layanan terkait. Hasilnya adalah sebuah sistem yang berpotensi meningkatkan efektivitas administrasi pendidikan melalui teknologi informasi. Studi ini menekankan pentingnya manajemen *Back Office* yang efisien dalam mendukung layanan *Telemedicine*. Dengan demikian, aplikasi ini dapat berdampak positif pada penyampaian pendidikan dan layanan kesehatan melalui pendekatan *Telemedicine*.

Kata kunci: *Administrasi Kesehatan, Manajemen Back Office, Pendidikan, Sistem Informasi, Telemedicine*

1. PENDAHULUAN

Sebuah praktik kesehatan sebagai media pertukaran data medis dan diskusi ilmiah jarak jauh dengan menggunakan komunikasi audio, visual, dan data yang mencakup perawatan, diagnosis, konsultasi dan pengobatan merupakan bagian dari ruang lingkup *Telemedicine* [1]. Berdasarkan pemaparan tersebut, dapat diketahui bahwa *Telemedicine* mencakup secara keseluruhan, yakni diantaranya adalah penyediaan pelayanan kesehatan jarak

jauh meliputi klinis, pendidikan dan pelayanan administrasi, melalui transfer informasi seperti audio, video, dan grafik, menggunakan perangkat telekomunikasi seperti audio-video interaktif dua arah, komputer, dan telemetri dengan melibatkan dokter, pasien serta pihak lain. Dalam kehidupan sehari-hari, praktik *Telemedicine* telah digambarkan secara sederhana saat dua orang dokter mendiskusikan masalah kesehatan pasien melewati telepon [2].

Dalam memberikan sebuah pelayanan yang ramah, menenangkan dan baik perlu membangun sebuah komponen penting guna menunjang proses penyembuhan pasien. Hal ini berkaitan dengan proses komunikasi antara dokter dan pasien yang akan menimbulkan energi positif dari pasien sehingga optimis dalam menjalani proses pengobatan hingga mencapai kesembuhan. Ketika dokter memberikan empati melalui komunikasi secara verbal dan nonverbal, maka dapat membentuk pertukaran informasi (*exchange of information*), hubungan interpersonal yang baik (*creating a good interpersonal relationship*), serta dapat pengambilan keputusan medis (*medical decision making*) [3].

Perkembangan teknologi komunikasi seiring dengan berjalannya waktu semakin meningkat. Salah satu contoh dari perkembangan tersebut adalah adanya inovasi situs daring yang menyediakan konsultasi kesehatan secara daring. Alodokter dan Halodoc merupakan contoh dari beberapa situs konsultasi secara daring yang cukup terkenal di Indonesia. Konsultasi kesehatan daring ini didukung oleh para dokter dengan latar belakang ilmu kesehatan yang mampu menganalisis serta mendiagnosis kondisi kesehatan pasiennya [4]. Dibalik kemudahan berkonsultasi dengan dokter pilihan, keberadaan situs konsultasi *online* ini memiliki risiko permasalahan yang akan terjadi yakni, kesulitan dalam memfasilitasi perilaku serta proses memotivasi pasien menjadi tidak efektif. Kesulitan lain yang timbul dan cukup serius adalah informasi kesehatan yang disampaikan terkadang kurang relevan maka akan berisiko terjadinya penurunan kualitas pelayanan dan kepercayaan pasien [5]. Sebuah permasalahan selanjutnya adalah bagaimana seorang dokter dapat menganalisis dan menetapkan diagnosa penyakit pasien tanpa melihat langsung melainkan hanya melalui keluhan yang disampaikan pasien dan bagaimana tingkat kepercayaan pasien terkait hasil diagnosis yang diberikan dokter.

Pelayanan terhadap *customer* adalah bagian dari strategi manajemen *Back Office*, hal ini dapat terjadi meskipun bagian *Back Office* tidak berhubungan langsung dengan *customer*. Sementara itu, baik buruknya pelayanan *front-liner* secara keseluruhan di dalamnya terdapat kontribusi tugas *Back Office*. Mengingat tugas *Back Office* yang berpengaruh dalam menunjang kinerja pada *front liner*, oleh karena itu merealisasikan manajemen *Back Office* menjadi sebuah ketentuan khusus, terutama manajemen *Back Office* yang berdasar atas teknologi informasi yang sesuai dengan perkembangan teknologi yang ada saat ini. Di samping itu, data yang masih berbentuk lembaran kertas serta catatan buku menimbulkan hambatan bagi staf di bagian *front-liner* untuk mencari informasi yang dibutuhkan saat ingin membuat keputusan dalam menyelesaikan masalah, baik itu masalah yang terkait dengan pelanggan, transaksi, maupun masalah internal antara staf di bagian belakang dan staf di bagian depan. Berdasarkan pengamatan yang dilakukan oleh staf di bagian belakang, saat ini mereka memerlukan sebuah aplikasi yang dapat digunakan sebagai alat

manajemen yang dapat menyimpan informasi secara digital tidak hanya mudah digunakan, namun juga dapat membantu mempercepat proses transaksi pada bagian depan. Aplikasi ini diharapkan dapat mengelola informasi yang telah tersimpan pada *database*, seperti mengubah, menambahkan, menghapus, dan mencari informasi [6].

Penelitian sebelumnya, penelitian yang dilakukan oleh Suhandha Saputra dkk. dalam mengimplementasikan aplikasi *Back Office* ini bertujuan untuk membantu perusahaan dalam memonitoring kinerja dari aplikasi utamanya (aplikasi *mobile* berbasis *android*) dimana aplikasi ini digunakan oleh *end user*, dan aplikasi ini bisa digunakan pula oleh admin perusahaan untuk memonitoring dan membuat laporan kinerja sehari-hari [7].

Penelitian tentang *Back Office* pada salah satu bank ternama di Asia, yang dilakukan oleh Abdul Jamil. dkk menghasilkan *prototype* bahwa aplikasi manajemen pengelolaan *Back Office* yang berdasar pada teknologi informasi sebagai solusi alternatif terhadap solusi dari masalah yang muncul pada kegiatan bagian *Back Office* dalam menunjang keberhasilan *front-liner* [8].

Penelitian ini berbeda dengan penelitian sebelumnya, hal ini dikarenakan penelitian ini mengarah pada rancang bangun sistem informasi *Back Office* khusus untuk administrasi dalam aplikasi *Telemedicine*. Fokusnya adalah pada penggunaan teknologi informasi untuk mengelola aspek administratif dalam konteks pendidikan kedokteran, dengan aplikasi tersebut berperan sebagai media pembelajaran bagi mahasiswa [9].

Aplikasi *Back Office* adalah bagian aspek terpenting dari sebuah produk bahkan untuk administrasi, *Back Office* akan digunakan oleh admin/operator yang dipercaya oleh sebuah perusahaan untuk memonitoring sebuah aplikasi dan bahkan membuat sebuah laporan dari hasil kinerja sebuah jasa/produk. sebuah aplikasi *Back Office* tidak berjalan sendiri yang artinya aplikasi ini memiliki objek untuk di monitor, dalam hal ini biasanya sebuah perusahaan akan membuat sebuah aplikasi *Back Office* guna menunjang kinerja dari sebuah aplikasi lainnya seperti aplikasi web yang digunakan oleh *end user*. Pembuatan dan implementasi aplikasi *Back Office* ini bertujuan untuk membantu administrasi dalam memonitoring kinerja dari aplikasi utamanya yakni aplikasi berbasis web, dimana aplikasi ini digunakan oleh *end user* dan aplikasi *Back Office* ini bisa digunakan pula oleh admin perusahaan untuk memonitoring dan membuat laporan kinerja sehari-hari [10].

Berdasarkan latar belakang tersebut, maka dirancang penelitian dengan judul “Rancang Bangun Sistem Informasi *Back Office* untuk Administrasi pada Aplikasi *Telemedicine* sebagai Media Pembelajaran Mahasiswa Kedokteran” sebagai solusi untuk mengatur data yang telah tersimpan pada *database* seperti mengubah data, menambah data, menghapus data dan mencari data [11].

Sistem Informasi

Sistem informasi adalah kerangka terorganisir yang mencakup perangkat lunak, perangkat keras, prosedur, data, dan orang-orang yang bekerja bersama untuk mengumpulkan, menyimpan, mengelola, mengolah, dan menyampaikan informasi yang relevan dan bermanfaat kepada pengguna dalam sebuah organisasi [12]. Tujuan utama dari sistem informasi adalah untuk membantu dalam pengambilan keputusan, pengelolaan operasional, analisis data, dan peningkatan efisiensi dalam berbagai aspek bisnis atau aktivitas. Dengan menggunakan teknologi komputer dan komunikasi, sistem informasi mendukung pengumpulan data, transformasi menjadi informasi, dan pengiriman ke berbagai tingkat manajemen serta pemangku kepentingan lainnya, sehingga memainkan peran krusial dalam mencapai tujuan organisasi secara lebih efektif dan efisien [13].

Manajemen *Back Office*

Menurut James A.F Stonner (1978) memaparkan bahwa manajemen adalah sebuah proses yang meliputi pengorganisaasian, perencanaan, pengawasan serta pengarahan usaha-usaha dari para anggota organisasi beserta penggunaan sumber-sumber daya organisasi lain yang berfungsi untuk mencapai tujuan organisasi yang sudah lama ditetapkan [14].

Back Office adalah bagian dari suatu organisasi atau perusahaan yang berkepentingan atas proses dan kegiatan administratif serta operasional yang mendukung fungsi utama perusahaan, tetapi tidak berhubungan langsung dengan interaksi pelanggan atau pihak eksternal. Fokus utama *Back Office* adalah efisiensi dalam pengelolaan data, administrasi, keuangan, dan logistik, termasuk tugas-tugas seperti pemrosesan transaksi, manajemen inventaris, pelaporan keuangan, pengelolaan sumber daya manusia, serta pemeliharaan infrastruktur teknologi informasi. Meskipun tidak terlihat oleh pelanggan, peran *Back Office* yang teratur dan efisien penting dalam memastikan keseluruhan operasional perusahaan berjalan dengan baik dan sesuai dengan tujuan strategis yang telah ditetapkan [8]. Definisi yang telah dijabarkan tersebut membantu dalam memberikan informasi bahwa pada konteks manajemen *Back Office* akan lebih tepat diartikan sebagai bagian proses administrasi yang berkaitan dengan proses perencanaan, pengorganisaasian, pengawasan dan pengarahan terhadap usaha para anggota organisasi serta penggunaan sumber-sumber daya organisasi lain guna mencapai tujuan organisasi yang sudah ditetapkan sejak lama [15].

Telemedicine

Telemedicine adalah praktik pemberian layanan kesehatan jarak jauh yang memanfaatkan teknologi komunikasi seperti telepon, komputer, dan perangkat seluler untuk melakukan konsultasi medis, diagnosis, pengobatan, dan pemantauan pasien [16]. Melalui *Telemedicine*, dokter dan pasien dapat

berinteraksi secara virtual tanpa harus bertatap muka secara langsung di tempat yang sama. Ini memungkinkan akses ke perawatan medis yang lebih mudah, terutama bagi mereka yang berada di daerah terpencil atau sulit dijangkau, serta memungkinkan para profesional kesehatan untuk memberikan nasihat dan layanan medis cepat dalam situasi darurat atau keterbatasan fisik tertentu. *Telemedicine* juga mencakup berbagai layanan seperti konsultasi video, perawatan jarak jauh, dan monitoring kesehatan jarak jauh menggunakan teknologi *wearable* [17].

Model *Prototyping*

Pendekatan di mana pengembang dan pemangku kepentingan bekerja sama untuk menciptakan *prototype* awal dari perangkat lunak yang akan dikembangkan atau sering juga disebut dengan *Software Development Life Cycle* (SDLC) adalah model *prototyping* dalam melakukan penelitian ini. Tujuan utama dari model ini adalah untuk menggambarkan dan menguji konsep, fitur, dan fungsi dari perangkat lunak secara lebih konkret sebelum mengembangkan versi akhirnya. Proses ini melibatkan iterasi berulang, di mana *prototype* awal dikembangkan, dievaluasi, dan diperbaiki berdasarkan umpan balik yang diterima dari pemangku kepentingan [18]. Pendekatan ini membantu mengidentifikasi persyaratan yang mungkin belum jelas atau perlu diubah, sehingga mengurangi risiko kesalahan yang mungkin terjadi pada tahap akhir pengembangan. Setelah *prototype* dianggap memadai, pengembang dapat melanjutkan ke tahap pengembangan lebih lanjut dengan panduan yang lebih jelas dari hasil *prototyping* [19]. Dengan demikian, model *prototyping* memberikan ruang bagi fleksibilitas dan adaptasi yang lebih besar selama proses pengembangan perangkat lunak [20].

Application Programming Interface (API)

Application Programming Interface (API) merupakan aturan, set protokol, dan alat yang memungkinkan berbagai perangkat lunak atau aplikasi untuk berkomunikasi dan berinteraksi satu sama lain. API mengatur cara bagaimana berbagai komponen perangkat lunak dapat saling berbicara, menukar data, dan mengakses fungsionalitas masing-masing. Dengan API, pengembang dapat mengintegrasikan fitur atau layanan dari satu aplikasi ke dalam aplikasi lain tanpa harus memahami secara mendalam bagaimana kode sumber asli bekerja. Ini memungkinkan pengembangan lebih cepat, kolaborasi antar tim, dan inovasi yang lebih baik dalam ekosistem perangkat lunak. API sering digunakan untuk mengakses layanan web, basis data, perangkat keras, atau fungsi internal dari suatu aplikasi, dan berfungsi sebagai jembatan untuk menghubungkan berbagai sistem secara efisien dan efektif [21].

2. METODE PENELITIAN

2.1 Teknik Pengumpulan Data

Penelitian ini dikerjakan oleh penulis dengan menggunakan teknik pengumpulan data sebagai berikut:

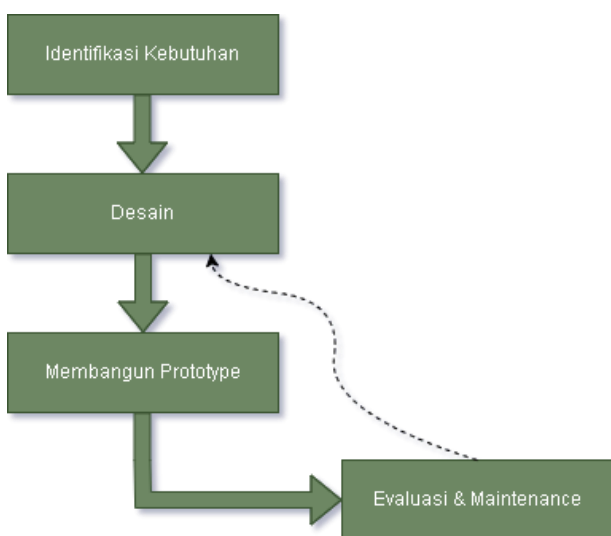
1. Melakukan proses tanya jawab secara langsung terhadap orang yang bertanggung jawab mengerjakan aplikasi *Telemedicine* saat bekerja, dengan bertanya terkait hal-hal yang berhubungan dengan keperluan sistem.
2. Studi Pustaka, menelaah dokumen penting dilakukan juga agar lebih mengetahui permasalahan yang pernah terjadi dan bagaimana cara mengatasinya.

2.2 Metode Pengembangan Sistem

Metode pengembangan sistem ini, penulis memakai cara *prototype* tipe satu (1) yang mencakup jenjang sebagai berikut:

1. Mengidentifikasi kebutuhan *user* : Mencari informasi apa saja yang akan menjadi keperluan oleh *user*.
2. Mengembangkan *prototype* : Tahap awal yaitu membuat *prototype* dengan perancangan sementara, penkodean sistem, serta melakukan uji pada sistem aplikasi yang telah dibuat sehingga pantas dan berfungsi untuk kebutuhan yang telah teridentifikasi.
3. Menggunakan sistem : Setelah dievaluasi dan diterima *user* atau pengguna perangkat lunak siap untuk digunakan

Secara diagram tahapan pengembangan perangkat lunak (*software*) dengan metode *prototype type 1* dapat dilihat pada gambar 1 berikut ini:



Gambar 1. Diagram Metode Prototype Type 1

3. HASIL DAN PEMBAHASAN

Pada sesi ini, penulis menjelaskan hasil dari pembahasan dan penelitian secara menyeluruh, teliti dan meliputi banyak hal baik dari tampilan desain *website* hingga analisa kebutuhan. Kebutuhan fungsional dan non fungsional dari sistem adalah penjabaran pada analisa kebutuhan sistem [22].

3.1 Identifikasi Kebutuhan

Rancangan aplikasi Sistem Informasi *Back Office* terdiri dari 2 kebutuhan terhadap sistem, yaitu kebutuhan non-fungsional dan kebutuhan fungsional. Kebutuhan fungsional sistem mencakup dari fungsi primer yang sangat berkaitan erat dan sebagai penunjang satu sama lain. Sementara itu, kebutuhan non-fungsional sistem yang berfungsi untuk sarana pendukung agar fungsi utama berjalan dengan lancar dan beroperasi sesuai dengan ekspektasi. Kebutuhan fungsional sistem terdiri dari beberapa fungsi utama yang saling berkaitan erat dan saling mendukung satu sama lain.

1. Analisa Kebutuhan Fungsional

Dalam rancangan aplikasi Sistem Informasi *Back Office* terdiri dari dua akses, yaitu admin setiap tempat pelayanan dan pasien. Adapun kegiatan atau skenario yang akan diterapkan oleh admin dan pasien adalah sebagai berikut:

- a) Skenario Kebutuhan Bagian Admin: *login*, mengelola data pasien, mengelola data konsultasi pasien, mengelola data dokter, mengelola data spesialisasi, mengelola data *log activity*, dan melakukan *logout*.
- b) Skenario Kebutuhan Bagian Pasien: registrasi, *login*, meng-*input* data konsultasi pasien, melihat data *specialist* dokter, melihat data dokter, dan melakukan *logout*.

2. Analisa Kebutuhan Non-Fungsional

Kebutuhan non fungsional adalah kebutuhan pengoptimalan kebutuhan fungsional agar saling mendukung yang meliputi kebutuhan akan spesifikasi perangkat keras dan kebutuhan perangkat lunak yang sangat mempengaruhi beroperasinya *website* dengan jaringan lokal. Berikut ini adalah kebutuhan non-fungsional tersebut meliputi:

a) Hardware

Spesifikasi minimum *hardware* yang dibutuhkan aplikasi Sistem Informasi *Back Office* ini sebagai berikut:

1. *Processor* : 1.5 GHz Dual Core
2. *Memory (RAM)* : 8 GB
3. *Hard disk* : 256 GB
4. *Monitor* : 1366 x 768

b) Software

Spesifikasi minimum *software* yang dibutuhkan aplikasi Sistem Informasi *Back Office* ini sebagai berikut:

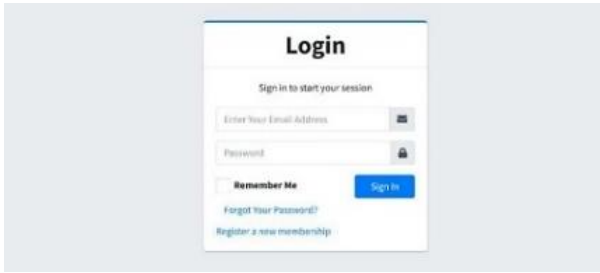
1. *Operation System* : Windows 11
2. *Software Web Server* : XAMPP
3. *Programming Language* : PHP, Laravel, JS
4. *Editor Text* : Visual Studio Code
5. *Browser* : Google Chrome

3.2 Implementasi Program

Pada tahap implementasi program, program yang sudah dibuat memiliki tujuan yang diinginkan sesuai dengan apa

yang ingin dicapai. Berikut yaitu implementasi program harus dilakukan pengimplementasian agar mempunyai dampak yang baik bagi pengguna.

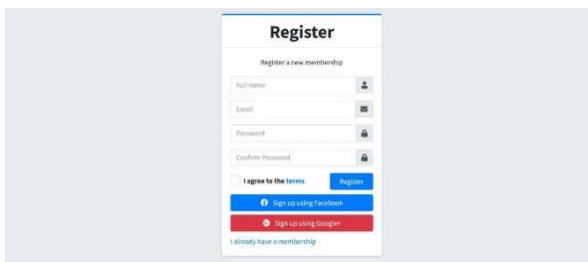
1. Halaman Masuk



Gambar 2. Halaman Masuk

Gambar 2 menunjukkan halaman masuk yang terdiri dari elemen pada *user interface*, yaitu *login button* untuk masuk ke dalam aplikasi, *input e-mail* dan *password*. Jika *user* belum memiliki akun, maka *user* haruskan daftar terlebih dahulu melalui link register pada bagian paling bawah pada halaman *login*.

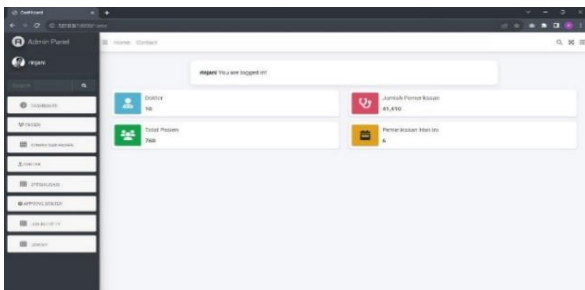
2. Halaman Registrasi



Gambar 3. Halaman Registrasi

Gambar 3 menunjukkan halaman registrasi yang terdiri beberapa elemen yang terdapat pada *user interface*, seperti *input* nama, *e-mail* dan *password*, dan *register* menggunakan sosial media seperti *facebook* dan *google* pada halaman *register*. Serta terdapat *register button* pada bagian bawah.

3. Halaman Dasbor

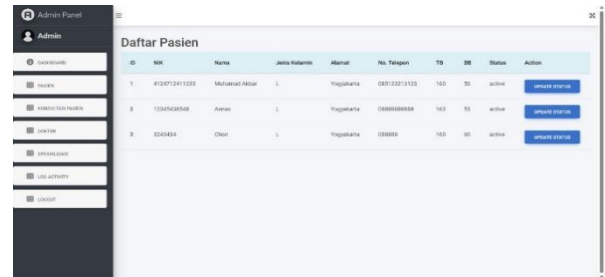


Gambar 4. Halaman Dasbor

Gambar 4 menunjukkan halaman dasbor tersebut, terdapat sejumlah elemen penting dalam antarmuka pengguna. Setiap halaman *dashboard* menampilkan data kunci, termasuk jumlah dokter yang tersedia, jumlah pemeriksaan

yang telah dilakukan, total jumlah pasien yang terdaftar, dan total pemeriksaan yang berlangsung pada hari ini. Selain itu, dalam setiap halaman juga terdapat sebuah kotak pencarian (*search bar*) yang memungkinkan pengguna untuk dengan mudah mencari informasi spesifik yang dibutuhkan. Elemen-elemen ini secara komprehensif memberikan pandangan langsung tentang kinerja dan aktivitas terkini dalam lingkup pelayanan kesehatan, sambil memberikan kemampuan pencarian yang efisien bagi pengguna dalam mengakses data yang diinginkan.

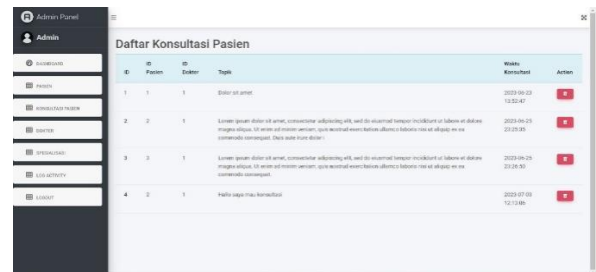
4. Halaman Pasien



Gambar 5. Halaman Pasien

Halaman daftar Pasien terdiri dari profil pasien yang sudah *login* dan dapat melihat persetujuan pesanan pasien. Profil pasien meliputi Nama, NIK, TB, BB, Alamat, Status dan Jenis Kelamin. Adapun bentuk dari halaman data pasien dapat dilihat pada Gambar 5.

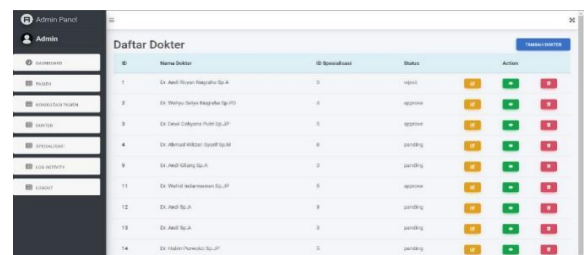
5. Halaman Konsultasi Pasien



Gambar 6. Halaman Konsultasi Pasien

Gambar 6 dibuat oleh pasien yang memaparkan halaman konsultasi dokter. Halaman konsultasi ini tidak akan muncul di halaman dokter jika pasien tidak melakukan konsultasi. Halaman konsultasi adalah halaman interaksi dan percakapan konsultasi antara pasien dan dokter, dikala pasien melaksanakan konsultasi kepada dokter yang bersangkutan.

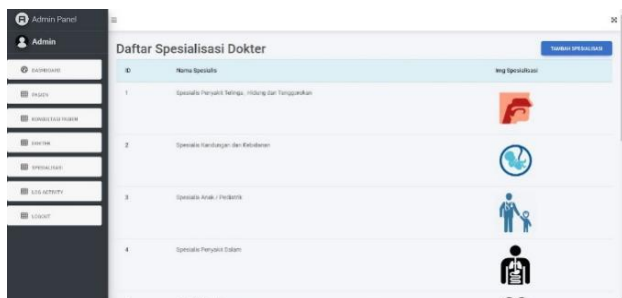
6. Halaman Daftar Dokter



Gambar 7. Halaman Daftar Dokter

Gambar 7 merupakan halaman daftar dokter yaitu sebuah halaman dalam suatu sistem informasi yang digunakan untuk mengelola informasi mengenai dokter-dokter yang terlibat dalam praktik medis. Fungsi utama halaman ini adalah untuk memberikan kemampuan kepada pengguna (biasanya *administrator* atau staf medis) untuk melakukan operasi seperti menambah, mengedit, dan menghapus data dokter dalam basis data sistem.

7. Halaman Daftar Spesialisasi Dokter



Gambar 8. Halaman Daftar Spesialisasi Dokter

Gambar 8 merupakan tampilan yang memuat informasi terkait berbagai spesialisasi yang dimiliki oleh dokter. Melalui halaman ini, pengguna dapat melihat daftar lengkap spesialisasi yang tersedia dalam praktik medis. Tujuan utamanya adalah untuk memberikan gambaran menyeluruh tentang beragam bidang spesialisasi yang dapat ditawarkan oleh dokter-dokter yang tergabung dalam sistem ini. Selain itu, pengguna juga memiliki kemampuan untuk menambahkan spesialisasi baru ke dalam daftar, memungkinkan pengembangan dan penyesuaian yang lebih lanjut sesuai dengan perkembangan kebutuhan layanan kesehatan. Dengan adanya halaman ini, informasi terkait spesialisasi dokter dapat dikelola dengan lebih terstruktur dan efisien.

8. Halaman Aktivitas Log

ID	ID User	Akses	Changed ID	Activity	Properties	Date
1	1	App Mobile Operasia	23	inserted	E	2023-04-15 01:51:41
2	1	App Mobile Operasia	24	deleted	E	2023-04-15 04:19:23
3	1	App Mobile Operasia	24	inserted	E	2023-04-15 04:41:03
4	1	App Mobile Operasia	25	inserted	E	2023-05-08 04:23:35
5	1	App Mobile Operasia	26	inserted	E	2023-05-08 08:19:22
6	1	App Mobile Operasia	27	inserted	E	2023-05-08 08:22:19
7	1	App Mobile Operasia	1	deleted	E	2023-05-09 01:51:30
8	1	App Mobile Operasia	1	deleted	E	2023-05-09 01:56:51
9	1	App Mobile Operasia	28	inserted	E	2023-05-09 04:34:19
10	1	App Mobile Operasia	2	deleted	E	2023-05-17 02:58:07

Gambar 9. Halaman Aktivitas Log

Gambar 9 aktivitas log web biasanya berisi catatan atau riwayat aktivitas yang dilakukan oleh pasien atau pengguna di platform tersebut. Ini meliputi proses *Create*, *Read*, *Update*, dan *Delete* digunakan untuk mengelola data pada sistem. Tujuannya adalah untuk memberikan pandangan yang jelas tentang sejarah medis dan aktivitas pengguna di platform perangkat lunak tersebut.

4. KESIMPULAN

Penelitian ini menghasilkan sebuah rancang bangun sistem informasi *Back Office* yang memiliki fokus utama pada pengelolaan administratif dalam konteks pendidikan kedokteran melalui aplikasi *Telemedicine*. Dalam konteks ini, aplikasi *Back Office* menjadi elemen kunci yang mendukung efisiensi administrasi dengan memberikan kemampuan kepada operator/admin perusahaan untuk memonitor dan melaporkan kinerja produk atau jasa yang terkait. Aplikasi *Back Office* ini tidak berdiri sendiri, melainkan berintegrasi dengan aplikasi utama berbasis web yang digunakan oleh pengguna akhir. Tujuan utama pembuatan dan implementasi aplikasi *Back Office* adalah untuk membantu administrasi dalam mengawasi kinerja aplikasi utama serta memberikan kemudahan kepada admin perusahaan dalam menghasilkan laporan kinerja harian. Metodologi pengembangan sistem yang digunakan adalah metode *prototype* tipe 1, yang mencakup langkah-langkah seperti identifikasi kebutuhan pengguna, pembangunan *prototype*, evaluasi *prototype*, dan penggunaan sistem.

Dengan demikian, penelitian ini menghasilkan kontribusi berupa sebuah sistem yang memiliki potensi untuk meningkatkan efektivitas seperti penghematan waktu, pengurangan kesalahan manusia, peningkatan produktivitas, dan peningkatan kualitas layanan. dalam konteks pendidikan kedokteran melalui pemanfaatan teknologi informasi. Untuk penelitian ke depannya, disarankan untuk lebih mendalam dalam mengintegrasikan aspek teknologi *Telemedicine* dengan sistem informasi *Back Office* yang ada. Dapat dipertimbangkan untuk memperluas cakupan fungsionalitas *Back Office* agar mencakup pengelolaan data pasien, jadwal dokter, dan pemberian layanan *Telemedicine* secara lebih komprehensif. Selain itu, penelitian bisa mengeksplorasi cara meningkatkan interaksi antara aplikasi utama berbasis web dan aplikasi *Back Office*, mungkin dengan penerapan alat analisis data untuk memberikan wawasan yang lebih mendalam kepada pengelola dalam mengambil keputusan. Selanjutnya, perlu juga diidentifikasi potensi pengembangan lanjutan seperti integrasi dengan platform *mobile* untuk akses yang lebih fleksibel.

DAFTAR PUSTAKA

- [1] G. G. Sari and W. Wirman, "Telemedicine sebagai Media Konsultasi Kesehatan di Masa Pandemic COVID 19 di Indonesia," *J. Komun.*, vol. 15, no. 1, pp. 43–54, 2021, doi: 10.21107/ilkom.v15i1.10181.
- [2] M. Jamil, A. Khairan, and A. Fuad, "The Implementation of Social Network Based Telemedicine Application with The Use of Cloud Computing technology," *J. Edukasi dan Penelit. Inform.*, vol. 1, no. 1, pp. 1–5, 2015, [Online]. Available: <https://jurnal.untan.ac.id/index.php/jepin/article/view/9930>
- [3] T. M. dan H. K. Liansyah, "Pentingnya Komunikasi

- Dalam Pelayanan Kesehatan Primer,” *J. Kedokt. Syiah Kuala*, vol. 15, no. 2, pp. 120–124, 2015, [Online]. Available: <https://jurnal.unsyiah.ac.id/JKS/article/view/3264/3075>
- [4] A. Suprianto and A. A. Fathia Matsea, “Rancang Bangun Aplikasi Pendaftaran Pasien Online Dan Pemeriksaan Dokter Di Klinik Pengobatan Berbasis Web,” *J. Rekayasa Inf.*, vol. 7, no. 1, pp. 48–58, 2018.
- [5] M. Iqbal and Husin, “Analisis Perbandingan Kualitas Dan Fasilitas Web Konsultasi Kesehatan Dengan Pendekatan Pieces Framework,” *MIKROTIK J. Manaj. Inform.*, vol. 7, no. 2, pp. 62–70, 2017.
- [6] M. Zaini and R. Riyadi, “Penguatan Pengelolaan Back Office E-Commerce Untuk Peningkatan Kapasitas Ukm Di Kota Samarinda,” *J. Adm. Bisnis Fisipol Unmul*, vol. 9, no. 3, p. 255, 2021, doi: 10.54144/jadbis.v9i3.6330.
- [7] S. Saputra and A. B. Prasetyo, “Scientia Sacra : Jurnal Sains , Teknologi dan Masyarakat Pembuatan dan Implementasi Aplikasi Backoffice untuk Menopang Kebutuhan Monitoring dan Reporting dari Aplikasi PPOB Mobile,” vol. 2, no. 3, pp. 263–271, 2022.
- [8] W. D. D. Jamil Abdul, Bachtiar Syamsul, “Manajemen Back Office Berbasis Teknologi Informasi Pada Pt. Bank Central Asia, Tbk,” *J. Sist. Inf. Univ. Suryadarma*, vol. 9, no. 1, 2014, doi: 10.35968/jsi.v9i1.844.
- [9] M. S. T. Sulila Iswanto , Agus Lahinta, “Sistem Informasi Pelayanan Kesehatan Berbasis Web Pada Klinik Gocare,” *J. Inform.*, vol. 1, no. 1, pp. 1–6, 2020.
- [10] D. H. P.H. and E. M. Sari Sakti, “Perancangan Table Manage Product dalam Aplikasi Back Office Alfagift Berbasis Web,” vol. 24, no. 1, pp. 1–7, 2023.
- [11] S. F. Karomah, A. P. Wicaksono, M. C. Roziqin, and E. Selviyanti, “Pembuatan Website Company Dengan Fitur Antrian Online Dan Pendaftaran Pasien Rawat Jalan Di Puskesmas Wringin Bondowoso,” *J-REMI J. Rekam Med. dan Inf. Kesehat.*, vol. 4, no. 1, pp. 1–11, 2022, doi: 10.25047/j-remi.v4i1.3344.
- [12] Maydianto and M. R. Ridho, “Rancang Bangun Sistem Informasi Point of Sale Dengan Framework Codeigniter Pada Cv Powershop,” *J. Comasie*, vol. 02, pp. 50–59, 2021.
- [13] M. F. Nurhakim, S. Y. Putranto, M. Faisal, and A. Nurohmah, “Perancangan Sistem Informasi Penyewaaan Lapangan Futsal Berbasis Website D ’ soccer Karawang,” vol. 3, no. 1, pp. 51–58, 2023.
- [14] J. A. F. Stoner, “Management,” 1996.
- [15] B. Aji Pamungkas and S. Handayaningsih, “Implementasi Back Office Aplikasi M-Goverment,” *J. Sarj. Tek. Inform. e-ISSN 2338-5197 Vol. 2 Nomor 1, Februari 2014*, vol. 2, pp. 60–66, 2014.
- [16] A. Wicaksono and B. Setianto, “Layanan Telemedicine Rumah Sakit Islam Surabaya sebagai Upaya dalam Menurunkan Kasus Covid 19,” *To Maega J. Pengabd. Masy.*, vol. 5, no. 2, p. 292, 2022, doi: 10.35914/tomaega.v5i2.1081.
- [17] R. Andriani, A. Nisaa, and F. Nurbaya, “Model Penerimaan Teknologi Telemedicine Pada Masa Pandemi COVID-19,” *J. Manaj. Inf. Kesehat. Indones.*, vol. 11, no. 1, 2023, doi: 10.33560/jmiki.v11i1.505.
- [18] T. Qintari, T. Suratno, and M. Mauladi, “Rancang Bangun Sistem Informasi Tahanan dan Barang Bukti Menggunakan Model Prototype Pada Kepolisian Daerah Jambi,” *JUSS (Jurnal Sains dan Sist. Informasi)*, vol. 2, no. 1, pp. 36–44, 2019, doi: 10.22437/juss.v2i1.7400.
- [19] S. Sismadi, “Penerapan Model Prototipe Aplikasi Perangkat Lunak Pemesanan Air Bersih Pdam Tirta Pakuan Kota Bogor,” *INTI Nusa Mandiri*, vol. 15, no. 2, pp. 119–126, 2021, doi: 10.33480/inti.v15i2.1822.
- [20] P. Yoko, R. Adwiya, and W. Nugraha, “Penerapan Metode Prototype dalam Perancangan Aplikasi SIPINJAM Berbasis Website pada Credit Union Canaga Antutn,” *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 7, no. 3, p. 212, 2019, doi: 10.24843/jim.2019.v07.i03.p05.
- [21] N. A. Rakhmawati, S. H. Suryawan, M. A. Furqon, and D. Hermansyah, “Indonesia’s Public Application Programming Interface (API),” *J. Penelit. Pos dan Inform.*, vol. 9, no. 2, p. 85, 2019, doi: 10.17933/jppi.2019.090201.
- [22] A. Prabowo, “Pendaftaran Pasien pada UPT Puskesmas Tewah Berbasis Web Mobile Analysis and Design of Mobile Web-Based Information System of Patient Registration at UPT Puskesmas Tewah,” vol. 5, no. 2, pp. 41–46, 2023.



ANALISIS KRIPTOGRAFI ALGORITMA BLOWFISH PADA KEAMANAN DATA MENGGUNAKAN DART

Yoga Pratama¹, Tata Sutabri²

^{1,2}Teknik Informatika, Universitas Bina Darma
Palembang, Sumatera Selatan, Indonesia 30111
yogapratama.sc@gmail.com, tata.sutabri@gmail.com

Abstract

Data security is a vital aspect of information technology to maintain the integrity and authenticity of data from unauthorized access. In supporting this, an approach is taken to protect sensitive information and data. In this research, cryptographic techniques were analyzed and tested to secure data in text, which is commonly used as a password using the Blowfish cryptographic algorithm. The goal is to determine how strong, secure, and fast the blowfish algorithm maintains and executes the text data encoding. The research method used is an in-depth theoretical analysis followed by computational experiments using dart programming. Based on the study and testing carried out, the results of this research show that the use of cryptographic techniques with the blowfish algorithm on text data samples can be used as an alternative for securing data. The results show that the blowfish algorithm pattern has security because it uses various techniques to change and present data, supported by an execution time of less than 10 seconds and minimal memory usage of under 5 MB. Therefore, the blowfish algorithm can be used as a cryptographic choice in developing robust and fast text-based data security.

Keywords: Algorithm, Blowfish, Cryptography, Dart, Text

Abstrak

Keamanan data merupakan aspek vital dalam teknologi informasi untuk menjaga integritas dan keaslian data dari sebuah akses yang tidak sah. Dalam menjaga hal tersebut, dilakukan pendekatan untuk melindungi informasi sensitif sebuah data. Pada penelitian ini dilakukan analisis dan pengujian terhadap teknik kriptografi untuk mengamankan data berupa teks yang umum dipakai sebagai *password* dengan kriptografi algoritma *blowfish*. Tujuannya adalah untuk menguji seberapa kuat, aman, dan cepat algoritma *blowfish* menjaga dan mengeksekusi penyandian data teks tersebut. Metode penelitian yang digunakan adalah analisis teoritis secara mendalam dan dilanjutkan dengan eksperimen komputasi menggunakan pemrograman *dart*. Berdasarkan analisis dan pengujian yang dilakukan, hasil penelitian ini menunjukkan bahwa penggunaan teknik kriptografi dengan algoritma *blowfish* pada sampel data teks dapat dijadikan sebagai alternatif dalam mengamankan data. Didapatkan hasil, pola algoritma *blowfish* memiliki keamanan karena dilakukan berbagai teknik untuk mengubah dan menyajikan data, didukung dengan ukuran waktu eksekusi kurang dari 10 detik serta penggunaan memori yang sangat kecil di bawah 5 MB. Oleh karena itu, algoritma *blowfish* dapat dijadikan sebagai salah pilihan kriptografi dalam mengembangkan keamanan data berbasis teks yang kuat dan cepat.

Kata kunci: Algoritma, *Blowfish*, *Dart*, Kriptography, Teks

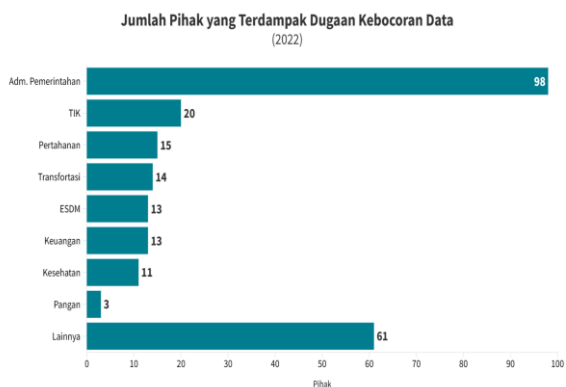
1. PENDAHULUAN

Terbuka dan mudahnya informasi dalam era modern memberikan berbagai akses dan informasi. Teknologi informasi sudah banyak diterapkan di banyak tempat, baik bisnis maupun pelayanan dari berbagai macam institusi maupun instansi lembaga[1]. Seiring dengan itu, pada era digital sekarang ini, isu keamanan data dalam informasi yang tersimpan pada *database* menjadi hal yang penting dan sensitif dalam penerapan sebuah teknologi dan sistem informasi[2]. Laju teknologi yang semakin pesat juga

berdampak pada risiko ancaman terhadap keamanan data yang ada di dalam kegiatan terkomputerisasi[3]. Penyebabnya dipicu karena data dapat dijadikan senjata dan disalahgunakan oleh mereka yang tidak memiliki otoritas, wewenang, dan tanggung jawab. Pada banyak kasus yang berkaitan dengan pencurian data yang terjadi, banyak sekali kerugian materiil dan non-materiil.

Pihak yang tidak bertanggung jawab tersebut juga dapat menggunakan data sebagai alat untuk melakukan tindak

kriminal yang mengarah pada pemerasan terhadap para korbannya. Pengiriman dan penerimaan dalam proses transfer data juga harus terjaga orisinalitas dan dimodifikasi sedemikian rupa untuk mengantisipasi dan melindungi dari kemungkinan bocornya jalur akses pada pengiriman data atau antisipasi hal paling buruk yakni runtuhnya sebuah sistem penyimpanan data. Data dugaan kebocoran data tersebut dapat kita lihat pada Gambar 1 berikut.



Gambar 1. Data Dugaan Kebocoran Data[4]

Pada teknik pengamanan data umumnya dikenal berbagai macam teknik, salah satunya kriptografi. Ini merupakan sebuah disiplin ilmu komputer yang mempelajari teknik untuk melindungi kerahasiaan, integritas, dan autentikasi data[5]. Kriptografi bertugas menjaga keamanan sebuah data menggunakan metode penyisipan data dengan suatu algoritma khusus dengan tujuan data tersebut terlindungi dari pengguna lain. Teknik ini selain mengubah data, dapat juga dipakai untuk melindungi integritas, keaslian serta nirsangkal[6]. Maka kita dapat menarik pernyataan bahwa kriptografi adalah sesuatu yang vital dan penting karena digunakan sebagai upaya mendukung keamanan dan menjaga rahasia pada informasi[7].

Penerapan teknik kriptografi dapat kita temukan pada banyak penelitian lain yang telah dilakukan sebelumnya, khususnya yang menggunakan algoritma *blowfish*. Misalnya pada penelitian yang dilakukan Simanullang dan Silalahi dalam penelitian yang berjudul Algoritma *Blowfish* Untuk Meningkatkan Keamanan *Database MySQL*, yang diketahui bahwa penerapan algoritma *blowfish* pada *database MySQL* dapat dilakukan peningkatan. Selain itu pada penelitian lain yang berjudul Algoritma *blowfish* pada *Watermarking* Video Digital oleh Khairani dan Nurwulan dengan menggunakan alat bantu aplikasi *Visual Basic*, didapatkan jika penggunaan metode algoritma *blowfish* dapat diimplementasikan dengan baik untuk *watermarking* video digital dengan tingkat keamanan yang sangat baik. Penelitian berkaitan dengan algoritma ini banyak menggunakan alat bantu aplikasi *Visual Basic*, seperti pada penelitian lain yang berjudul Implementasi Teknik Enkripsi dan Dekripsi di File Video Menggunakan Algoritma *Blowfish* yang dilakukan oleh Fahriani dan Rosyid. Diketahui bahwa dengan memanfaatkan algoritma ini

implementasi teknik enkripsi berhasil dilakukan yang mengubah file asli agar tidak dapat dibaca.

Dalam penelitian ini penulis akan menggunakan algoritma *blowfish* yang menjadi salah satu alternatif pada teknik kriptografi untuk melindungi data yang bersifat teks sebagai sampel yang diasumsikan sebagai sebuah data sandi atau *password*. Peneliti akan menggunakan bahasa pemrograman *dart* untuk menguji kinerja algoritma *blowfish*. Selain itu di pilihan bahasa pemrograman *dart* karena jika kita melihat berbagai sumber informasi ilmiah, peneliti belum menemukan algoritma ini diterapkan dan diuji coba dengan basis bahasa *dart*. Ini akan menjadi tantangan tersendiri bagi penulis untuk membaca pola dan memahami alur dari konsep algoritma *blowfish* agar bisa diterapkan pada bahasa *dart*.

Kemudian, selain itu pemilihan terhadap algoritma *blowfish* juga didasari karena jika kita melakukan studi literatur, kita akan menemukan banyak kelebihan dari algoritma *blowfish* ini seperti kompatibilitas, efisiensi baik waktu ataupun sumber daya, dan tidak ada lisensi yang diperlukan untuk menggunakannya sehingga legal untuk dipergunakan serta hanya pemilik yang akan mengetahui isi dari *file* yang dimodifikasi dengan teknik ini. Algoritma *blowfish* menerapkan *simetrics key*, *chipper block*, *fistel network* dan *s-box* pada pemrosesan enkripsi dan dekripsi[8]. Lalu kemudian pada domain dan area ilmu kriptografi, algoritma ini banyak dipergunakan karena algoritma *blowfish* kuat, cepat, serta tidak terbentur terhadap lisensi[9]. Untuk itu, dalam dunia kriptografi algoritma *blowfish* menjadi salah satu pilihan yang banyak dipakai oleh pengembang perangkat lunak yang berkebutuhan dan menginginkan algoritma kriptografi yang bersifat terbuka, kuat, dan cepat[10].

Selain itu secara keamanan, hingga saat ini celah yang muncul pada algoritma *blowfish* ini relatif sulit dipecahkan. Karena kemungkinannya jika terdapat kunci yang lemah, namun untuk memecahkan hal tersebut tetap harus melewati proses lain, yakni mencari pola pembentukan kunci terlebih dahulu. Sehingga pesan atau data misalnya data teks yang ingin dibongkar tanpa mengetahui kuncinya, tentu sangat tidak efisien, dan menghabiskan banyak sumber daya[11]. Secara tujuan, dengan kunci yang kuat, algoritma ini memiliki performansi yang sangat optimal sehingga algoritma ini ditujukan untuk penggunaan pada banyak tempat dan jenis pengembangan[12].

Selanjutnya, dalam penelitian ini juga bertujuan untuk mencoba menerapkan algoritma *blowfish* dalam membantu mengamankan data dan informasi pada sebuah data yang dalam penelitian ini kami menggunakan data berupa teks yang diasumsikan sebagai sebuah sandi. Lalu kami melihat cara kerja algoritma *blowfish* sehingga didapatkan analisis keamanan data dan kemungkinan untuk dikembangkan menjadi lebih kuat dan handal, serta sejauh apa efektivitas sumber daya yang dipakai algoritma *blowfish* dalam bekerja. Selanjutnya kami juga akan menganalisis kekuatan

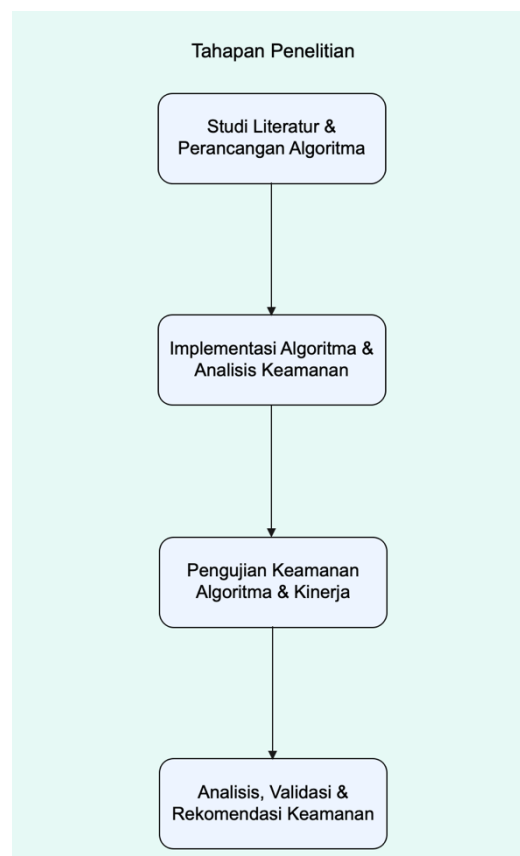
dan kelemahan serta mengukur kinerja algoritma ini terhadap efisiensi waktu eksekusi dan kebutuhan sumber daya pada memori. Sehingga tujuan kami dalam penelitian ini untuk mengukur sejauh apa kinerja algoritma *blowfish* dapat didukung dan ditopang dengan ilmu terapan yang diharapkan dapat memberikan manfaat dan menambah wawasan, dan khazanah pengetahuan. Sehingga ke depan diharapkan terdapat pengembangan lanjutan dalam segala aspek terutama terhadap alat bantu *dart* dan algoritma yang akan dan telah digunakan dalam penelitian ini.

Tentu dilakukan dan diperlukan analisa dan pengujian menggunakan pemrograman. Kami memilih menggunakan bahasa pemrograman *dart*. *Dart* digunakan oleh penulis karena *dart* merupakan bahasa pemrograman yang dirancang untuk dioptimalkan oleh pengguna, dengan memprioritaskan pengembangan dan pengalaman untuk memproduksi kode berkualitas tinggi dan dapat berjalan pada berbagai *platform*[13]. Selain itu, *dart* adalah bahasa pemrograman yang kuat dan modern serta serbaguna. Jika kemudian kita memiliki pengalaman dengan bahasa pemrograman berorientasi objek, atau bergaya pemrograman C/C++ atau sejenisnya, maka bisa dipastikan *dart* dapat dengan cepat dan mudah untuk dipahami. Karena serbaguna, maka dengan *dart* kita bisa menggunakannya untuk menulis apapun, misalnya baris kode perintah dan *server* untuk *backend* hingga sistem *native* misalnya untuk *android*, *ios*, *web*, *mac os*, *windows*, *linux*[14].

2. METODE PENELITIAN

Tahapan penelitian yang penulis gunakan dalam menganalisis algoritma *blowfish* adalah kombinasi antara analisis teoritis dan eksperimen komputasi. Dengan mengumpulkan pengetahuan kriptografi, algoritma dan pemrograman yang dipakai pada algoritma *blowfish*, lalu diikuti dengan melakukan berbagai percobaan susunan algoritma dengan menggunakan data teks sebagai sampel sandi percobaan pada komputasi yang akan diterapkan pada pemrograman *dart* sebagai jawaban dari pengumpulan pengetahuan dan pengujian yang dilakukan. Dalam metode ini, selain dilakukan analisa, yang kami lakukan juga mengukur keakuratan algoritma tersebut dan melihat ukuran waktu eksekusi yang digunakan serta sumber daya yang dihabiskan dalam menjalankan algoritma.

Peneliti memberikan alur cara dan proses kerja dari algoritma *blowfish* ini, dimulai dari inialisasi kunci, lalu *S-Box* dan *P-Box*, berikutnya data di bagi dan di enkripsi melalui serangkaian berbagai macam putaran, kemudian data diambil dan untuk selanjutnya di dekripsi. Proses ini kami jabarkan secara lengkap pada setiap tahapan ke tahapan berikutnya dari setiap tahapan penelitian yang kami lakukan. Untuk itu, kami gambarkan pada Gambar 2 di bawah ini dari setiap tahapan metode penelitian yang dipakai untuk menganalisis kriptografi algoritma *blowfish* dan menerapkannya pada pemrograman *dart*:



Gambar 2. Tahapan Penelitian

2.1 Studi Literatur dan Perancangan Algoritma

Langkah awal dalam merancang algoritma *blowfish* adalah melakukan studi literatur tentang kriptografi, algoritma *blowfish*, enkripsi, dekripsi dan teknik pengamanan data menggunakan algoritma lainnya. Memahami dasar-dasar kriptografi, prinsip-prinsip enkripsi, dan algoritma kriptografi yang ada yang akan membantu dalam menganalisis algoritma *blowfish*. Melakukan pemilihan terhadap algoritma *blowfish* didukung dengan pengumpulan literasi terkait konsep akan kriteria algoritma itu sendiri.

Secara konsep, algoritma ini mencakup pada ekspansi kunci yang dilakukan sebelum enkripsi dan dekripsi dilakukan, dengan kombinasi pada rentang 32 bit hingga 448 bit[15]. Juga menggunakan struktur *Feistel Network* pada saat enkripsi dan dekripsi yang memanfaatkan *subkey* melalui hasil dari prosedur penggandaan kunci atau *key expansion* dengan 16 kali putaran yang dilakukan[16].

Selain itu algoritma *blowfish* dirancang dengan kriteria dan keunggulan seperti berikut ini:

1. *Fast*
Algoritma ini dirancang untuk mengenkripsi data pada mikroprosesor 32 bit dengan kecepatan 26 siklus *clock* per bit.
2. *Compact*
Algoritma ini bisa berjalan pada memori kurang dari 5K
3. *Simple*

Algoritma ini dapat dijalankan dengan metode yaitu penjumlahan, penggeseran, XOR, serta *tables lookup* pada 32 bit.

4. *Variable Secure*

Algoritma ini sangat aman, dengan keamanan dalam tingkat yang bervariasi yang dapat mencapai hingga 448 bit.

Setelah memahami dasar-dasar kriptografi dan analisis keamanan, serta algoritma *blowfish*, berikutnya tahap perancangan algoritma dimulai. Perancangan algoritma juga harus mempertimbangkan efisiensi dalam penggunaan sumber daya, seperti kecepatan enkripsi dan dekripsi, ukuran kunci, dan kompleksitas algoritma.

Ini melibatkan dan diawali dengan inisialisasi dari pengembangan skema kunci, proses enkripsi, dekripsi, serta struktur algoritma secara keseluruhan. Dimulai dari inisialisasi algoritma menggunakan *P-Box* dan *S-Box*. Ini dilakukan untuk menghitung dan menjadikan algoritma sebagai nilai awal, yang akan disubstitusikan agar diubah dan diatur menjadi kunci yang akan dipakai dalam proses perubahan data. Di sini kami menggunakan nilai konstan untuk kami masukkan ke dalam larik. *P-Box*. Ini adalah larik yang memiliki 18 elemen dengan 32 bit. Secara visualisasi, larik tersebut akan tampak seperti ini:

P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12, P13,
P14, P5, P16, P17, P18

Nilai di dalam tersebut lalu diisi dan di-inisialisasikan menggunakan nilai konstanta yang sulit untuk ditebak. Dan setiap elemen tersebut kami lakukan XOR. Proses ini akan terus berulang sehingga semua elemen dalam larik P semuanya terisi. Sebagai gambaran agar Anda mudah memahami proses XOR yang didefinisikan, XOR di sini adalah sebuah operasi berbasis logika yang diterapkan pada nilai dua bit. Dua bit yang kami maksud di sini adalah nilai *boolean*. Misalnya 0 XOR 0 maka nilainya adalah 0. Proses ini dilakukan sebagai bagian dari penggabungan pesan dengan konversi pada enkripsi data yang dimaksudkan sebagai bagian dalam menjaga keamanan data.

Berikutnya adalah inisialisasi untuk larik S-Box. Terdapat larik S-Box berjumlah 4 buah, yang di dalamnya masing-masing berisi nilai konstanta yang juga tidak mudah ditebak dengan isi elemen yang berjumlah 256 elemen. Secara visualisasi larik tersebut akan berbentuk seperti ini:

S1,0, S1,1, S1,2, S1,3, S1,4, ..., ..., ..., ..., S1,255

S2,0, S2,1, S2,2, S2,3, S2,4, ..., ..., ..., ..., S2,255

S3,0, S3,1, S3,2, S3,3, S2,4, ..., ..., ..., ..., S3,255

S4,0, S4,1, S4,2, S4,3, S2,4, ..., ..., ..., ..., S4,255

2.2 Implementasi Algoritma dan Analisis Keamanan

Setelah merancang algoritma secara teoritis, langkah berikutnya adalah mengimplementasikannya dalam bentuk

prototipe pada lingkungan komputasi. Implementasi prototipe memungkinkan peneliti untuk menguji algoritma *blowfish* dalam skenario praktis, dan memastikan bahwa algoritma berfungsi dengan benar dan sesuai dengan desain yang diinginkan. Kami melakukan percobaan dengan skenario praktis dengan mencoba data teks "PASSWORD KU". Algoritma akan membagi menjadi blok terhadap data yang ada. Blok ini dibagi menjadi blok-blok dengan panjang 64 bit, pada skenario ini misalnya pemisahan blok untuk kata "PASSWORD" dan kata "KU".

Setiap blok akan diolah pada serangkaian putaran yang di dalamnya terdapat operasi substitusi, XOR, *bitwise*, dan permutasi. Nilai blok akan ditukar dan digunakan untuk menghitung bagian blok lainnya pada putaran berikutnya. Proses ini akan terus berulang sesuai dengan jumlah kunci yang digunakan.

2.3 Pengujian Keamanan dan Kinerja

Langkah berikutnya adalah melakukan pengujian analisis keamanan terhadap algoritma yang akan dirancang. Ini termasuk mengidentifikasi potensi kelemahan dan kerentanannya terhadap berbagai serangan dan potensi celah manipulasi kriptografi dan potensi lainnya yang dapat terjadi dalam kode komputasi. Analisis keamanan akan membantu dalam mengidentifikasi pada area dimana algoritma perlu diperkuat atau dimodifikasi untuk menjaga tingkat keamanan yang optimal.

Setelah implementasi prototipe selesai, algoritma *blowfish* harus diuji untuk mengevaluasi keamanan dan kinerjanya. Pengujian keamanan melibatkan pengujian algoritma dan pengujian kinerja serta pengukuran waktu eksekusi dan penggunaan sumber daya lainnya untuk memastikan algoritma berjalan dengan efisien.

Skenario yang telah dilakukan mulai dari inisialisasi, dan pembagian blok, maka selanjutnya dilakukan proses operasi logika yang melibatkan pemutaran blok teks yang dienkripsi yang melibatkan operasi XOR, *bitwise*, dan permutasi. Geser kiri hingga geser kanan digunakan untuk mengacak data. Misalnya seperti blok A XOR B = "PASSW XOR "ORD KU" = "P!!QR!!" yang menghasilkan bentuk karakter ASCII. Lalu operasi geser kanan $W \gg 1 = "E\$KT"$ dengan menggeser bit ke kanan dan menggantinya dengan bit baru.

2.4 Analisis, Validasi, dan Rekomendasi

Hasil dari pengujian keamanan dan kinerja dari proses eksekusi algoritma yang didasarkan pada keamanan dan percobaan dalam data berbasis teks memberikan gambaran bahwa dengan panjang kunci yang besar, yang memungkinkan penggunaan kunci dengan panjang hingga 448 bit, maka membuat serangan menjadi lebih sulit karena akan melakukan ruang pencarian yang sangat besar tentunya. Selanjutnya algoritma ini juga menggunakan struktur *feistel* yang akan melakukan cakupan putaran yang

memungkinkan setiap blok yang berisi data akan teracak dan mencegah terjadinya pola yang berulang. Juga didukung dengan substitusi larik membuat algoritma ini sangat aman dalam mencegah serangan.

Lalu secara kecepatan pada setiap operasinya jika dilihat dari operasi yang digunakan algoritma ini menggunakan operasi yang dimiliki XOR dan *bitwise* yang secara umum sangat efisien. Ditopang dengan substitusi larik, maka kebutuhan dan perhitungan selama operasi dilakukan akan sangat berkurang. Dan secara akurasi, setiap putaran yang dilakukan akan mencampur ulang data dengan kunci dan akan memasukkan kembali hasil tersebut ke putaran lain yang berikut-berikutnya. Hal ini menunjukkan bahwa terjadi akurasi enkripsi dengan transformasi data yang variatif.

Ukuran kompleksitas yang terjadi dalam algoritma merupakan parameter kami bahwa algoritma ini sangat baik untuk dipakai dan disisipkan ke dalam pemrograman untuk menjadi perlindungan dalam mengamankan data. Hal itu berdasarkan pada tangkapan penggunaan memori, CPU, dan waktu yang terlampir pada saat algoritma dioperasikan dan dieksekusi. Hal ini yang menjadi ukuran dan parameter yang kami perhatikan yang akan membantu dan mengindikasikan kami dalam memberikan validasi dan rekomendasi bahwa algoritma ini memiliki kecepatan terkait waktu dan memiliki penggunaan sumber daya yang baik saat digunakan dan diterapkan.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Key Expansion, Enkripsi, dan Dekripsi

3.1.1 Key Expansion

Kami menggunakan konstruktor untuk kelas *blowfish* yang digunakan untuk membuat objek dengan menginisialisasi pada tabel berdasarkan kunci yang akan diberikan. Lihat pada Gambar 3(a), konstruktor kelas diinisialisasi. Konstruktor ini akan meng-inisialisasi *_p* dan *_sBoxes* dengan diisi tabel awal lalu kemudian metode *_setupKey* yang terdapat pada Gambar 4(b) akan dipanggil karena akan diteruskan inisialisasinya lebih lanjut dari kunci yang diberikan yang akan dilanjutkan pada *method* yang lain.

```
Blowfish(List<int> key) {
    _p = List.from(_initialP);
    _sBoxes = List.from(_initialS);

    _setupKey(key);
}
```

Gambar 3. Konstruktor Kelas (a)

```
void _setupKey(List<int> key) {
    final keyLength = key.length;
    var dataIndex = 0;
}
```

Gambar 4. Metode Kelas Konstruktor (b)

Algoritma *blowfish* menghasilkan *subkey* dari kunci utama yang diberikan menggunakan proses penggandaan kunci. Kunci utama yang dimasukkan ke dalam algoritma *blowfish*, lalu dipecah menjadi sejumlah *subkey* menggunakan fungsi yang disebut *P-Box*. Pengulangan iteratif dilakukan untuk menghasilkan *subkey* dengan menggunakan mekanisme XOR dan fungsi non-linier khusus. Ini akan mengubah kunci (minimal 32bit, maksimal 448 bit) ke dalam larik (*subkey*) hingga 4168 bit (18x32 bit pada larik P dan 4x256x32 bit pada kotak S, hingga menjadi 33344 bit, dan disimpan dalam larik K. Lalu di-*produce* melalui *subkey* kemudian dihitung. Setelah itu barulah proses berlanjut pada enkripsi dan dekripsi. *Subkey* yang dipakai yakni: Larik P berjumlah 18 (P1, ..., P18) dan 32 bit *subkey*. Selanjutnya kotak S berjumlah 4 dan 32 bit *subkey* yang terdiri entri sebanyak 256 mulai dari (S1,0, ..., S1,255, S2,0, ..., S2,255, S3,0, ..., S3,255, S4,0, ..., S4,255). Selanjutnya *subkey* dihitung dengan:

1. Menginisial larik P1 hingga P4 dengan *string* dari awal dan kotak S, berurutan yang memuat digit *hexadecimal*.

```
InitialP = [
    0x243F6A88,
    0x85A308D3,
    0x13198A2E,
    0x6E85076A,
];
```

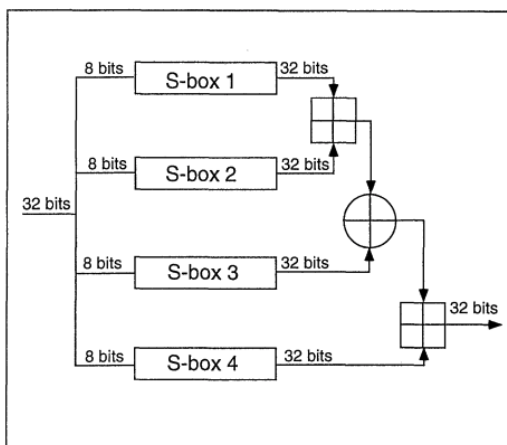
Proses inisialisasi yang dilakukan pada larik P dan kotak S seperti yang terdapat pada Gambar 5.

```
class Blowfish {
    static const List<int> _initialP = [
        0x243F6A88,
        0x85A308D3,
        0x13198A2E,
        0x6E85076A,
    ];
};
```

Gambar 5. Inisialisasi P (InitialP)

2. Lakukan XOR pada P1 menggunakan 32 bit kunci awal, hingga seterusnya dengan semua kunci. Siklus ini dilakukan berulang dan berurutan hingga semua larik P.
3. Lakukan enkripsi yang seluruh string bernilai 0 menggunakan subkey.
4. Ubah P3 dan P4 menggunakan hasil dari proses 3
5. Lalu enkripsi dengan subkey menggunakan blowfish
6. Ubah P3 dan P4 dengan hasil dari proses sebelumnya.
7. Lakukan berurutan untuk mengubah semua elemen larik P dan kotak S.

Pada Gambar 6, terlihat alur langkah dan proses yang dilakukan key expansion.



Gambar 6. Proses Key Expansion

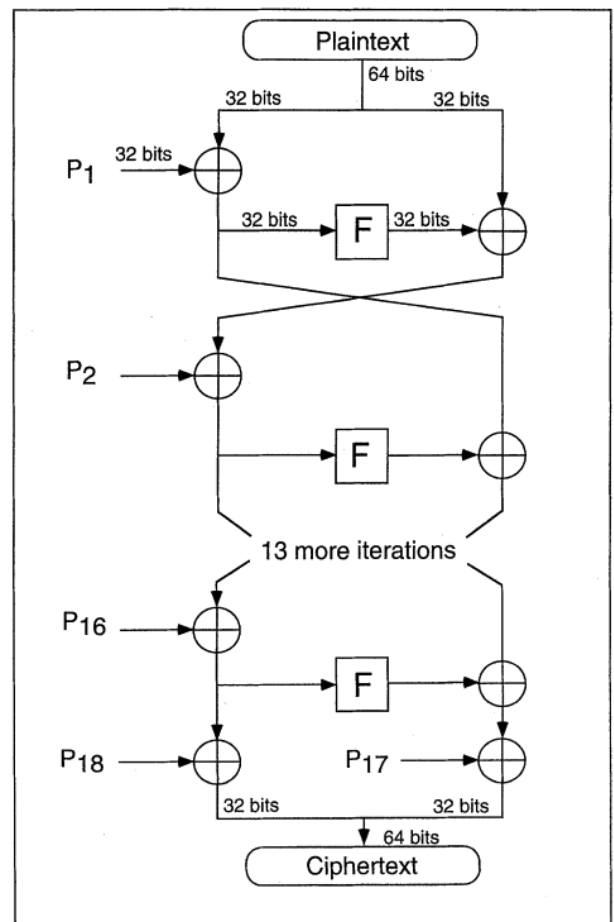
3.1.2 Enkripsi

Proses enkripsi *blowfish* melibatkan pengulangan 16 putaran dengan menggunakan *subkey* yang dihasilkan dari proses penggandaan kunci. Setiap putaran terdiri dari operasi XOR, substitusi, dan permutasi. Data yang akan dienkripsi dibagi menjadi blok-blok 64-bit. Blok data melewati 16 putaran enkripsi. Dalam setiap putaran enkripsi, blok data dipecah ke dua bagian, yaitu kiri (L) dan kanan (R). X data akan melewati fungsi putaran 16 kali putaran (*Feistel Network*) 64 bit, seperti ini:

1. Buat X ke dalam dua bagian, XL, dan XR. Setiap bagian masing-masing memiliki 32 bit.

2. Selanjutnya membuat putaran sebanyak 16 kali, dan ubah data XL ke XR.
 for $i = 1$ to 16:
 $XL = XL \text{ XOR } P_i$
 $XR = F(XL) \text{ XOR } XR$
3. Berikutnya putar kembali XL dan XR sehingga pertukaran sebelumnya menjadi dibatalkan.
4. Kemudian, kita lakukan:
 $XR = XR \text{ XOR } P_{17}$
 $XL = XL \text{ XOR } P_{18}$
5. Dan kembali gabungkan XL dan XR agar didapatkan *chiphertext*.

Secara lengkap alur proses enkripsi menggunakan algoritma *blowfish*, dipetakan pada Gambar 7.



Gambar 7. Blok Alur Diagram Enkripsi Algoritma Blowfish

Dari gambar tersebut diketahui bahwa *plaintext* yang berupa data masukan yang akan dienkripsi dimasukkan ke dalam algoritma. Lalu akan dibagi menjadi blok-blok data yang biasanya berukuran 64-bit (8 byte) masing-masing. Selanjutnya dilakukan Inisialisasi kunci dan subkunci dilakukan menggunakan kunci rahasia yang diberikan. Subkunci ini akan digunakan dalam setiap putaran enkripsi. Proses enkripsi sendiri terdiri atas 16 putaran, di mana setiap putaran melibatkan operasi XOR, substitusi dengan *S-boxes*, dan permutasi dengan *P-boxes*. Setiap putaran mengubah blok data secara berulang. Setelah semua putaran selesai dilakukan, maka blok data hasil enkripsi adalah hasil

gabungan dari bagian kiri dan kanan yang telah diubah selama putaran enkripsi. *Ciphertext* atau data yang terenkripsi merupakan hasil akhir dari proses enkripsi dan dapat kita dikeluarkan dari proses algoritma.

3.1.3 Dekripsi

Proses dekripsi menggunakan algoritma *blowfish* adalah kebalikan dari proses enkripsi. Data terenkripsi dibagi menjadi blok-blok 64-bit. Setiap blok data melewati 16 putaran dekripsi, dengan *subkey* yang digunakan dalam urutan terbalik dari putaran enkripsi. Dalam setiap putaran dekripsi, operasi XOR dilakukan antara bagian kiri (L) dan *subkey* yang sesuai dari putaran dekripsi saat ini. Hasil XOR kemudian melewati fungsi *Feistel*, yang melibatkan operasi substitusi dan permutasi pada bagian kanan. Sebelah kiri (L) dan sebelah kanan (R) diperbarui berdasarkan hasil fungsi *Feistel*. Setelah 16 putaran dekripsi selesai, blok data diubah kembali menjadi bentuk aslinya. Proses dekripsi dilakukan dengan menggunakan *subkey* yang dihasilkan dalam urutan terbalik.

for $i = 1$ to 16:

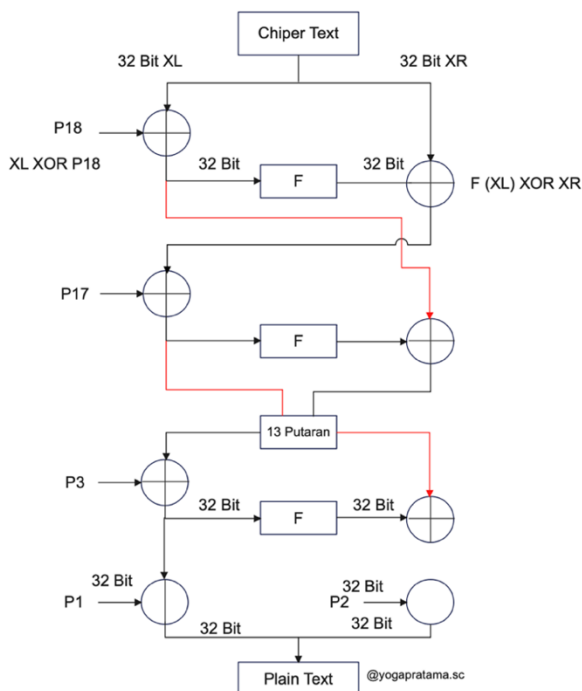
$$XR_i = XL_{i-1} \text{ XOR } P_{19-i}$$

$$XL_i = F[XR_i] \text{ XOR } XR_{i-1}$$

$$XL_{17} = XR_{16} \text{ XOR } P_1$$

$$XR_{17} = XL_{16} \text{ XOR } P_2$$

Secara lengkap alur proses dekripsi menggunakan algoritma *blowfish*, dapat dilihat pada Gambar 8.



Gambar 8. Blok Alur Diagram Dekripsi Algoritma *Blowfish*

Melihat gambar di atas maka diketahui *ciphertext* adalah data yang terenkripsi yang akan di-dekripsi lalu dimasukkan ke dalam algoritma. Kemudian subkunci yang sama yang digunakan dalam proses enkripsi, tetapi diterapkan secara terbalik, akan digunakan untuk inialisasi kunci dan subkunci dekripsi. Dalam proses dekripsi juga terdiri atas 16 putaran yang sangat mirip dengan proses enkripsi. Setiap putaran melibatkan operasi XOR, substitusi dengan *S-boxes*, dan dilakukan permutasi dengan *P-boxes*, tetapi dengan subkunci yang diterapkan dilakukan secara terbalik. Kemudian setelah semua putaran dekripsi selesai, blok data hasil dekripsi adalah hasil gabungan dari bagian kiri dan kanan yang telah diubah selama putaran dekripsi. Disini data asli atau *plaintext* merupakan hasil akhir dari proses dekripsi dan dapat dikeluarkan dari algoritma.

3.2 Pengujian dan Pengukuran Kinerja *Blowfish*

3.2.1 Inialisasi Kunci

Algoritma *blowfish* memulai dengan meng-inialisasi dua tabel, yaitu *P-array* (*Plain Array*) dan *S-boxes* (*Substitution boxes*). *P-array* adalah larik berisi 18 elemen 32-bit yang akan digunakan selama proses enkripsi dan dekripsi. *S-boxes* terdiri dari 4 larik, masing-masing berisi 256 elemen 32-bit. *S-boxes* digunakan dalam langkah enkripsi yang disebut fungsi F.

3.2.2 Pembentukan *Key Expansion*

Kunci awal yang diberikan oleh pengguna (128-bit) akan digunakan untuk membentuk subkunci yang lebih banyak. Proses ini menggunakan *P-array* dan *S-boxes* yang telah diinisialisasi sebelumnya. Kunci 128-bit akan dipecah menjadi beberapa bagian dan akan di-XOR dengan *P-array* secara berulang untuk membentuk subkunci. Pembentukan *key expansion* pada proses komputasi dapat kita lihat pada Gambar 9 berikut.

```
for (var i = 0; i < _p.length; i += 2) {
    _encryptBlock(dataBlock);
    _p[i] = _bytesToInt(dataBlock, 0);
    _p[i + 1] = _bytesToInt(dataBlock, 4);
}

_sBoxes = List.from(_initialS);

for (var i = 0; i < _sBoxes.length; i += 2) {
    _encryptBlock(dataBlock);
    _sBoxes[i] = _bytesToInt(dataBlock, 0);
    _sBoxes[i + 1] = _bytesToInt(dataBlock, 4);
}
```

Gambar 9. Pembentukan *Key Expansion*

3.2.3 Enkripsi

Data yang ingin dienkripsi akan dibagi menjadi blok-blok 64-bit (8-byte). Data blok akan di-XOR dengan subkunci. Blok data yang di-XOR akan masuk ke fungsi F. Fungsi F mengambil blok 32-bit dan menghasilkan blok 32-bit sebagai *output* dengan melakukan operasi *bitwise* menggunakan *S-boxes* dan operasi matematika lainnya. Hasil dari fungsi F akan di-XOR dengan blok data lain dan kemudian data akan di-*switch* (pertukaran) sebelum masuk ke putaran berikutnya. Semua proses ini akan diulang sebanyak 16 putaran.

Lihat pada Gambar 10 di bawah ini untuk melihat gambaran komputasi dari pembentukan enkripsi yang dilakukan.

```
for (var i = 0; i < 16; i++) {
  left ^= _p[i];
  right ^= _encryptF(left);
  right ^= _p[i + 1];
  left ^= _encryptF(right);
}
```

Gambar 10. Pembentukan Enkripsi

3.2.4 Dekripsi

Proses dekripsi mirip dengan proses enkripsi, namun subkunci diaplikasikan dalam urutan terbalik. Data yang dienkripsi akan di-XOR dengan subkunci terakhir dan kemudian masuk ke fungsi F. Fungsi F yang dilakukan dan dibentuk dapat dilihat pada Gambar 11 berikut.

```
int _encryptF(int x) {
  var name1 = _sBoxes[(x >> ..) & 0xFF];
  var name2 = _sBoxes[(x >> ..) & 0xFF];
  var name3 = _sBoxes[(x >> ..) & 0xFF];
  var name4 = _sBoxes[x & 0xFF];
  return ((name1 << ..) | (name2 << ..) | (name3 << ..) | name4);
}
```

Gambar 11. Pembentukan Fungsi F

Hasil dari fungsi F akan di-XOR dengan subkunci lainnya dan setelah 16 putaran, data akan terdekripsi. Teks data dienkripsi hanya dapat di-dekripsi kembali dengan menggunakan kunci yang sama yang digunakan untuk enkripsi. Pada kode yang diberikan, Anda dapat melihat

proses inialisasi kunci, *key expansion*, enkripsi, dekripsi, diimplementasikan dalam bentuk kelas *blowfish*. Proses enkripsi dan dekripsi terjadi pada fungsi *_encryptBlock* dan *_decryptBlock* yang menggunakan fungsi F *_encryptF* untuk menghasilkan blok data terenkripsi dan terdekripsi. Untuk menggambarkan komputasi dekripsi dapat dilihat pada Gambar 12 berikut.

```
for (var i = 16; i >= 0; i--) {
  left ^= _p[i + 1];
  right ^= _encryptF(left);
  right ^= _p[i];
  left ^= _encryptF(right);
}
```

Gambar 12. Pembentukan Dekripsi pada Komputasi

3.2.5 Hasil Pengujian

Hasil dari algoritma *blowfish* adalah data yang telah dienkripsi dan dapat dikirimkan dengan aman ke pihak lain. Enkripsi yang telah dilakukan bisa terdekripsi kembali melalui *key* yang sama yang digunakan untuk enkripsi. Pada kode kita dapat melihat proses inialisasi kunci, *key expansion*, enkripsi, dekripsi, diimplementasikan dalam bentuk kelas *blowfish*. Tahapan ini terjadi pada fungsi *_encryptBlock* dan *_decryptBlock* yang menggunakan fungsi F *_encryptF* untuk menghasilkan blok data terenkripsi dan terdekripsi. Secara lengkap hasil pengujian pada komputasi dapat dilihat pada Gambar 13.

```
Launching lib/main_blowfish.dart on macOS in debug mode... main_blowfish.dart:1
Connecting to VM Service at ws://127.0.0.1:53025/szxQKP7JGRM=/ws
Flutter: Plaintext: [78, 111, 119, 32, 105, 115, 32, 116, 104, 101, 32, 116, 1
05, 109, 101, 32, 102, 111, 114, 32, 97, 108, 108, 32, 103, 111, 111, 100, 32,
109, 101, 110]
Flutter: Encrypted: [126, 105, 177, 46, 172, 226, 50, 179, 76, 244, 16, 12, 7
9, 74, 223, 191, 236, 94, 134, 143, 134, 79, 120, 10, 48, 191, 104, 40, 185, 2
34, 95, 88]
Flutter: Decrypted: [78, 111, 119, 32, 105, 115, 32, 116, 104, 101, 32, 116, 1
05, 109, 101, 32, 102, 111, 114, 32, 97, 108, 108, 32, 103, 111, 111, 100, 32,
109, 101, 110]
```

Gambar 13. Hasil Komputasi

3.2.6 Analisis dan Validasi Metode Algoritma

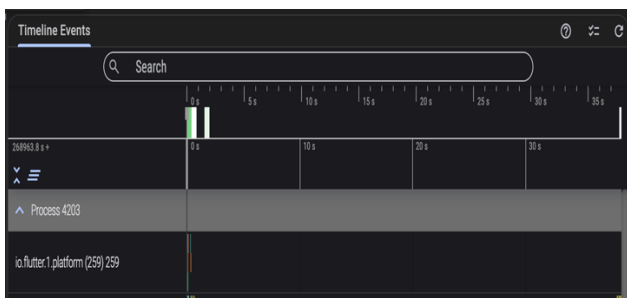
Metode enkripsi dan dekripsi data teks melalui seleksi kunci dalam penggunaan algoritma *blowfish* adalah kunci dalam penggunaan algoritma *blowfish* harus diperhatikan dipilih dengan cermat untuk memastikan kekuatan keamanan yang optimal. Kunci dapat dieksekusi secara acak atau menggunakan teknik pengambilan kunci lain yang aman yang dipilih. Proses enkripsi data teks menggunakan *blowfish* data dalam *database* dienkripsi menggunakan

algoritma *blowfish* dengan menggunakan kunci yang telah dipilih.

Data dibagi menjadi blok-blok 64-bit, dan setiap blok dienkripsi secara independen menggunakan *subkey* yang dihasilkan dari proses penggandaan kunci. Proses dekripsi data dalam *database* menggunakan *blowfish* proses dekripsi data dalam *database* dilakukan dengan menggunakan algoritma *blowfish* yang sama dengan menggunakan *subkey* yang dihasilkan dalam urutan terbalik. Setiap blok data dienkripsi secara independen dan diubah kembali menjadi bentuk aslinya. Mempertahankan integritas dan autentikasi data dalam *database*. Selain mengenkripsi data, teknik kriptografi dengan algoritma *blowfish* juga dapat digunakan untuk mempertahankan integritas dan keaslian data yang dihasilkan dari data terenkripsi.

3.2.7 Analisis dan Validasi Hasil Pengukuran Kinerja

Metode pengujian dan pengukuran terhadap kinerja dilakukan untuk mengukur kinerja teknik kriptografi yang dilakukan menggunakan algoritma *blowfish*, termasuk waktu eksekusi dan kebutuhan sumber daya seperti penggunaan *CPU* dan memori. Pengukuran waktu eksekusi diukur dengan mengamati waktu yang diperlukan pada saat proses enkripsi dan dekripsi data. Kebutuhan sumber daya seperti penggunaan *CPU*, memori, dan penyimpanan juga dievaluasi untuk memahami tingkat efisiensi teknik kriptografi yang diusulkan dalam penggunaan keamanan pada pengiriman data. Selain itu penggunaan memori dan waktu eksekusi yang cukup cepat juga memberikan nilai tambah pada algoritma ini. Secara lengkap dapat dilihat berdasarkan tangkapan layar *debug* mengenai hasil pengujian terhadap waktu eksekusi yang digunakan dapat dilihat pada Gambar 14.



Gambar 14. Tangkapan Waktu Saat Eksekusi

3.2.8 Analisis dan Validasi Keamanan Algoritma

Selanjutnya dari segi keamanan algoritma *blowfish* telah diuji dalam sampel pengujian dan algoritma ini termasuk sebagai algoritma yang kuat. Parameter ini diambil dari proses operasi-operasi logika yang terjadi di dalam algoritma tersebut. Dimana proses tersebut diantaranya operasi geser kanan, geser kiri, *bitwise*, XOR, substitusi blok, maupun logika *boolean*. Namun, masih ada kelemahan seperti rentan terhadap manipulasi dengan kunci yang sudah ada juga masih perlu diperhatikan. Secara eksekusi, efisiensi dan kinerja algoritma *blowfish* dalam

memanipulasi data memiliki kecepatan yang tinggi dan efisiensi dalam penggunaan sumber daya. Algoritma ini dapat diimplementasikan dengan baik dalam pengamanan data pada data teks.

4. KESIMPULAN

Berdasarkan hasil penerapan bahwa keamanan dan keefektifan teknik kriptografi algoritma *blowfish* telah dianalisis kekuatannya pada kode yang telah disusun dan diterapkan, lalu dievaluasi serta dikur agar memperoleh rekomendasi dan hasil yang baik dan akurat. Peneliti menggunakan batasan data sampel berupa teks yang dikategorikan sebagai sebuah *password*. Hasil pengukuran didapatkan bahwa penggunaan waktu eksekusi di bawah 10 detik dan penggunaan memori kurang dari 5 MB, menjadi ukuran bahwa waktu eksekusi dan memori yang digunakan cukup efektif.

Lalu secara keamanan algoritma ini adalah algoritma yang kuat, juga memiliki keamanan yang baik dan menawarkan efisiensi yang tinggi dalam penggunaannya. Karena data yang dikirimkan akan melewati berbagai tahapan perubahan sebelumnya hingga pada akhirnya data sudah dikirim. Hal ini menunjukkan keefektifan teknik kriptografi dalam mengamankan data menggunakan algoritma ini efisien dan andal.

Dengan demikian diharapkan hasil penelitian ini kiranya dapat memberikan implikasi dalam pengembangan keamanan data dengan mempertimbangkan penggunaan teknik kriptografi dengan algoritma *blowfish* dan dapat menjadi salah satu pilihan dalam membangun atau mengembangkan sistem berbasis teknologi informasi, khususnya menggunakan pemrograman *dart*. Diharapkan pada penelitian berikutnya, penelitian lainnya yang membahas keamanan data, baik jenis data ataupun jenis algoritmanya menggunakan pemrograman *dart* dapat terus dilakukan, sebab didukung dengan kinerja yang baik dari algoritma *blowfish* dan ke depannya tentu penelitian lainnya juga dapat mengembangkan kekuatan algoritma yang sudah ada dan meminimalkan celah yang ada pada algoritma ini.

DAFTAR PUSTAKA

- [1] Y. Pratama and T. Sutabri, "Service Operation ITIL V3 Pada Analisis dan Evaluasi Layanan Teknologi Informasi," *Nuansa Inform.*, vol. 17, pp. 169–178, 2023, [Online]. Available: <https://journal.uniku.ac.id/index.php/ilkom/article/view/7233%0Ahttps://journal.uniku.ac.id/index.php/ilkom/article/download/7233/3490>.
- [2] P. P. Santoso *et al.*, "Systematic literature review: Comparison study of symmetric key and asymmetric key algorithm," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 420, no. 1, 2018, doi: 10.1088/1757-899X/420/1/012111.
- [3] T. Sutabri, *Konsep Sistem Informasi*, vol. 3, no. 1. Yogyakarta, 2012.

- [4] R. Mustajab, "BSSN: Ada 311 Kasus Kebocoran Data di Indonesia pada 2022," *dataindonesia.id*, 2023. <https://dataindonesia.id/internet/detail/bssn-ada-311-kasus-kebocoran-data-di-indonesia-pada-2022>.
- [5] S. Suhandinata, R. A. Rizal, D. O. Wijaya, P. Warren, and Srinjiwi, "Analisis Performa Kriptografi Hybrid Algoritma RSA," *Jurteksi*, vol. VI, no. 1, pp. 1–10, 2019.
- [6] D. A. Meko, "Perbandingan Algoritma DES, AES, IDEA Dan Blowfish dalam Enkripsi dan Dekripsi Data", *j. teknologi terpadu*, vol. 4, no. 1, Jul 2018.
- [7] I. A. W. Arnawa, P. E. W. Hary, and A. A. G. B. Putra, "Perbandingan Waktu Enkripsi antara Metode Electronic Codebook (ECB) dan Chipher Block Chaining (CBC) dalam Algoritma Blowfish," *J. Ilmu Komput. Indones.*, vol. 5, no. 1, pp. 50–54, 2020, [Online]. Available: <https://ejournal-pasca.undiksha.ac.id/index.php/jik/article/download/3056/1723>.
- [8] N. Fahriani and H. Rosyid, "Implementasi Teknik Enkripsi dan Dekripsi di File Video Menggunakan Algoritma Blowfish," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 6, p. 697, 2019, doi: 10.25126/jtiik.2019661465.
- [9] N. Fahriani and I. Kurniawati, "Keamanan Data Pasien dengan Algoritma Blowfish pada HOTSPOT," *J. Comput. Sci. Informatics Eng.*, vol. 5, no. 2, pp. 140–148, 2021, doi: 10.29303/jcosine.v5i2.416.
- [10] M. Khairani and N. Nurwulan, "Algoritma Blowfish Pada Watermarking Video Digital," *JURIKOM (Jurnal ...*, vol. 5, no. 4, pp. 357–361, 2018, [Online]. Available: <http://ejournal.stmik-budidarma.ac.id/index.php/jurikom/article/view/842>.
- [11] H. G. Simanullang and A. P. Silalahi, "Algoritma Blowfish Untuk Meningkatkan Keamanan Database Mysql," *Method. J. Tek. Inform. dan Sist. Inf.*, vol. 4, no. 1, pp. 10–14, 2018, doi: 10.46880/mtk.v4i1.58.
- [12] S. Retno and N. Hasdyna, "Analisis Kinerja Algoritma Honey Encryption Dan Algoritma Blowfish Pada Proses Enkripsi Dan Dekripsi," *TECHSI - J. Tek. Inform.*, vol. 10, no. 2, p. 82, 2018, doi: 10.29103/techsi.v10i2.858.
- [13] Google, "Dart overview | Dart," *Dart Dev*, 2011. <https://dart.dev/overview>.
- [14] J. Sande, *Dart Apprentice: Beyond the Basics*. Kodeco, 2022.
- [15] M. H. T. A. Thoriq, A. I. H. Asep, and P. N. S. Puspita, "Data Encryption Pada File Video Menggunakan Algoritma Blowfish Berbasis Android," *Informatics Digit. Expert*, vol. 4, no. 1, pp. 33–39, 2022, doi: 10.36423/index.v4i1.880.
- [16] M. Muhathir, "Perbandingan Algoritma Blowfish Dan Twofish Untuk Kriptografi File Gambar," *J. Informatics Telecommun. Eng.*, vol. 2, no. 1, p. 23, 2018, doi: 10.31289/jite.v2i1.1673.



ANALISA KINERJA APLIKASI DIGITAL FORENSIK *AUTOPSY* UNTUK PENGEMBALIAN DATA MENGGUNAKAN METODE *NIST SP 800-86*

Dedek Julian¹, Tata Sutabri²

^{1,2}Magister Teknik Informatika, Universitas Bina Darma
Palembang, Sumatera Selatan, Indonesia 30111

dedek.julian99@gmail.com, tata.sutabri@binadarma.ac.id

Abstract

One example of digital crime that often occurs is data theft, such as transaction information, and important company data. The thief will delete files to remove traces so that it is necessary to search for and restore data that has been deleted to be used as digital evidence. This activity is usually called digital forensics. Paid digital forensic applications are sold at quite expensive prices, so one alternative is Autopsy, which is an open source based investigation application that can restore data. This research aims to analyze the performance of the autopsy application in returning 70 files including documents, videos and images as digital evidence based on the crime case scenario of data theft with formatted flash disks. NIST SP 800-86 was chosen as the research method because it has simple stages and is in accordance with the research theme. The stages in this method start from collecting evidence, analyzing the contents of the flash disk with the autopsy application, searching for and returning the found files, to validating the files with hash compare. The analysis report shows that the autopsy application succeeded in returning 81.42% of the data that had been deleted and could be used as evidence based on the crime case scenario that had been created. The files that were successfully returned were 10 DOCX, 10 XLSX, 10 PDF, 6 TXT, 1 MP3, 10 MP4, and 10 PNG.

Keywords: *Autopsy, Cyber Digital, Digital Proof, Forensics, NIST*

Abstrak

Salah satu kasus kejahatan digital yang kerap terjadi adalah pencurian data, seperti informasi transaksi, hingga data penting milik perusahaan, pelaku pencurian akan menghapus berkas untuk menghilangkan jejak sehingga perlu dilakukan pencarian dan pengembalian data yang telah dihapus untuk dijadikan bukti digital, kegiatan ini biasa disebut digital forensik. Aplikasi digital forensik berbayar dijual dengan harga yang cukup mahal, sehingga salah satu alternatifnya adalah *autopsy*, yang merupakan aplikasi investigasi berbasis *open source* dan dapat melakukan pengembalian data. Penelitian ini bertujuan untuk menganalisis kinerja aplikasi *autopsy* dalam melakukan pengembalian 70 berkas dengan jenis dokumen, video dan gambar sebagai bukti digital berdasarkan pada skenario kasus kejahatan pencurian data dengan *flashdisk* yang telah diformat. *NIST SP 800-86* dipilih sebagai metode penelitian karena memiliki tahapan yang sederhana dan sesuai dengan tema penelitian, tahapan dalam metode tersebut dimulai dari pengumpulan barang bukti, melakukan analisa isi *flashdisk* dengan aplikasi *autopsy*, mencari dan mengembalikan berkas temuan, hingga memvalidasi berkas dengan *hash compare*. Laporan hasil analisis menunjukkan bahwa aplikasi *autopsy* berhasil mengembalikan sebanyak 81,42% dari data yang telah dihapus dan dapat dijadikan bukti berdasarkan skenario kasus kejahatan yang telah dibuat, berkas yang berhasil dikembalikan yaitu 10 *DOCX*, 10 *XLSX*, 10 *PDF*, 6 *TXT*, 1 *MP3*, 10 *MP4*, dan 10 *PNG*.

Kata kunci: *Autopsy, Bukti Digital, Forensik, Kejahatan Digital, NIST*

1. PENDAHULUAN

Era teknologi informasi membawa banyak perubahan, diantaranya merupakan dampak baik yaitu kemudahan serta kenyamanan untuk terus produktif dengan memanfaatkan teknologi. Namun tak dapat dipungkiri bahwa dampak negatif juga ikut bertumbuh, seperti kasus kejahatan digital atau *cyber crime*, yang merupakan tindak kejahatan dunia

maya seperti pembajakan program komputer, kegiatan *cracking*, *carding*, penyebaran hal berbau pornografi, pembobolan bank, dan berbagai kejahatan yang lainnya [1]. *Cyber crime* sendiri semakin populer seiring dengan perkembangan teknologi, bahkan kerap kali dampak negatif tersebut terus dilakukan melalui berbagai cara yang tidak terduga. Untuk itu, diperlukan pengamanan teknologi

informasi yang bertujuan untuk meyakinkan integritas, kelanjutan, dan kerahasiaan dari pengolahan data [2]. Sementara itu, setiap perusahaan mempunyai sistemnya masing-masing, sistem sendiri dapat diartikan sebagai jaringan kerja prosedur-prosedur yang saling berhubungan [3], salah satu sistem yang bergantung dengan teknologi biasa disebut sebagai sistem informasi, yang mengandung berbagai data untuk kepentingan perusahaan. Sehingga, hal-hal seperti mencuri data rahasia perusahaan juga dapat dikategorikan sebagai kasus kejahatan digital, karena melanggar aturan dari perusahaan, dan dapat merugikan perusahaan. Pencurian data yang dilakukan seorang kriminal dapat berupa identitas nasabah pada perusahaan perbankan [4], data tersebut digunakan untuk mengambil keuntungan pribadi pelaku, contoh kasus pencurian data lainnya seperti data konsumen dari anak perusahaan *Lion Air* yakni *Malindo Air* dan *Thai Lion Air* yang juga mengalami kebocoran sebanyak 21 juta data penumpang [5]. Pencurian data seperti ini dapat diantisipasi dengan meningkatkan keamanan sistem, dan apabila telah terjadi mesti dilakukan investigasi untuk mengusut pelaku kejahatan tersebut.

Dalam mengungkapkan kasus kejahatan digital, maka diperlukan bukti-bukti digital sebagai acuan di ranah hukum, sementara untuk menghindari kejahatan sejenis ini maka diperlukan tindakan seperti menjaga perangkat dalam mode isolasi [6]. Sebuah teknik yang menerapkan analisis dan penyidikan komputer sehingga memungkinkan seorang penyidik mendapatkan barang bukti digital dari komputer biasa disebut sebagai komputer forensik atau digital forensik [7]. Terdapat dua jenis teknik pengangkatan barang bukti atau forensik, yakni *dead forensik* yang membutuhkan data yang disimpan secara permanen dalam perangkat media penyimpanan, biasanya *harddisk*, dan *live forensik* melibatkan data berjalan pada sistem atau data volatile yang biasanya tersimpan pada *RAM* atau transit pada jaringan [8]. Analisis forensik akan memberikan detail untuk membantu para penyelidik dan lembaga investigasi memecahkan dan menghubungkan kasus-kasus dengan kejahatan yang dilaporkan. Salah satu kegiatan dalam digital forensik yang sering dilakukan adalah *file recovery* atau pemulihan file, yang mana berkas yang telah dihapus oleh pengguna dapat dikembalikan agar bisa diolah lebih lanjut, sebagaimana dijelaskan dalam penelitian [9] bahwa proses *delete*/menghapus suatu berkas bukan berarti menghilangkan data tersebut secara permanen dari media penyimpanan, akan tetapi kegiatan tersebut dapat diartikan sebagai pemberitahuan kepada komputer bahwa ruang yang ditempati data tersebut telah tersedia untuk ditimpa/diisi oleh data yang lain. Sehingga masih memungkinkan untuk dilakukan pengembalian berkas yang telah dihapus dari media penyimpanan meskipun telah diformat.

Untuk dapat melihat apakah berkas yang telah dipulihkan masih bersifat utuh dan merupakan berkas yang sama dengan file asli, maka perlu dilakukan suatu langkah validasi dengan cara mencocokkan *file hash* dari berkas

yang asli dengan berkas yang telah dipulihkan. Hingga saat ini, salah satu algoritma *hash* yang paling populer adalah *Message-Digest 5* atau *MD5*, yang merupakan suatu fungsi *hash* kriptografi dan digunakan untuk melakukan pemeriksaan integritas *file* dalam berbagai situasi [10]. Untuk menjalankan aktivitas validasi *hash* tersebut dapat dilakukan dengan memanfaatkan penggunaan aplikasi *hash compare* dari *securityxplored*. Komparasi *hash* ini akan dijalankan setelah semua berkas berhasil dipulihkan.

Salah satu metode yang umum digunakan untuk melakukan analisis forensik adalah metode dari NIST (*National Institute of Standards and Technology*), yaitu *NIST SP 800-86*, NIST sendiri merupakan suatu lembaga yang mengembangkan standar, panduan, dan persyaratan minimum untuk menyediakan keamanan informasi yang cukup bagi tiap aset serta pihak yang mempunyai kemampuan di bidang *digital forensic*, metode yang dikembangkan oleh NIST ini umumnya digunakan oleh pemerintah pusat di Amerika, namun tidak menutup kemungkinan dapat diimplementasikan juga oleh organisasi seperti akademisi, badan penyidik swasta dan lainnya [11]. Salah satu penelitian yang menerapkan metode ini adalah [9], yang melakukan uji coba pengembalian data, dimana hasil tertinggi yang diperoleh yaitu 100% untuk 20 dokumen, dan 90% untuk berkas dengan tipe gambar. Penelitian lain seperti [6], juga menerapkan metode NIST untuk pengembalian data dengan hasil yaitu *FTK Imager* memperoleh nilai 100% dan *Autopsy* memperoleh nilai 70%. Sementara itu, penelitian yang dilakukan oleh Riadi, dkk. [12] menerapkan metode yang lain, yaitu *National Institute of Justice* dalam melakukan pengembalian data dari *SSD* yang di implementasi *shadow defender* dengan aplikasi *X-ways forensics*, keberhasilan restorasi *file* hanya 28,7%. Sementara penelitian lainnya [13] menerapkan metode yang sama dalam melakukan pengembalian data menggunakan aplikasi *MOBILedit*, *Wondershare dr Fone*, dan *Belkasoft* dengan salah satu kesimpulannya menyatakan bahwa aplikasi forensik yang digunakan tidak cukup baik untuk mengembalikan data gambar, video dan berkas dokumen. Metode *NIJ* menerapkan 5 tahapan penelitian, sementara metode *NIST SP 800-86*, menerapkan langkah yang lebih sederhana yaitu dalam 4 tahapan, sehingga metode *NIST SP 800-86* dipilih karena merupakan metode yang dikembangkan langsung oleh lembaga pengembang standar dan panduan teknologi terutama, metode tersebut juga dapat diterapkan dengan lebih sederhana hanya dalam 4 tahapan, lebih singkat dan cocok dengan tema penelitian seperti yang pernah diterapkan juga dalam penelitian-penelitian sebelumnya.

Selain pemilihan metode yang tepat, penggunaan aplikasi yang sesuai juga dapat mendukung jalannya proses pemulihan data dengan lebih baik, aplikasi yang dapat digunakan untuk melakukan analisis forensik dan pengembalian data juga beragam, mulai dari yang berbayar hingga yang bersifat *open source*, seperti *Autopsy*. Meskipun bersifat *open source*, kinerja dari *tools* ini dapat

bersaing dengan aplikasi lain yang sejenis dengan harga tinggi. Seperti jika digunakan untuk menggali aktivitas transaksi dompet digital, *autopsy* dapat mengungguli aplikasi *Belkasoft Evidence Center*, dengan temuan sebanyak 8 aktivitas transaksi, sementara *Belkasoft Evidence Center* dengan temuan sebanyak 7 aktivitas transaksi [14]. Pada penelitian sebelumnya yang telah dilakukan oleh [15], dilakukan perbandingan untuk mengukur kinerja dari 3 aplikasi forensik untuk pengembalian data dengan media SSD yaitu *Autopsy*, *Belkasoft*, dan *Testdisk*, hasil akhir penelitian menunjukkan bahwa *Persentase recovery TRIM disable* dengan menggunakan aplikasi *Autopsy* dan *Testdisk* adalah 100% sehingga dapat menemukan barang bukti dan menjaga integritas dari barang bukti tersebut. Pada penelitian tersebut dapat dilihat bahwa aplikasi *autopsy* mempunyai potensi yang baik terutama untuk kategori kegiatan *data recovery* dengan media SSD.

Untuk itu, penelitian ini akan melakukan analisa kinerja dari aplikasi *autopsy* dalam mencari dan mengembalikan data yang telah dihapus dari media penyimpanan berupa *flashdisk*. Instrumen penilaian akan dilakukan dengan melihat seberapa banyak *file* yang dapat dikembalikan dengan aplikasi tersebut dan dinyatakan identik oleh aplikasi *Hash Compare*. Dalam penelitian sebelumnya, seperti [9] dilakukan uji coba menggunakan *file type JPG, PNG, DOCX, dan PDF*, maka pada penelitian ini akan dilengkapi dengan total 70 berkas sebagai bahan uji coba, yaitu berkas dengan ekstensi *DOCX, XLSX, MP3, MP4, TXT, PDF dan PNG* masing-masing sebanyak 10 berkas. Penelitian juga akan dilakukan dengan menerapkan metode *NIST SP 800-86*.

2. METODE PENELITIAN

Metode yang digunakan untuk melakukan penelitian ini yaitu *National Institute of Standard and Technology (NIST) SP 800-86*, dengan tahapan dan skenario kasus kejahatan sebagai berikut:

2.1 Tahapan Penelitian

Tahapan dalam metode *NIST SP 800-86* dapat dilihat pada Gambar 1 berikut ini.



Gambar 1. Tahapan Metode *NIST 800-86*

Berdasarkan gambar di atas, metode ini akan melalui 4 langkah, yakni dimulai dari *collection*, *examination*, *analysis*, hingga *reporting* [16]. Dengan penjelasan lebih lanjut sebagai berikut:

a) *Collection*, tahapan koleksi ini disebut juga tahap persiapan, koleksi yang dimaksud merupakan pengumpulan barang bukti dan peralatan yang akan digunakan untuk mengumpulkan data digital, proses ini mengikuti langkah pengamanan integritas data.

Dalam kasus ini, pengumpulan barang bukti dilakukan dengan mengumpulkan *flashdisk*.

- b) *Examination*, yakni tahap pengambilan data atau penggalian artefak dalam rangka menemukan data pada barang bukti menggunakan aplikasi *autopsy*.
- c) *Analysis*, merupakan tahapan analisa dan melakukan evaluasi terhadap data yang didapatkan pada tahapan sebelumnya.
- d) *Reporting*, adalah tahapan terakhir dalam metode ini yaitu proses pelaporan hasil analisis dari tahapan-tahapan sebelumnya untuk diambil kesimpulan.

2.2 Skenario Kasus Kejahatan

Selanjutnya untuk memperjelas kondisi penelitian, dibuatlah suatu skenario kejahatan yang berkaitan dengan pencurian data perusahaan dan dengan menggunakan media penyimpanan berupa *flashdisk drive*, skenario kejahatan digital yang telah dirancang tersebut secara sederhana dapat dilihat seperti Gambar 2 di bawah ini.



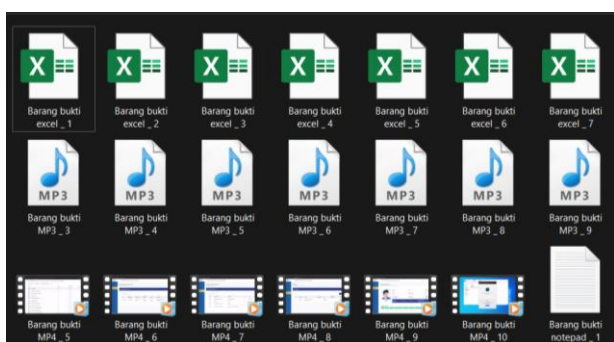
Gambar 2. Skenario Kasus Kejahatan

Tahapan-tahapan yang terjadi pada skenario tersebut secara lebih terperinci adalah sebagai berikut.

- a) Pelaku masuk ke ruang atasan dan menggunakan komputer atasan tanpa seizin nya.
- b) Pelaku kemudian menyalin data penting seperti foto, video, dokumen, dll. yang totalnya berjumlah 70 berkas dan dengan *file type* yaitu *png, docx, xlsx, pdf, txt, mp3, dan mp4*.
- c) Pelaku tertangkap *CCTV* yang memperlihatkan aksinya masuk ke ruang atasan dengan membawa *flashdisk*.
- d) Pelaku pun dicurigai telah mencuri data-data penting perusahaan yang ada di komputer atasan dengan melalui media *flashdisk*.
- e) Setelah di wawancara, pelaku tidak mengatakan yang sebenarnya dan hanya berkata bahwa ia masuk ke ruangan untuk menemui atasan, tetapi karena tidak disana, ia kemudian hanya pergi.
- f) *Investigator* kemudian memutuskan untuk mencari bukti dengan *flashdisk* tersebut, namun *flashdisk* yang diterima dalam kondisi telah diformat sehingga tidak ditemukan berkas apapun.
- g) *Investigator* lalu melakukan akuisisi dan *recovery data* untuk melihat jejak data yg pernah disimpan oleh pelaku dan memastikan kebenaran yang ingin diungkap.

- h) *Investigator* selanjutnya memastikan kembali bahwa data yang dikembalikan masih utuh dengan membandingkan *file hash* dari berkas yang telah di *recovery* dengan berkas asli yang masih disimpan.

Skenario kasus kejahatan tersebut hampir sama dengan skenario kasus kejahatan pada penelitian [6], yaitu pelaku melakukan penggelapan dana perusahaan dan menyimpan bukti transaksi pada *DVD-R*, sehingga *DVD-R* tersebut menjadi barang bukti untuk di investigasi. Berkas yang disiapkan untuk menjadi bukti digital pada *flashdisk* berupa 70 berkas, yaitu berkas dengan ekstensi *DOCX*, *XLSX*, *MP3*, *MP4*, *TXT*, *PDF* dan *PNG*. Dengan nama berkas masing-masing adalah Barang bukti *word*, Barang bukti *excel*, Barang bukti *mp3*, Barang bukti *mp4*, Barang bukti *notepad*, Barang bukti *PDF* dan Barang bukti *png* yang masing-masingnya terdapat 10 berkas, seperti yang dapat dilihat pada Gambar 3 di bawah ini.



Gambar 3. Data yang Disiapkan sebagai Barang Bukti

Filetype yang dipilih terdapat 7 tipe untuk melihat sejauh mana kinerja dari aplikasi *autopsy* dalam mengembalikan data dengan tipe berbeda-beda. Dan dimasukkan tipe yang belum di uji coba pada penelitian sebelumnya [9] seperti *TXT*, *XLSX*, dan *MP3*. Setelah semua data tersebut dimasukkan ke *flashdisk*, data tersebut kemudian dihapus dengan melakukan *format disk* langsung terhadap media *flashdisk* sebagai suatu percobaan dalam menghilangkan bukti atau jejak digital. Sehingga, instrumen penilaian adalah untuk melihat seberapa banyak *file* yang dapat dikembalikan dengan aplikasi tersebut.

3. HASIL DAN PEMBAHASAN

Dalam melakukan proses analisis forensik, diperlukan adanya alat bantu sehingga pekerjaan dapat dilakukan dengan lebih baik, maka digunakanlah beberapa alat yang berupa perangkat keras serta aplikasi yang berupa perangkat lunak, yaitu seperti dijelaskan pada Tabel 1 di bawah ini.

Tabel 1. Alat dan Bahan

Nama	Spesifikasi	Keterangan
Laptop	Acer Swift X, Windows 11	Perangkat Keras
Flashdisk	Sandisk, Kapasitas 16GB	Perangkat Keras
Autopsy	Aplikasi	Perangkat Lunak

3.1 Collection

Barang bukti yang dikumpulkan pada tahapan ini yaitu perangkat *flashdisk* dengan merk *Sandisk Cruzer Blade CZ50 16GB*, dengan bentuk seperti pada Gambar 4 berikut.

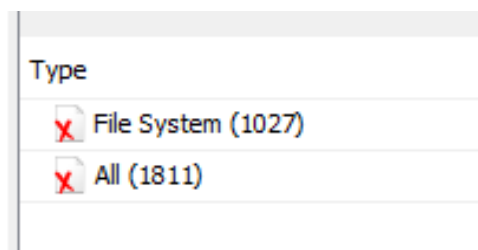


Gambar 4. Flashdisk yang Digunakan

Sebelum menjalankan penelitian ini, kondisi *flashdisk* tersebut sebelumnya telah digunakan untuk kebutuhan sehari-hari yaitu dengan menyimpan berbagai macam berkas dalam berbagai jenis seperti video, musik, dokumen, dan lain sebagainya, sebelum akhirnya *flashdisk* tersebut digunakan untuk menyimpan berkas yang terkait dengan skenario kejahatan dalam penelitian ini

3.2 Examination

Untuk mendapatkan rekam jejak digital berupa berkas apa saja yang pernah disimpan ke dalam *flashdisk*, maka dilakukanlah proses akuisisi data dari perangkat *flashdisk* yang dilakukan dengan aplikasi *autopsy*, kegiatan tersebut memakan waktu yang cukup lama karena perlu menggali setiap data yang sebelumnya telah dihapus. Setelah proses selesai, aplikasi ini berhasil mendapatkan total 1811 data dan 1027 *file system* seperti pada Gambar 5 di bawah ini.



Gambar 5. Proses Akuisisi Data

Jika dikategorikan berdasarkan ekstensinya, ditemukan sebanyak 2118 berkas dengan jenis gambar, 17 berkas dengan jenis video, 353 dengan jenis audio, dan sebanyak 3 berkas dengan jenis arsip.

3.3 Analysis

Pada tahapan analisis dilakukan pencarian data berupa barang bukti, hal tersebut dapat dilakukan melalui pencarian dengan kata kunci atau dicari secara manual. Setelah dilakukan penelusuran, pencarian dengan kata kunci 'barang bukti' sesuai dengan nama berkas sebelum di format, dapat menampilkan beberapa berkas seperti yang dapat dilihat pada Gambar 6 berikut ini.

kemudian dibuat dalam bentuk laporan. Tabel 2 berikut ini merupakan hasil pelaporan dari analisis forensik yang telah dilakukan.

Tabel 2. Hasil Tool *Autopsy*

Data	Ekstensi	Temuan	Komparasi Hash
Barang bukti word	DOCX	10 berkas	Semua berkas identik
Barang bukti excel	XLSX	10 berkas	Semua berkas identik
Barang bukti mp3	MP3	1 berkas	1 berkas identik
Barang bukti mp4	MP4	10 berkas	Semua berkas identik
Barang bukti notepad	TXT	6 berkas	6 berkas identik
Barang bukti pdf	PDF	10 berkas	Semua berkas identik
Barang bukti gambar	PNG	10 berkas	Semua berkas identik

Total temuan dalam penelitian adalah 57 berkas dari 70 berkas dalam skenario. Dan semua temuan telah divalidasi merupakan berkas yang sama dengan berkas aslinya, dengan temuan paling sedikit yaitu berkas dengan tipe *MP3* yang hanya berjumlah 1 berkas.

4. KESIMPULAN

Berdasarkan hasil akhir dari penelitian dengan skenario kasus kejahatan yang telah dibuat yaitu pencurian data melalui media *flashdisk drive*, penggunaan metode *NIST SP 800-86* dan aplikasi *autopsy* dapat diandalkan untuk melakukan analisis forensik khususnya untuk melakukan pemulihan atau pengembalian data, pada kasus ini aplikasi tersebut dapat membantu merestorasi 57 dari total 70 berkas berdasarkan skenario kasus kejahatan, sehingga kinerja aplikasi tersebut dinilai 81,42%, karena berhasil menemukan dan mengembalikan 10 berkas *DOCX*, 10 berkas *XLSX*, 10 berkas *PDF*, 6 berkas *TXT*, 1 berkas *MP3*, 10 berkas *MP4*, dan 10 berkas *PNG*. Dengan fitur yang cukup banyak dan sifat aplikasi yang berbasis *open source*, aplikasi *autopsy* mempunyai keunggulan tersendiri dibandingkan aplikasi lain yang sejenis, terutama pada harga aplikasi yang gratis dan dapat digunakan oleh siapa saja.

DAFTAR PUSTAKA

- [1] Y. Bellini and T. Sutabri, "Sistem Pakar Mendeteksi Tindak Pidana Cyber Crime untuk Penanganan Komputer Forensik Menggunakan Backward Chaining," *JDTI*, vol. 6, no. 1, p. 42, Mar. 2023, doi: 10.32502/digital.v6i1.5619.
- [2] T. Sutabri, *Konsep Sistem Informasi*. Andi, 2012. [Online]. Available: <https://books.google.co.id/books?id=u15eDwAAQB AJ>
- [3] T. Sutabri, *Analisis Sistem Informasi*. Andi, 2012. [Online]. Available: <https://books.google.co.id/books?id=ro5eDwAAQB AJ>
- [4] N. Sulisrudatin, "Analisa Kasus Cybercrime Bidang Perbankan Berupa Modus Pencurian Data Kartu Kredit," *jihd*, vol. 9, no. 1, Jun. 2014, doi: 10.35968/jh.v9i1.296.
- [5] E. F. Thalib and K. L. Maswari, "Perlindungan Hukum terhadap Data Pribadi Perusahaan Akibat Penyalahgunaan Data Digital Oleh Karyawan Perusahaan," *Prosiding Seminar Nasional FH UNMAS Denpasar: Urgensi dan Implikasi RUU Perlindungan Keamanan Kerahasiaan Data Diri Berbasis Digitalisasi*, vol. 1, no. 1, pp. 55–66, Oct. 2020.
- [6] I. Riadi, A. Fadlil, and M. I. Aulia, "Investigasi Bukti Digital Optical Drive Menggunakan Metode National Institute of Standard and Technology (NIST)," *RESTI*, vol. 4, no. 5, pp. 820–828, Oct. 2020, doi: 10.29207/resti.v4i5.2224.
- [7] Y. Andi Putra and T. Sutabri, "Analisis Penyadapan pada Aplikasi Whatsapp Dengan Menggunakan Metode Sinkronisasi Data," *Blantika*, vol. 2, no. 1, pp. 11–20, Feb. 2023, doi: 10.57096/blantika.v2i1.8.
- [8] A. Yudhana, I. Riadi, and I. Anshori, "Analisis Bukti Digital Facebook Messenger Menggunakan Metode Nist," *ITJRD*, vol. 3, no. 1, pp. 13–21, Aug. 2018, doi: 10.25299/itjrd.2018.vol3(1).1658.
- [9] D. T. Yuwono, A. Fadlil, and S. Sunardi, "Performance Comparison of Forensic Software for Carving Files using NIST Method," *Jurnal Teknologi dan Sistem Komputer*, vol. 7, no. 3, pp. 89–92, Jul. 2019, doi: 10.14710/jtsiskom.7.3.2019.89-92.
- [10] S. U. Lubis, "Implementasi Metode MD5 untuk Mendeteksi Orisinalitas File Audio," *KOMIK*, vol. 3, no. 1, Nov. 2019, doi: 10.30865/komik.v3i1.1620.
- [11] N. Nasirudin, S. Sunardi, and I. Riadi, "Analisis Forensik Smartphone Android Menggunakan Metode NIST dan Tool MOBILedit Forensic Express," *JiUP*, vol. 5, no. 1, p. 89, Mar. 2020, doi: 10.32493/informatika.v5i1.4578.
- [12] I. Riadi, R. Umar, and I. M. Nasrulloh, "Analisis Forensik Digital pada Frozen Solid State Drive Dengan Metode National Institute Of Justice (NIJ)," *ELINVO*, vol. 3, no. 1, pp. 70–82, Jul. 2018, doi: 10.21831/elinvo.v3i1.19308.
- [13] I. Riadi, S. Sunardi, and S. Sahrudin, "Analisis Forensik Recovery pada Smartphone Android Menggunakan Metode National Institute Of Justice (NIJ)," *JURTI*, vol. 3, no. 1, p. 87, Jun. 2019, doi: 10.30872/jurti.v3i1.2292.
- [14] R. Umar, A. Yudhana, and M. N. Fadillah, "Perbandingan Tools Forensik Pada Aplikasi Dompot Digital," *JIKO*, vol. 6, no. 2, p. 242, Sep. 2022, doi: 10.26798/jiko.v6i2.621.

- [15] W. Pranoto, I. Riadi, and Y. Prayudi, "Perbandingan Tools Forensics pada Fitur TRIM SSD NVMe Menggunakan Metode Live Forensics," *ITJRD*, vol. 4, no. 2, Feb. 2020, doi: 10.25299/itjrd.2020.vol4(2).4615.
- [16] I. Riadi, Sunardi, and Sahiruddin, "Perbandingan Tool Forensik Data Recovery Berbasis Android Menggunakan Metode NIST," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 7, no. 1, pp. 197–204, Jan. 2020, doi: 10.25126/jtiik.202071921.



PERANCANGAN ALAT TEKANAN DARAH OTOMATIS SECARA BERKALA MENGGUNAKAN SENSOR MPX5050DP UNTUK PASIEN HIPERTENSI

Afdhal Kurniawan¹, Karli Eka Setiawan²

^{1,2}Computer Science Department, School of Computer Science, Bina Nusantara University
Jakarta Barat, DKI Jakarta, Indonesia 11530

afdhal.kurniawan@binus.ac.id, karli.setiawan@binus.ac.id

Abstract

Regular blood pressure monitoring is crucial in caring for outpatient patients undergoing treatment. Hypertension, commonly known as high blood pressure, is a prevalent medical condition that can increase the risk of heart disease, stroke, and other vascular disorders. Patients with hypertension often require regular blood pressure measurements to monitor their health status and evaluate the effectiveness of ongoing treatment. To mitigate the risks associated with hypertension, routine health check-ups, including periodic blood pressure measurements at home, can be taken. This research aims to make a regular automatic blood pressure meter to help hypertensive patients. The design of an automatic periodic blood pressure measurement device using the MPX5050DP sensor for hypertensive patients is expected to facilitate regular monitoring and assist doctors in making accurate diagnoses. Research results indicate a larger error percentage for diastolic values than systolic values. Specifically, the error percentage is 3.45% for systolic values and 7.17% for diastolic values.

Keywords: Blood Pressure Measurement, Diastolic, Hypertension, Sensor, Systolic

Abstrak

Pemantauan tekanan darah secara teratur memiliki peran yang sangat penting dalam merawat pasien rawat jalan yang menjalani pengobatan. Hipertensi, yang dikenal sebagai tekanan darah tinggi, merupakan kondisi medis umum yang dapat meningkatkan risiko penyakit jantung, stroke, dan gangguan vaskular lainnya. Pasien dengan hipertensi sering perlu menjalani pengukuran tekanan darah berkala untuk memantau status kesehatan dan menilai efektivitas pengobatan yang sedang berlangsung. Untuk mengurangi risiko hipertensi, salah satu tindakan yang dapat diambil adalah pemeriksaan kesehatan rutin, termasuk pengukuran tekanan darah berkala di rumah. Tujuan dari penelitian ini adalah untuk membuat alat pengukur tekanan darah otomatis secara berkala untuk membantu pasien hipertensi. Desain alat pengukuran tekanan darah berkala otomatis dengan menggunakan sensor MPX5050DP untuk pasien hipertensi diharapkan dapat memfasilitasi pemantauan berkala dan membantu dokter dalam membuat diagnosa yang akurat. Hasil penelitian menunjukkan bahwa *persentase error* diastolik lebih besar dibandingkan dengan *persentase error* sistolik. Dengan rincian *persentase error* sebesar 3,45% untuk nilai sistolik dan 7,17% untuk nilai diastolik.

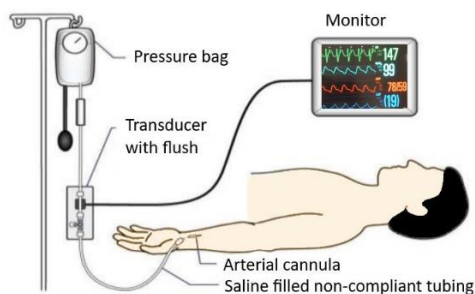
Kata kunci: Diastolik, Hipertensi, Pengukuran Tekanan Darah, Sensor, Sistolik

1. PENDAHULUAN

Pengukuran tekanan darah secara teratur memainkan peran yang sangat vital dalam merawat pasien yang menjalani perawatan sebagai pasien rawat jalan. Hipertensi, yang dikenal juga sebagai tekanan darah tinggi, merupakan kondisi medis yang umum dan dapat meningkatkan risiko terhadap penyakit jantung, stroke, dan gangguan pembuluh darah lainnya [1]. Pasien yang mengalami hipertensi seringkali perlu melakukan pengukuran tekanan darah secara berkala untuk memonitor kondisi kesehatan mereka serta menilai efektivitas dari pengobatan yang sedang dilakukan[2]. Menurut hasil Penelitian Kesehatan Dasar

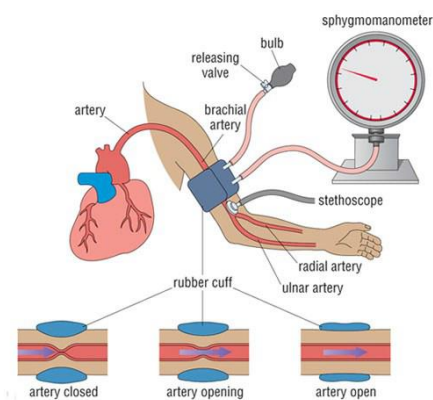
(Risesdas) yang dilakukan oleh Kementerian Kesehatan, angka kejadian hipertensi di Indonesia mengalami peningkatan dari 25,8 persen pada tahun 2013 menjadi 34,11 persen pada tahun 2018[3]. Data juga menunjukkan bahwa hipertensi tidak hanya terbatas pada kelompok usia lanjut, tetapi juga mengenai individu berusia di bawah 45 tahun, bahkan termasuk remaja dan dewasa muda. Untuk mengurangi risiko terkena hipertensi, salah satu tindakan yang dapat diambil adalah menjalani pemeriksaan kesehatan secara rutin [4], termasuk melakukan pengukuran tekanan darah secara berkala di lingkungan rumah[5], [6]. Pengukuran tekanan darah dapat dilakukan dengan dua

pendekatan yang berbeda: metode invasif dan metode non-invasif. Teknik invasif melibatkan pengukuran tekanan darah secara langsung melalui metode yang menembus kulit [7]. Umumnya digunakan di lingkungan medis khusus seperti unit perawatan intensif atau prosedur bedah yang memerlukan pemantauan tekanan darah yang teliti dan konsisten, pendekatan ini melibatkan pemasangan kateter ke dalam pembuluh darah. Kateter ini memungkinkan pengukuran tekanan darah secara *real-time* dan akurat, menghasilkan informasi berharga untuk manajemen pasien. Pada Gambar 1, menjelaskan bagaimana pengukuran tekanan darah secara invasif.

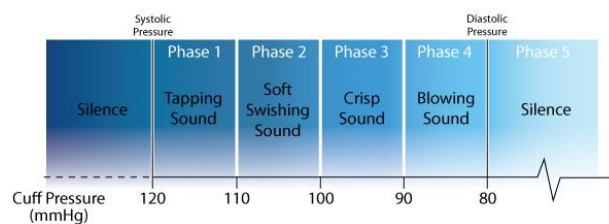


Gambar 1. Invasive measurement method [8]

Di sisi lain, metode *non-invasif* mengacu pada pengukuran tekanan darah tanpa penetrasi kulit atau cedera fisik [9]. Alat yang paling umum digunakan dalam pendekatan ini adalah sphygmomanometer, yang dipadukan dengan stetoskop [10]. Prosedur diagnostik yang dikenal luas dalam kategori *non-invasif* melibatkan penggunaan manset penekan (*Cuff*), yang ditempatkan di sekitar lengan bagian atas. Dengan cara ini, tekanan manset diisi hingga melebihi tekanan sistolik darah, menyebabkan pembuluh arteri terjepit sehingga aliran darah berhenti seperti yang terlihat pada gambar 2. Dengan perlahan mengurangi tekanan manset, saat aliran darah kembali dan suara denyut nadi terdengar melalui stetoskop (fase Korotkoff) seperti pada gambar 3, bacaan tekanan pada manometer ketika suara denyut pertama terdeteksi menunjukkan tekanan darah sistolik [11]. Terus mengurangi tekanan manset hingga suara denyut nadi hilang sepenuhnya memungkinkan bacaan pada saat itu mengindikasikan tekanan darah diastolik.



Gambar 2. Non-invasive measurement method [8]



Gambar 3. Korotkoff Phase [12]

Perancangan alat tekanan darah otomatis secara berkala menggunakan sensor MPX5050DP untuk pasien hipertensi diharapkan nantinya dapat mempermudah dalam melakukan pemantauan secara berkala [13]. Serta mempermudah dokter dalam melakukan diagnosa sehingga pasien mendapatkan penanganan yang sesuai. Hal tersebut juga dapat mengurangi angka kematian. Alat monitoring tekanan darah ini nantinya dapat mengukur sistol dan diastol secara otomatis dan data yang didapat tersebut dikirim ke dalam database secara berkala sehingga data tersebut dapat diamati di kemudian hari jika saat dibutuhkan.

2. METODE PENELITIAN

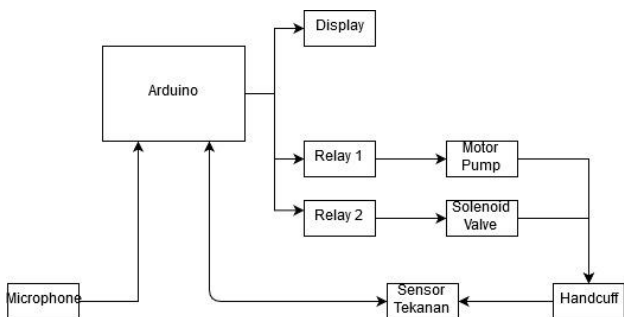
2.1 Sensor dan Ringkasan Spesifikasi

Dalam sistem ini, digunakan dua jenis sensor, yaitu sensor tekanan MPX5050DP untuk mengukur tekanan di manset dan sensor mikrofon MAX9814 sebagai pengganti stetoskop. Dalam pemilihan sensor tekanan, tersedia berbagai pilihan tipe, dan MPX5050DP dipilih karena memiliki kemampuan mengukur tekanan dari 0 hingga 50 Kpa, setara dengan 375 mmHg, dengan output analog dalam kisaran 0,2V hingga 4,7V [14]. Hal ini sesuai dengan kebutuhan pengukuran tekanan darah yang berada dalam rentang 60 mmHg hingga 120 mmHg. Sedangkan sensor mikrofon MAX9814 dipilih karena menghasilkan output analog antara 2,7V hingga 5,5V, dan berdasarkan percobaan, sensor ini mampu menangkap suara korotkoff yang penting untuk menghitung nilai tekanan sistolik dan diastolik.

2.2 Blok Diagram dan Perencanaan Alat

Alat yang dirancang harus dapat dengan mudah dioperasikan dan menggunakan komponen yang sering ditemukan dipasaran sehingga alat ini dapat digunakan oleh orang banyak dan memiliki manfaat. Metode yang digunakan adalah dengan melakukan pengukuran tekanan darah dengan menggunakan sebuah tensimeter digital berbasis arduino dengan menggunakan metode menangkap korotkoff sound untuk memperoleh sistol dan diastol. Dalam Gambar 4, yaitu Diagram Blok Perancangan Tensimeter Digital, dapat diamati langkah-langkah dalam proses pengukuran tekanan darah. Pertama, *handcuff* (manset) akan dipompa menggunakan motor pump hingga mencapai tekanan yang diinginkan. Ketika tekanan ini tercapai, aliran darah dalam arteri akan terhenti karena tekanan dari handcuff. Setelah itu, *valve* (katup) akan diaktifkan untuk perlahan-lahan mengeluarkan udara dari

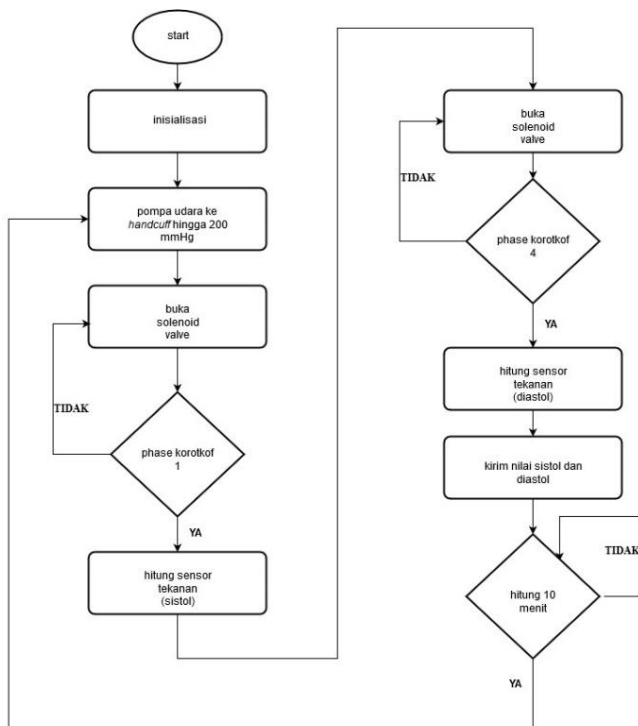
dalam *handcuff*, memungkinkan aliran darah kembali normal. Ketika aliran darah mulai mengalir kembali, sensor mikrofon akan mendeteksi suara korotkoff. Bunyi pertama yang terdengar menandai nilai sistolik, sedangkan bunyi terakhir menunjukkan nilai diastolik. Setelah nilai-nilai sistolik dan diastolik diperoleh, informasi ini akan ditampilkan pada layar LCD untuk visualisasi hasil pengukuran.



Gambar 4. Diagram automatic blood pressure test.

2.3 Alur Kerja Sistem

Proses ini diilustrasikan melalui sebuah *flowchart* yang terdapat pada Gambar 5. *Flowchart* tersebut merinci langkah-langkah yang terlibat dalam proses pengukuran tekanan darah secara digital.



Gambar 5. Program flowcharts.

Sistem dimulai dengan memasang *handcuff* pada pasien dengan posisi mikrofon di bagian dalam siku, tempat bunyi korotkoff terdengar. Setelah itu, alat terhubung ke laptop dan diaktifkan. Begitu diaktifkan, alat akan beroperasi secara otomatis. Tahap awal melibatkan pemompaan udara ke dalam *handcuff* pengukuran tekanan darah menggunakan

motor pump, hingga mencapai tekanan berkisar antara 180 mmHg hingga 200 mmHg, atau melebihi tekanan darah normal yaitu 120 mmHg. Tekanan ini menghentikan aliran darah dalam arteri karena tekanan dari *handcuff*. Selanjutnya, tekanan di dalam *handcuff* secara perlahan dikurangi, memungkinkan aliran darah kembali normal. Pengurangan tekanan ini menghasilkan bunyi korotkoff yang akan ditangkap oleh mikrofon. Bunyi pertama menandai nilai sistolik, sedangkan bunyi terakhir menunjukkan nilai diastolik. Setelah nilai-nilai ini terdeteksi, mereka ditampilkan pada layar LCD sebagai hasil pengukuran tekanan darah. Data ini dapat diamati pada layar LCD. Setelah proses pengukuran selesai, alat akan menunggu selama 10 menit sebelum dapat diaktifkan kembali untuk pengukuran berikutnya.

3. HASIL DAN PEMBAHASAN

3.1 Program Sensor Tekanan

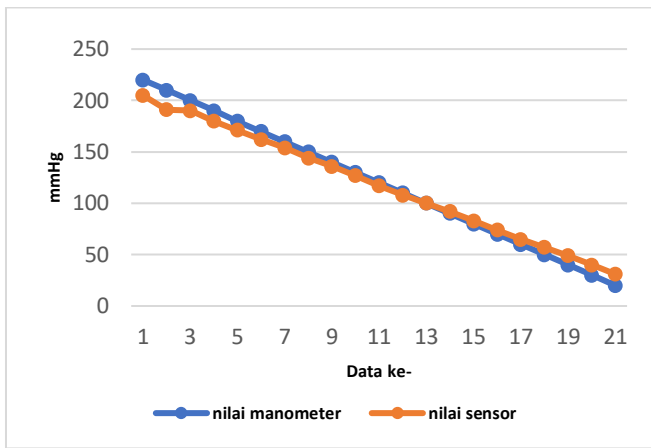
Sensor tekanan MPX5050DP memiliki keluaran berupa sinyal analog, yang perlu diubah menjadi digital melalui *Analog to Digital Converter* (ADC). Karena sinyal keluaran masih dalam bentuk tegangan, maka perlu dikonversi menjadi tekanan dalam satuan KPa. Konversi ini dilakukan berdasarkan rumus yang disesuaikan dengan spesifikasi perangkat pada datasheet. Dalam script, konversi dapat diwujudkan sebagai berikut:

$$kpa_pressure = (50000/1024) * (pressure_sensor);$$

Setelah itu, dilakukan konversi dari satuan tekanan KPa menjadi satuan tekanan darah yaitu mmHg. Karena 1 KPa setara dengan 7,50062 mmHg, Arduino dapat langsung memproses input tegangan dari sensor tekanan menjadi nilai dalam satuan mmHg. Dengan demikian, proses konversi ini memungkinkan sensor tekanan yang awalnya mengirimkan data dalam bentuk tegangan dapat diubah menjadi nilai tekanan darah dalam satuan mmHg.

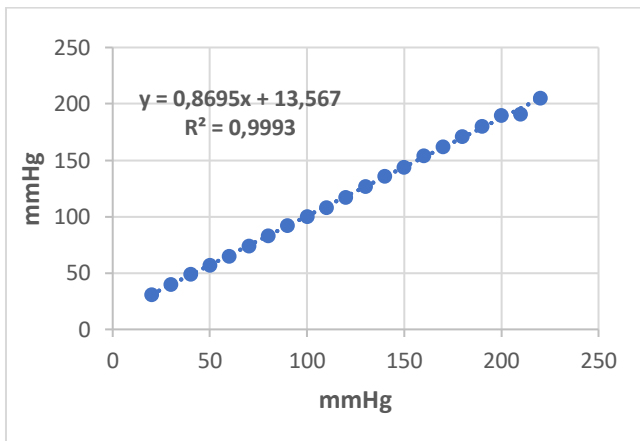
3.2 Uji Kalibrasi Tekanan

Sebelum mengambil data tekanan darah, sensor MPX5050DP yang digunakan harus dijalankan melalui proses kalibrasi. Tujuan dari kalibrasi ini adalah untuk memastikan bahwa nilai tekanan yang dihasilkan oleh sensor mendekati nilai yang diukur oleh manometer yang umumnya digunakan. Penggunaan manometer dalam proses kalibrasi dipilih karena alat tensimeter digital yang digunakan untuk membandingkan pengukuran tidak dapat diadaptasi untuk tujuan kalibrasi sensor tekanan, karena telah diatur oleh produsen untuk tujuan spesifik. Dari hasil proses kalibrasi ini, diperoleh perbandingan antara nilai tekanan yang dihasilkan oleh sensor MPX5050DP dengan nilai tekanan yang dihasilkan oleh manometer. Perbandingan ini dapat dilihat pada Gambar 6, yang menggambarkan perbedaan antara kedua nilai tekanan tersebut.



Gambar 6. Sensor comparison chart with manometer.

Karena adanya perbedaan antara hasil pengukuran sensor MPX5050DP dan nilai tekanan yang diukur oleh manometer, maka dari itu diperlukan kalibrasi dengan menggunakan metode *Least Square* [15] seperti yang ditunjukkan pada Gambar 7, untuk memperoleh hasil pengukuran tekanan yang lebih mendekati nilai pengukuran manometer. Dengan metode ini, diperoleh sebuah persamaan linear $y=ax+b$.

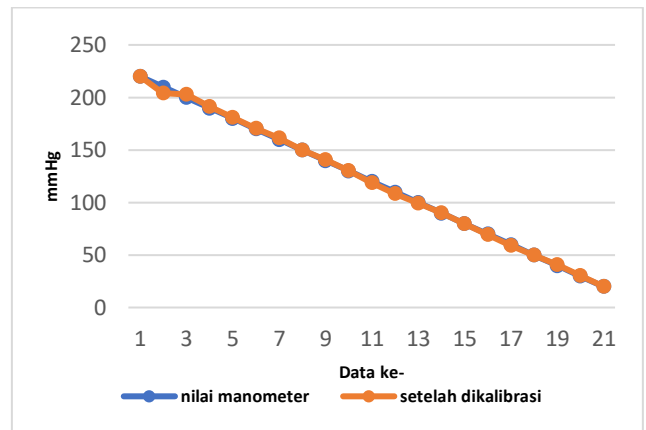


Gambar 7. Graph of Least Square Equation Results.

Dalam proses least square, persamaan tersebut diuraikan menjadi nilai $a = 0,8695$ dan $b = 13,567$. Selanjutnya, persamaan *least square* yang telah diperoleh akan diinverskan menjadi:

$$X = (Y - 13567) / 0,8695$$

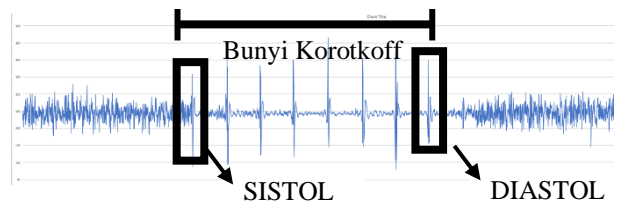
Untuk mendapatkan nilai tekanan dari sensor yang telah dikalibrasi agar sesuai dengan Gambar 8, digunakan persamaan *invers* yang telah dihitung sebelumnya. Nilai tekanan yang dihasilkan sensor dihitung dengan menggunakan persamaan invers tersebut sehingga nilai yang dihasilkan lebih akurat.



Gambar 8. Sensor Comparison Chart with manometer After Calibration.

3.3 Uji Sensor Mikrofon

Untuk mendapatkan suara korotkoff dalam pengukuran tekanan darah menggunakan alat tekanan darah otomatis, digunakan sensor mikrofon sebagai pengganti stetoskop. Sensor mikrofon yang digunakan adalah MAX9814. Sebelum sensor mikrofon digunakan dalam pengukuran sebenarnya, dilakukan pengujian untuk memastikan bahwa sensor tersebut dapat menghasilkan hasil yang diinginkan. Pengujian ini bertujuan untuk memeriksa kemampuan sensor mikrofon dalam menangkap suara korotkoff dengan baik. Hasil dari pengujian ini ditampilkan dalam bentuk grafik yang menunjukkan suara yang berhasil diambil oleh sensor mikrofon. Gambar grafik tersebut dapat dilihat pada Gambar 9.



Gambar 9. Sensor Graph Results.

3.4 Hasil Pengukuran terhadap Omron hem-7130

Pengambilan data dari responden terbatas pada populasi dewasa karena alat yang telah dibuat dirancang khusus untuk ukuran dan metode pengambilan tekanan darah pada orang dewasa. Pengambilan data dari anak-anak memerlukan ukuran *handcuff* yang berbeda dan pendekatan yang berbeda. Namun, dalam pengembangan selanjutnya, diharapkan alat ini dapat ditingkatkan untuk dapat mengakomodasi pengukuran pada anak-anak dengan penyesuaian yang sesuai. Pengambilan data dari populasi dewasa dilakukan untuk menganalisis variasi hasil yang dihasilkan oleh alat yang telah dibuat. Hasil ini nantinya akan dibandingkan dengan alat pengukur tekanan darah digital lain yang sudah banyak beredar di pasaran. Alat Omron hem-7130 digunakan sebagai perbandingan untuk hasil yang diperoleh. Selain itu, diperhatikan juga stabilitas

dan keselarasan hasil yang dicatat antara tekanan sistolik dan diastolik. Tampilan pemasangan alat pada responden dapat dilihat pada Gambar 10.



Gambar 10. Tool Installation.

Tata cara pelaksanaan pemeriksaan terhadap responden melibatkan tiga pengukuran menggunakan alat yang telah dibuat dan tiga pengukuran menggunakan Omron hem-7130 pada setiap responden. Setelah itu, dilakukan perhitungan rata-rata dari hasil pengukuran tersebut untuk mendapatkan nilai rata-rata tekanan sistolik dan diastolik yang lebih akurat. Melalui pendekatan ini, diperoleh *persentase error* yang mencerminkan perbedaan antara hasil pengukuran dari alat yang dibuat dan Omron hem-7130. Rincian *persentase error* ini dapat ditemukan dalam Tabel 1.

Tabel 1. Results of Comparison of Measuring Instruments with Omron hem-7130.

Pasien	Rata-rata pengukuran Omron hem-7130		Rata-rata pengukuran alat		Persentase error	
	Sistol	Diastol	Sistol	Diastol	Sistol	Diastol
A	111	67	114	73	2,70%	8,96%
B	109	74	113	80	3,67%	8,11%
C	115	84	119	89	3,48%	5,95%
D	105	74	109	79	3,81%	6,76%
E	112	82	116	87	3,57%	6,10%
Rata-rata					3,45%	7,17%

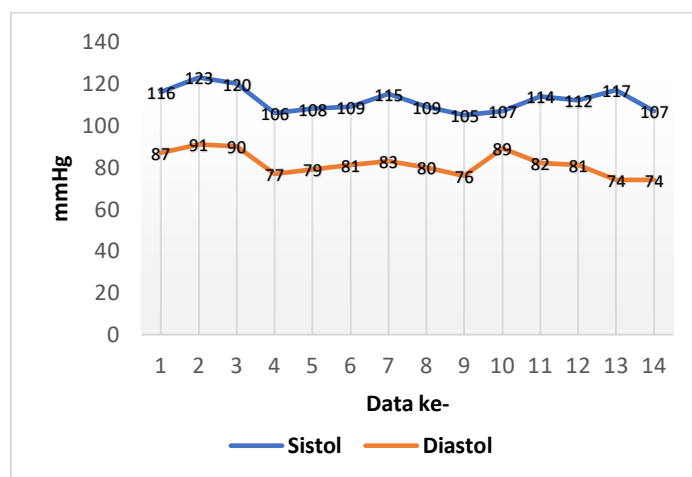
Dari hasil pengujian terhadap lima responden, diperoleh *persentase error* keseluruhan untuk tekanan sistolik sebesar 3,45% dan tekanan diastolik sebesar 7,17%. *Persentase error* tersebut menunjukkan bahwa perbedaan antara hasil pengukuran dari alat yang dibuat dan Omron hem-7130 tidak melebihi 8%, sehingga alat yang dibuat dapat dianggap cukup dapat diandalkan dalam penggunaannya. Meskipun demikian, perlu diingat bahwa penggunaan Omron hem-7130 sebagai alat pembanding mungkin tidak selalu menghasilkan hasil yang akurat, dan alat yang dibuat mungkin memiliki potensi untuk lebih akurat. Oleh karena itu, sebagai saran pengembangan ke depan, diharapkan penggunaan alat pembanding yang lebih akurat dapat

dipertimbangkan untuk menguji dan memvalidasi hasil pengukuran alat yang dibuat. Berikutnya merupakan hasil pengukuran secara berkala yang diilustrasikan sedang dirawat dalam sebuah ruangan ICU yang mana diperlukan pengukuran tekanan darah secara berkala untuk dapat membantu seorang dokter dalam mengamati perkembangan pasien. Hasil pengukuran secara berkala tersebut dapat dilihat pada Tabel 2.

Tabel 2. Results of Periodic Measurements

Date	Time	Sistol	Diastol
16/08/2023	2.40.44 PM	116	87
16/08/2023	2.51.24 PM	123	91
16/08/2023	3.02.04 PM	120	90
16/08/2023	3.12.44 PM	106	77
16/08/2023	3.23.24 PM	108	79
16/08/2023	3.34.04 PM	109	81
16/08/2023	3.44.44 PM	115	83
16/08/2023	3.55.24 PM	109	80
16/08/2023	4.06.04 PM	105	76
16/08/2023	4.16.45 PM	107	89
16/08/2023	4.27.25 PM	114	82
16/08/2023	4.38.05 PM	112	81
16/08/2023	4.48.45 PM	117	74
16/08/2023	4.59.25 PM	107	74

Pada Tabel 2, tersebut terlihat pengukuran pada tabel tersebut pengambilan data sistol dan diastol secara berkala setiap 10 menit sehingga dapat dibuat grafik seperti pada Gambar 11.



Gambar 11. Graph of Periodic Measurement Results.

4. KESIMPULAN

Hasil analisis dan pengamatan dalam penelitian ini secara keseluruhan menunjukkan adanya perbedaan data antara

alat yang dikembangkan dan alat pembanding, yaitu Omron hem-7130. Penelitian ini mengungkapkan bahwa nilai diastol yang dihasilkan oleh alat yang dikembangkan tidak seakurat nilai sistol. Persentase kesalahan sebesar 3,45% untuk nilai sistol dan 7,17% untuk nilai diastol mencerminkan perbedaan tersebut. Agar hasil pengukuran lebih akurat, penting bagi pengguna untuk tetap rileks selama proses pengukuran dan memastikan manset menempel dengan baik pada kulit pasien. Kesimpulan ini memberikan panduan penting untuk pengembangan lebih lanjut dan penggunaan perangkat pengukur tekanan darah digital yang telah dirancang.

DAFTAR PUSTAKA

- [1] A. Abdullah and S. Othman, "The influence of self-owned home blood pressure monitoring (HBPM) on primary care patients with hypertension: A qualitative study," *BMC Fam Pract*, vol. 12, no. 1, p. 143, Dec. 2011, doi: 10.1186/1471-2296-12-143.
- [2] M. Xiao *et al.*, "Home Blood Pressure Monitoring by a Mobile-Based Model in Chongqing, China: A Feasibility Study," *Int J Environ Res Public Health*, vol. 16, no. 18, p. 3325, Sep. 2019, doi: 10.3390/ijerph16183325.
- [3] Kementerian Kesehatan RI Badan Penelitian dan Pengembangan Kesehatan, "Hasil Utama RISKESDAS 2018," 2018. Accessed: Aug. 20, 2023. [Online]. Available: https://kesmas.kemkes.go.id/assets/upload/dir_519d41d8cd98f00/files/Hasil-riskesdas-2018_1274.pdf
- [4] D. Shimbo *et al.*, "Self-Measured Blood Pressure Monitoring at Home: A Joint Policy Statement From the American Heart Association and American Medical Association," *Circulation*, vol. 142, no. 4, Jul. 2020, doi: 10.1161/CIR.0000000000000803.
- [5] AMd. K. Dyah Ayu Wulandari, "Ketahui Manfaat Mengukur Tekanan Darah Rutin Sendiri di Rumah, Apa Saja?," Feb. 16, 2023. https://yankes.kemkes.go.id/view_artikel/2164/ketahui-manfaat-mengukur-tekanan-darah-rutin-sendiri-di-rumah-apa-saja (accessed Aug. 20, 2023).
- [6] P. Natale *et al.*, "Perspectives and Experiences of Self-monitoring of Blood Pressure Among Patients With Hypertension: A Systematic Review of Qualitative Studies," *Am J Hypertens*, vol. 36, no. 7, pp. 372–384, Jun. 2023, doi: 10.1093/ajh/hpad021.
- [7] S. Romagnoli *et al.*, "Accuracy of invasive arterial pressure monitoring in cardiovascular patients: an observational study," *Crit Care*, vol. 18, no. 6, p. 644, Dec. 2014, doi: 10.1186/s13054-014-0644-4.
- [8] Ü. ŞENTÜRK, K. POLAT, and İ. YÜCEDAĞ, "Towards wearable blood pressure measurement systems from biosignals: a review," *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, vol. 27, no. 5, pp. 3259–3281, Sep. 2019, doi: 10.3906/elk-1812-121.
- [9] A. S. Meidert and B. Saugel, "Techniques for Non-Invasive Monitoring of Arterial Blood Pressure," *Front Med (Lausanne)*, vol. 4, Jan. 2018, doi: 10.3389/fmed.2017.00231.
- [10] S. N. A. Ismail, N. A. Nayan, R. Jaafar, and Z. May, "Recent Advances in Non-Invasive Blood Pressure Monitoring and Prediction Using a Machine Learning Approach," *Sensors*, vol. 22, no. 16, p. 6195, Aug. 2022, doi: 10.3390/s22166195.
- [11] W. Lin *et al.*, "Korotkoff sounds dynamically reflect changes in cardiac function based on deep learning methods," *Front Cardiovasc Med*, vol. 9, Aug. 2022, doi: 10.3389/fcvm.2022.940615.
- [12] Cindy Maley, "Intro to Blood Pressure," Jul. 16, 2013. <https://www.adctoday.com/blog/intro-blood-pressure> (accessed Aug. 20, 2023).
- [13] G. S. Stergiou *et al.*, "A Universal Standard for the Validation of Blood Pressure Measuring Devices," *Hypertension*, vol. 71, no. 3, pp. 368–374, Mar. 2018, doi: 10.1161/HYPERTENSIONAHA.117.10237.
- [14] NXP BV, "MPX5050 Integrated silicon pressure sensor, on-chip signal conditioned, temperature compensated and calibrated," 2023. Accessed: Aug. 20, 2023. [Online]. Available: <https://cache.nxp.com/docs/en/data-sheet/MPX5050.pdf>
- [15] F. Raposo, "Evaluation of analytical calibration based on least-squares linear regression for instrumental techniques: A tutorial review," *TrAC Trends in Analytical Chemistry*, vol. 77, pp. 167–185, Mar. 2016, doi: 10.1016/j.trac.2015.12.006.



PENGELOMPOKAN RUMAH SAKIT DI JAKARTA MENGGUNAKAN MODEL DBSCAN, GAUSSIAN MIXTURE, DAN HIERARCHICAL CLUSTERING

Karli Eka Setiawan¹, Afdhal Kurniawan²

^{1,2}Computer Science Department, School of Computer Science, Bina Nusantara University
Jakarta Barat, DKI Jakarta, Indonesia 11480

karli.setiawan@binus.ac.id, afdhal.kurniawan@binus.ac.id

Abstract

It should be recognized that after the COVID-19 pandemic, the distribution of hospital facilities should be a primary concern in meeting society's fundamental rights to local and national governments. Therefore, it is necessary to analyze data that describes the current conditions regarding the distribution of hospital facilities, especially in Jakarta. Data analysis using a machine learning technique can provide several benefits, such as understanding the distribution of health workers, providing insight into human resource planning, monitoring hospital performance, and even becoming a reference for future targets by predicting the need for health workers. The research proposed three models of unsupervised learning, such as the density-based spatial clustering of applications with noise algorithm (DBSCAN), the Gaussian mixture model, and the agglomerative hierarchical model, to cluster a database of hospitals in Jakarta, Indonesia, that contains information on the number of medical workers in the hospital and its bed facilities. This data set was obtained from a web scraping process on the Ministry of Health of the Republic of Indonesia website in 2022. The comparison of the third unsupervised model showed that the Gaussian Mixture model produced the smallest Davies-Bouldin value with a value of 0.645713. This study is an extension of our previous investigation, in which the study is, as far as we know, the first to discuss the classification of the data on the list of hospitals in Jakarta based on information on the number of medical personnel and the number of bed facilities that make up the contribution of this research.

Keywords: Agglomerative Hierarchical, DBSCAN, Gaussian Mixture, Hospital Clustering, unsupervised learning

Abstrak

Perlu disadari bahwa setelah mengalami pandemi COVID-19, distribusi persebaran fasilitas rumah sakit harus menjadi perhatian utama dalam memenuhi hak-hak dasar dari masyarakat oleh pemerintah tingkat daerah dan tingkat nasional. Oleh karena itu diperlukan sebuah analisis data yang menggambarkan kondisi terkini mengenai persebaran fasilitas rumah sakit khususnya di Jakarta. Analisis Data dengan pendekatan teknik machine learning mampu memberikan beberapa manfaat seperti mendapatkan manfaat dalam memberikan pemahaman mengenai distribusi tenaga kesehatan, memberikan wawasan dalam rangka perencanaan sumber daya manusia, melakukan pemantauan kinerja rumah sakit, dan bahkan menjadi acuan target di masa depan dengan prediksi kebutuhan tenaga kesehatan. Penelitian ini mengajukan tiga buah model *unsupervised learning*, seperti model *Density-based spatial clustering of applications with noise algorithm* (DBSCAN), model *Gaussian Mixture*, dan model *Agglomerative Hierarchical* untuk melakukan pengelompokan sekumpulan data rumah sakit se-Jakarta, Indonesia yang berisi informasi data jumlah berbagai tenaga kerja medis di rumah sakit tersebut dan fasilitas tempat tidurnya. *Dataset* ini diperoleh dari proses *web scraping* pada *website* Kementerian Kesehatan Republik Indonesia pada tahun 2022. Dari perbandingan ketiga model *unsupervised*, diperoleh bahwa model *Gaussian Mixture* menghasilkan nilai *Davies Bouldin* terkecil dengan nilai 0,645713. Penelitian ini merupakan lanjutan dari penelitian kami sebelumnya, dimana penelitian tersebut menurut sepengetahuan kami merupakan penelitian pertama yang membahas pengelompokan data daftar rumah sakit se-Jakarta berdasarkan informasi data jumlah tenaga medis dan jumlah fasilitas tempat tidur yang menjadikan kontribusi dari penelitian ini.

Kata kunci: Agglomerative Hierarchical, DBSCAN, Gaussian Mixture, pengelompokan rumah sakit, *unsupervised learning*

1. PENDAHULUAN

Pengembangan di sektor kesehatan kini menjadi bagian dari perhatian pemerintah pusat dan daerah dengan tindakan

nyata berupa menyediakan fasilitas kesehatan yang memadai dan meningkatkan angka harapan hidup Masyarakat Indonesia karena Kesehatan Masyarakat

merupakan salah satu dari hak-hak dasar dari masyarakat yang harus direalisasikan [1]. Pengumpulan data dan analisis performa dari fasilitas Kesehatan dalam suatu negara merupakan hal yang penting untuk kebijakan pemerintah di masa depan, khususnya kebijakan terkait Kesehatan Masyarakat. Indonesia negara yang memiliki jumlah penduduk yang sangat banyak yaitu sekitar lebih dari 200 juta penduduk yang tentunya memiliki permasalahan dalam melakukan distribusi persebaran fasilitas Kesehatan dan tenaga ahli Kesehatan seperti dokter, perawat, bidan dan lainnya yang dimana saat ini mayoritas tersebar di daerah kota-kota besar [2]. Hal tersebut dapat terlihat ketika Indonesia menghadapi pandemi COVID-19 dimana tenaga medis yang tersedia tidak mencukupi dan meningkatnya permintaan penggunaan fasilitas Kesehatan oleh para pasien [3]. Saat ini akses untuk memperoleh data fasilitas rumah sakit dengan informasi detail di Indonesia sangat terbatas. Oleh karena itu menjadi pekerjaan yang sangat berat bagi Masyarakat untuk menentukan level dari rumah sakit yang ada di Indonesia. Terlebih lagi data terkait berapa jumlah dokter dan dokter spesialis, perawat, bidan, jumlah tempat tidur, fasilitas *intensive care unit* (ICU), dan layanan unit gawat darurat di sebuah rumah sakit tertentu tersebar di *website* masing-masing setiap rumah sakit. Oleh karena itu penelitian ini melanjutkan penelitian kami sebelumnya yang setelah memiliki *dataset* rumah sakit se-Jakarta pada tahun 2022 dengan menerapkan model *unsupervised machine learning* untuk melakukan pengelompokan data rumah sakit se-Jakarta. Diharapkan dengan adanya riset ini maka dapat membantu pemerintah pusat maupun pemerintah daerah dalam membuat keputusan terkait kebijakan dibidang kesehatan dan distribusi tenaga kesehatan.

Kontribusi dari penelitian ini adalah komparasi tiga buah model *unsupervised learning* yaitu DBSCAN, GMM, dan *Hierarchical* untuk melakukan pengelompokan data rumah sakit se-Jakarta. Sepengetahuan kami sebelumnya belum ada pembahasan riset *paper* yang membahas pengelompokan rumah sakit di Indonesia berdasarkan tenaga kerja medis dan jumlah fasilitas tempat tidur kecuali penelitian kami sebelumnya [4].

Dalam bidang medis, pendekatan *unsupervised learning* untuk keperluan pengelompokan data terbukti sangat mendatangkan banyak manfaat dengan mengungkapkan fakta-fakta yang tidak dapat dilihat oleh mata manusia jika tidak dari kumpulan data yang sangat besar jumlahnya [5]. Pada penelitian sebelumnya pendekatan dengan *unsupervised learning* telah digunakan untuk keperluan mendiagnosis berbagai penyakit seperti kanker payudara, Parkinson, sakit kepala, penyakit mental, penyakit jantung, diabetes, dan banyak lainnya. Seorang peneliti mempelajari pendeteksian penyakit Alzheimer dengan pendekatan pengelompokan *unsupervised* dengan menggunakan model *K-Means*. *K-Means-Mode*, *clustering multilayer* dan *hierarchical agglomerative* [5]. Terdapat penelitian lainnya yang melakukan kalsterisasi dibidang medis dengan

pengelompokan data dari banyak kelompok penderita *tuberculosis* berdasarkan banyak atribut seperti usia, jenis kelamin, status infeksi *human immunodeficiency virus*, riwayat penyakit *diabetes mellitus*, hasil tes sinar X, dan hasil *Rapid test* dengan menggunakan model *K-Means* dan *Fuzzy C-Means* [6].

Terdapat sebuah penelitian tentang teknik *clustering* yang dilakukan oleh Ghosal dan beberapa peneliti lainnya, mereka menyebutkan bahwa model pengelompokan *Hierarchical* sudah digunakan di berbagai bidang seperti pendidikan, penggambaran pola di sosial media, mesin pencari *web*, penanganan informasi sampingan dan *metadata* di dalam sebuah dokumen, pengurangan konsumsi energi dalam sensor lintas jaringan sensor, pengelompokan perilaku *malware*, dan segmentasi gambar dengan model *hierarchical* dan model pengelompokan DBSCAN sudah digunakan pada sistem regulator anti pencucian uang, pemaksimalan keuntungan dan ekspansi pasar, manajemen keinginan pelanggan, manajemen lalu lintas jaringan, penyiapan fasilitas seperti bank, sekolah, rumah sakit dan sebagainya, dan instalasi ATM dan *e-corners* pada tempat strategis [7].

2. METODE PENELITIAN

2.1 Dataset

Dataset yang dieksplorasi pada studi ini merupakan *dataset* yang kami miliki dari penelitian kami sebelumnya pada tahun 2022 [4]. *Dataset* tersebut diperoleh dari proses web scraping dengan menggunakan teknik *parsing* HTML dari *website* yang bernama “Badan PPSDM Kesehatan Informasi SDM Kesehatan Kementerian Kesehatan Republik Indonesia versi 4.0” yang merupakan dibawah pertanggungjawaban Kementerian Kesehatan Republik Indonesia. Jika dicek kembali *website* tersebut pada tahun 2023, tampilan *website* tersebut sudah banyak berubah dan tidak dapat diperoleh kembali *dataset* yang sama persis penelitian ini miliki. Proses Web Scraping yang penelitian sebelumnya kami lakukan merupakan sebuah cara yang legal dan sah karena data tersebut bersifat publik. Proses *Web Scraping* merupakan proses yang cukup efektif dan legal untuk mengumpulkan dan melakukan ekstraksi data secara otomatis yang bekerja seperti *bot scraper* pada jaringan internet dari *website*, kemudian menyusunnya menjadi basis data terstruktur berdasarkan hasil proses dari *web scraping* [8].

Dataset yang diteliti pada eksperimen ini merupakan kumpulan dari berbagai macam tabel seperti tabel dokter spesialis, psikologi klinis, perawat, bidan, farmasi, kesehatan masyarakat, kesehatan lingkungan, nutrisonis, terapi fisik, teknisi medis, dan biomedis rekayasa. Secara keseluruhan *dataset* ini berisi data rumah sakit se-DKI Jakarta dengan detail jumlah tenaga kerja medis dan jumlah kapasitas tempat tidurnya dengan detail *dataset* yang dijelaskan secara rinci pada Tabel 1.

Tabel 1. Informasi yang Terkandung dalam *Dataset*

Nama Tabel	Deskripsi dari Jenis-jenis Tenaga Medis yang tersedia dalam Tabel	Jumlah tipe unik
Dokter Spesialis	Dokter umum, Dokter gigi, 42 jenis dari dokter spesialis yang berbeda, 10 jenis dokter gigi spesialis yang berbeda, dan 1 dokter sub spesialis	55
Psikologi Klinis	Psikolog Klinis	1
Perawat	Ners, Perawat (Non Ners), Perawat Kesehatan Anak, Perawat Maternitas, Perawat Medis Bedah, Perawat Geriatri – Lansia, Perawat Kesehatan Jiwa, Perawat Komunitas, Perawat Lainnya, dan Perawat (SPK)	10
Bidan	Bidan Klinik, Bidan Desa, Bidan Pendidik, Bidan Umum, dan Bidan (P2B, Bidan A & Bidan C)	5
Farmasi	Apoteker, Ahli Farmasi bukan Apoteker, Asisten Apoteker, Farmasi Analis, dan Farmasi (SMF)	5
Kesehatan Masyarakat	Ahli Epidemiologi Kesehatan, Promosi Kesehatan, Pakar Perilaku, Pakar Keselamatan dan Kesehatan Kerja, Pengelola dan Kebijakan Kesehatan, Statistik Kesehatan dan Demografi, Ahli Kesehatan Reproduksi dan Peranan Keluarga, Ahli Teknologi Informasi Kesehatan, Kesehatan Masyarakat Umum, Ahli Sanitasi dan Kebersihan Lingkungan, Pakar Entomolog Kesehatan, Peneliti dan ahli Mikrobiologi Medis, dan Kesehatan Lingkungan (SPPH)	9
Kesehatan Lingkungan	Sanitasi Lingkungan, Entomolog Kesehatan, Mikrobiolog Kesehatan, dan Kesehatan Lingkungan (SPPH)	4
Nutrisionis	Nutrisionis, Dietisien, dan Gizi (SPAG)	3
Terapi Fisik	Ahli Fisioterapi, Terapis Okupasi, Ahli Terapis Wicara, dan Pengobatan Akupunktur	4
Teknisi Medis	Tenaga Medis Rekam dan Informasi Kesehatan, Spesialis Teknisi Kardiovaskular, Spesialis Teknisi Pelayanan Darah, Spesialis Penglihatan, Teknisi Dental, Penata Anestesiologi, Terapis Kedokteran Gigi, Ahli Audiologi, dan Terapis Gigi dan Mulut/Perawat Gigi (SPRG)	9
Biomedis Rekayasa	Ahli Radiografi, Ahli Elektromedis, Tenaga Laboratorium Kesehatan (Analis Kesehatan), Spesialis Fisika Medis, Ahli Radioterapi, Ahli Ortotik Prostetik, dan Praktisi Analis Kesehatan (SMAK)	7

2.2 Principal Component Analysis (PCA)

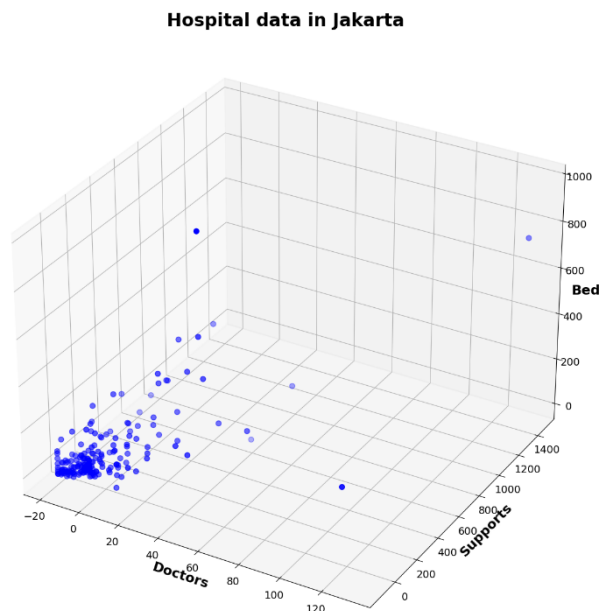
Merupakan sebuah pekerjaan yang sangat menantang untuk mengilustrasikan dimensi yang sangat besar yang terlebih lagi terdiri ratusan dimensi. *Principal Component Analysis* (PCA) yang merupakan metode matematika digunakan dalam penelitian ini untuk menurunkan jumlah dimensi sehingga data yang akan dikelompokkan dapat divisualisasikan dengan mudah untuk keperluan analisis yaitu menjadi tiga dimensi untuk penelitian ini [9].

Penelitian ini mentransformasikan 55 dimensi dari berbagai jenis dokter menjadi satu buah dimensi dokter atau sumbu Dokter, 57 dimensi dari berbagai jenis tenaga medis bukan dokter seperti psikologi klinis, keperawatan, kebidanan, farmasi, kesehatan masyarakat, kesehatan lingkungan, gizi, terapi fisik, teknisi medis, dan teknik biomedis menjadi satu buah dimensi pendukung atau sumbu pendukung.

$$x_n = W^t(t_n - \bar{t}) \quad (1)$$

Hasil dari penurunan dimensi diilustrasikan dalam persamaan (1) dengan dinotasikan dengan simbol x_n [10]. Simbol dari W merepresentasikan sumbu utama dengan $W = (w_1, w_2, \dots, w_q)$ dimana q merupakan sumbu ortonormal yang diperoleh ketika varians di bawah proyeksi maksimal, t_n mewakili sebuah set dari *vector* data d -

dimensi yang diobservasi dengan $n \in \{1, \dots, N\}$, dan \bar{t} merepresentasikan nilai rata-rata dari sampel.



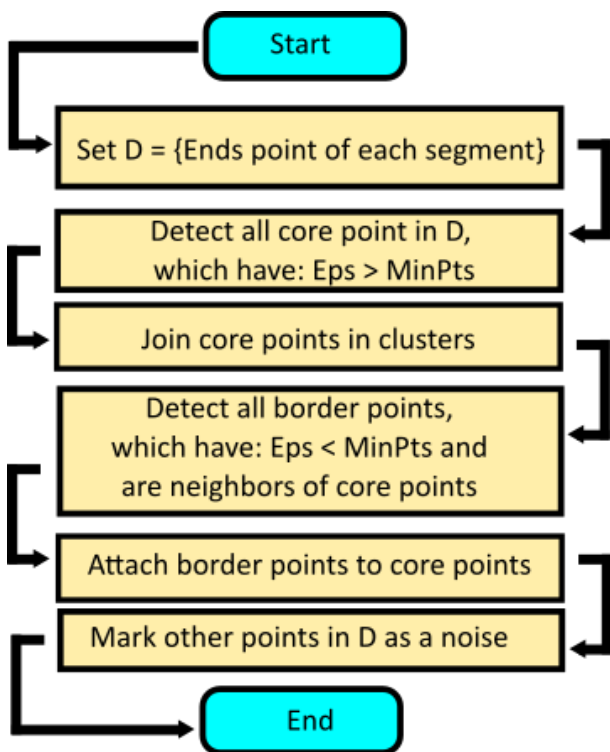
Gambar 1. Visualisasi Dataset Rumah Sakit di Jakarta dalam 3 Dimensi

Hasil dari proses PCA diperoleh nilai dimensi Dokter akan diperoleh 132,13 untuk nilai maksimum, -17,53 untuk nilai

minimum, dan $5,95 \times 10^{-16}$ untuk nilai rata-rata. Sementara untuk dimensi Pendukung diperoleh nilai 1.456,77 untuk nilai maksimum, -111,82 untuk nilai minimum, dan $4,6 \times 10^{-15}$ untuk nilai rata-rata. Sementara untuk dimensi fasilitas tempat tidur tidak melalui proses PCA sehingga datanya berupa data aktual dengan 968 sebagai nilai maksimum, 0 sebagai nilai minimum, dan 97,14 sebagai nilai rata-rata. Sehingga *dataset* dari yang digunakan dalam penelitian ini dapat divisualisasikan pada Gambar 1.

2.3 Model DBSCAN

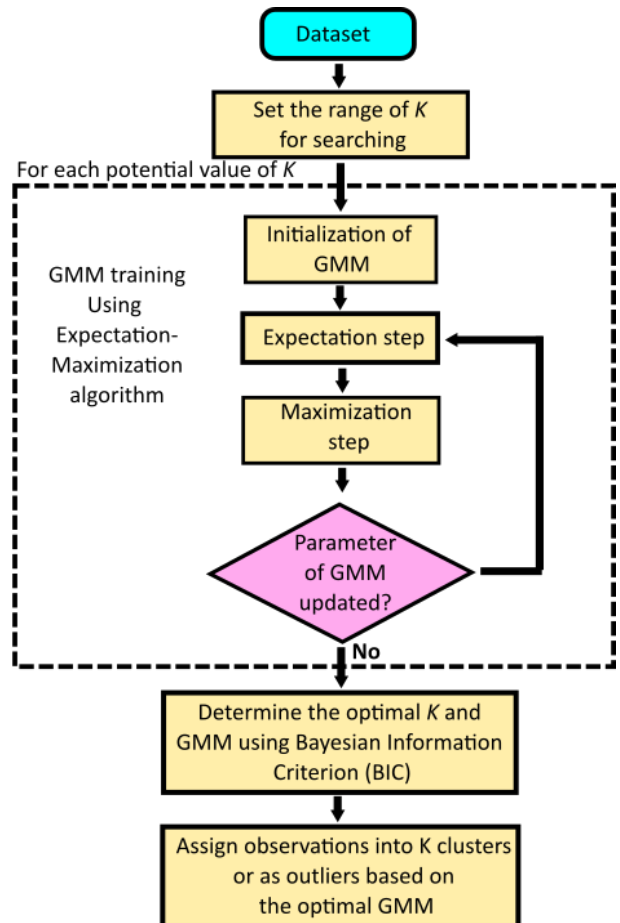
Density-based spatial clustering of applications with noise algorithm (DBSCAN) merupakan bagian dari salah satu dari pendekatan *unsupervised learning* yang merupakan algoritma *clustering* berbasis kepadatan [11], [12]. DBSCAN dapat membentuk *cluster* yang melalui penentuan area berdasarkan kepadatan tinggi yang dipisahkan dengan kepadatan yang rendah berdasarkan kepadatan cluster. DBSCAN dapat bekerja dengan baik untuk *dataset* yang berbentuk apa pun dalam *database* spasial dan efektif untuk mengelompokkan data yang sangat sulit sekalipun. Alur kerja DBSCAN dapat dilihat pada Gambar 2 dimana DBSCAN dimulai dengan titik acak. Notasikan *D* sebagai *dataset*, *Eps* sebagai radius dari lingkungan yang berdekatan dari titik data yang dipertimbangkan, dan *MinPts* sebagai jumlah minimum titik data yang berdekatan di wilayah tertentu.



Gambar 2. Flowchart dari DBSCAN

2.4 Model Gaussian Mixture (GMM)

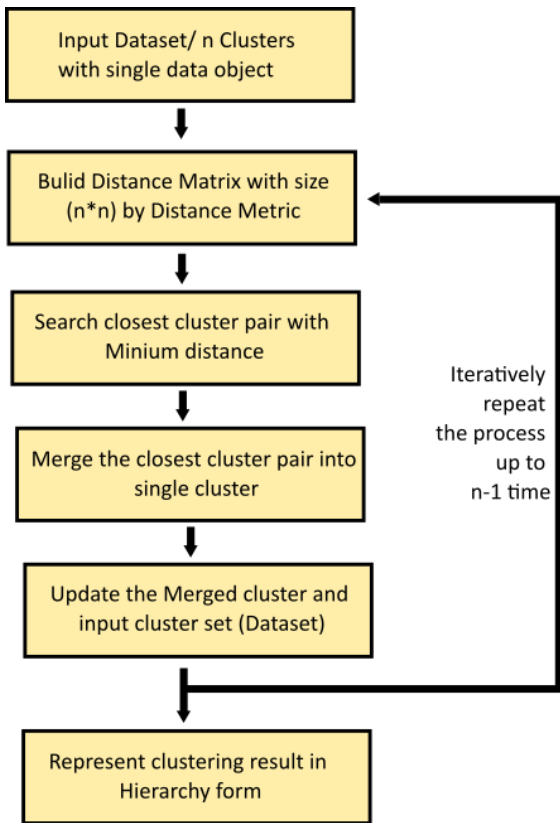
Gaussian Mixture Model (GMM) merupakan model *clustering* berbasis probabilitastas yang memanfaatkan nilai bobot dari kombinasi beberapa distribusi normal sehingga model ini sering disebut sebagai komponen campuran (*mixture component*) [13]. *Flowchart* dari GMM dapat dilihat pada Gambar 3. Algoritma *clustering* ini dapat mengidentifikasi dan menghilangkan data observasi yang memiliki probabilitas rendah dalam komponen campuran yang didasarkan pada *outliers* untuk meningkatkan kekokohan dalam hasil *clustering*.



Gambar 3. Flowchart dari GMM

2.5 Model Agglomerative Hierarchical

Model *agglomerative hierarchical cluster* (AHC) merupakan metode *clustering* hierarki *bottom-up* yang menggabungkan *N cluster* menjadi sebuah *cluster* tunggal [14]. Langkah-langkah dasar dari model AHC dapat dilihat pada Gambar 4 [15].



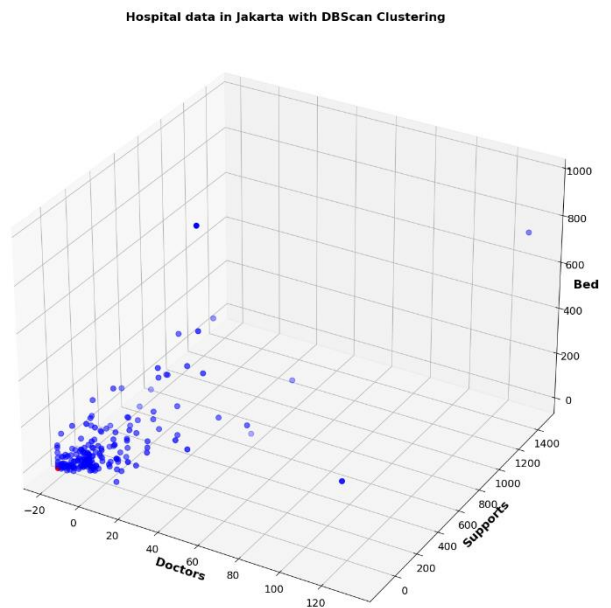
Gambar 4. Alur dari Agglomerative Hierarchical

3. HASIL DAN PEMBAHASAN

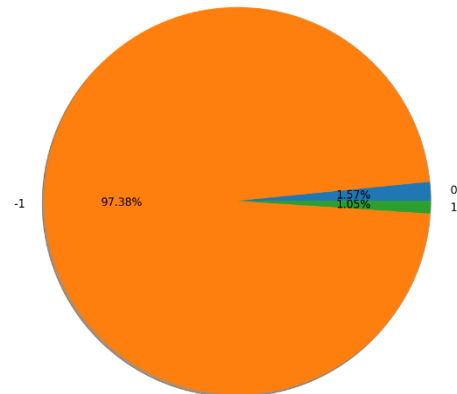
Eksperimen *clustering* dengan data rumah sakit di wilayah Provinsi Jakarta dilakukan dengan menggunakan model *clustering* DBSCAN, GMM, dan AHC yang tersedia pada *library sklearn*.

3.1 Pengelompokan dengan Model DBSCAN

Untuk mencari nilai parameter *eps* yang terbaik penelitian ini mencoba berbagai kemungkinan dalam rentang $0,3 \leq eps \leq 0,9$ dengan diperoleh nilai terbaiknya adalah $eps = 0,3$ dan diperoleh nilai *silhouette* sebesar 0,473. Dari hasil yang eksperimen dengan menggunakan model pengelompokan DBSCAN diperoleh bahwa dari total 191 rumah sakit di Jakarta dapat dikelompokkan menjadi 3 kelompok saja dimana 97,38% dikelompokkan menjadi *cluster* -1, sementara hanya 1,57% dikelompokkan menjadi *cluster* 0 dan 1,05% dikelompokkan menjadi *cluster* 1 seperti pada Gambar 6. Hasil pengelompokan dengan model DBSCAN dapat dilihat pada Gambar 5.



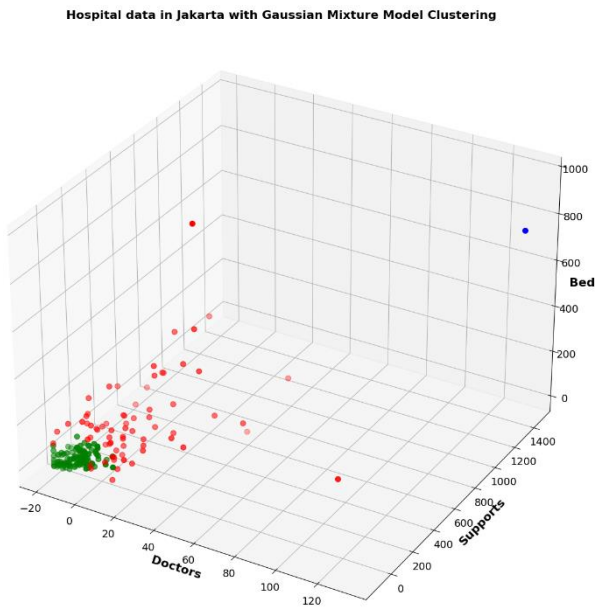
Gambar 5. Hasil pengelompokan Rumah Sakit di Jakarta dengan Model DBSCAN



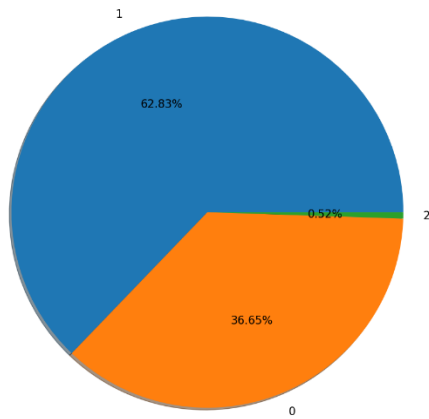
Gambar 6. Proporsi dari Hasil Pengelompokan Rumah Sakit di Jakarta dengan Model DBSCAN

3.2 Pengelompokan dengan Model Gaussian Mixture

Pada penelitian ini untuk memperoleh jumlah *cluster* yang optimal untuk model GMM ini, maka penelitian ini melakukan pencarian dengan menggunakan *elbow method* mencoba semua kemungkinan jumlah *cluster* dalam rentang $2 \leq \text{jumlah klaster} \leq 10$. Diperoleh jumlah kluster yang optimal sebanyak tiga buah *cluster* dengan nilai *silhouette* 0,441. Ilustrasi dari persebaran data rumah sakit di Jakarta dengan tiga buah *cluster* dapat dilihat pada Gambar 7. Distribusi persebaran data rumah sakit se-Jakarta dengan jumlah tiga *cluster* dapat dilihat pada Gambar 8, dimana dari total 191 rumah sakit tersebar sebanyak 62,83% masuk dalam *cluster* 1, sebanyak 36,65% masuk dalam *cluster* 0, dan sebanyak 0.52% masuk dalam *cluster* 2.



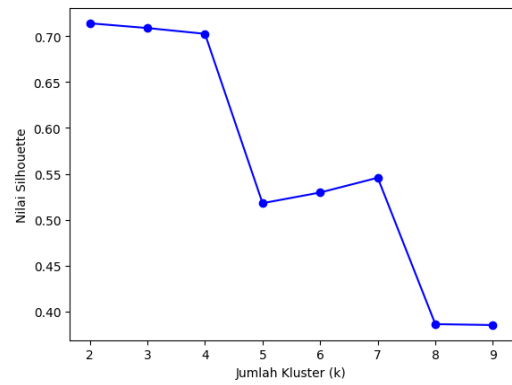
Gambar 7. Hasil Pengelompokan Rumah Sakit di Jakarta dengan Model Gaussian Mixture



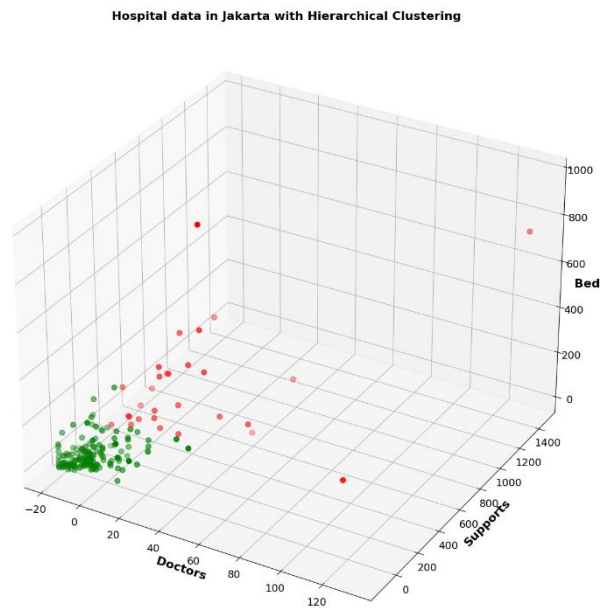
Gambar 8. Proporsi dari Hasil Pengelompokan Rumah Sakit di Jakarta dengan Model Gaussian Mixture

3.3 Pengelompokan dengan Model Hierarchical

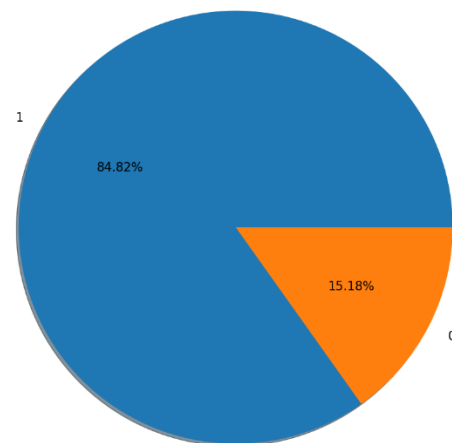
Pada eksperimen dengan model *Agglomerative Hierarchical* ini, peneliti melakukan pencarian jumlah *cluster* yang optimal dengan menggunakan *elbow method* dengan pencarian dari dua buah *cluster* hingga sembilan buah *cluster*, sehingga diperoleh hasil seperti di Gambar 9. Namun peneliti menggunakan bantuan pencarian sudut *elbow* dengan *library KneeLocator*, sehingga diperoleh jumlah *cluster* yang tepat adalah dua buah *cluster*. Visualisasi dari persebaran data rumah sakit di Jakarta dengan dua buah *cluster* dapat dilihat pada Gambar 10. Dari total 191 jumlah rumah sakit di Jakarta, sebanyak 84,82% masuk dalam kategori *cluster* 1 dan 15,18% masuk ke dalam kategori *cluster* 0 yang divisualisasikan di Gambar 11.



Gambar 9. Nilai Silhouette untuk Pencarian Jumlah Cluster yang Optimal



Gambar 10. Hasil Pengelompokan Rumah Sakit di Jakarta dengan Model Hierarchical



Gambar 11. Proporsi dari Hasil Pengelompokan Rumah Sakit di Jakarta dengan Model Hierarchical

3.4 Diskusi

Untuk membandingkan ketiga model *unsupervised* dalam melakukan pengelompokan data rumah sakit di Jakarta, penelitian ini menggunakan *index Davies Bouldin*. Index dengan *Davies Bouldin* digunakan untuk mengukur seberapa baik proses pengelompokan telah dilakukan dengan semakin rendah nilai *index* tersebut maka semakin baik hasil pengelompokan. Dengan menghitung rasio jarak antara intra-cluster dan jarak inter-cluster diperoleh bahwa model terbaik dalam kasus penelitian ini adalah model *clustering* dengan *Gaussian Mixture* seperti pada Tabel 2.

Tabel 2. Perbandingan Ketiga Model *Clustering*

<i>Clustering Model</i>	<i>Davies Bouldin Score</i>
<i>DBScan Clustering</i>	1,2258397217676922
<i>Gaussian Mixture Clustering</i>	0,6457134069852491
<i>Hierarchical Clustering</i>	0,6680066216696465

4. KESIMPULAN

Penerapan 3 model *clustering* yaitu dengan DBSCAN, *Gaussian Mixture*, dan *Agglomerative Hierarchical* dapat menghasilkan bentuk berbagai *cluster* yang berbeda-beda baik itu dari segi luas *cluster* maupun jumlah *cluster*. Dengan menggunakan DBSCAN diperoleh tiga buah *cluster* dengan proporsi 97,38%, 1,57%, dan 1,05%. Penggunaan model *Gaussian Mixture* diperoleh tiga buah *cluster* juga dengan proporsi 62,83%, 36,65%, dan 0.52%. Sementara jika dengan menggunakan model *clustering Agglomerative Hierarchical* dihasilkan dua buah *cluster* saja dengan proporsi 84,82% dan 15,18%. Berdasarkan perbandingan nilai *Davies Bouldin*, didapatkan bahwa model GMM menghasilkan nilai terendah dengan perbandingan DBSCAN menghasilkan nilai *Davies Bouldin* 89.84% lebih besar dibandingkan dengan GMM dan *agglomerative Hierarchical* menghasilkan nilai *Davies Bouldin* 3.45% lebih besar dibandingkan dengan GMM, sehingga hal ini menunjukkan bahwa model GMM adalah yang terbaik. Bagaimanapun dalam penelitian *unsupervised learning* sangatlah bergantung pada jenis *dataset* dalam kesamaan data, sehingga tidak ada jawaban yang tepat salah maupun tepat benar.

Untuk penelitian selanjutnya dimasa depan, peneliti merencanakan untuk melakukan pengumpulan *dataset* rumah sakit yang lebih luas lagi yaitu mencakup satu Negara Kesatuan Republik Indonesia baik melalui cara *web scraping* dari data publik ataupun mengajukan permintaan ke Kementerian Kesehatan Republik Indonesia.

DAFTAR PUSTAKA

[1] H. Bakri, "The planning of community health center in Indonesia," *European Journal of Research and Reflection in Management Sciences*, vol. 6, no. 3, pp. 12–18, 2018, [Online]. Available:

<http://www.idpublications.org/wp-content/uploads/2018/09/Full-Paper-THE-PLANNING-OF-COMMUNITY-HEALTH-CENTER-IN-INDONESIA.pdf>

- [2] M. H. Basri, D. A. Farmaciawaty, I. N. Rachmania, F. B. Widjaja, and A. F. Wiyana, "Healthcare Ecosystem Mapping in Primary Care: A Case Study in West Java, Indonesia," *Proceedings of the International Conference on Business and Engineering Management (ICONBEM 2021)*, vol. 177, pp. 184–190, 2021, doi: 10.2991/aebmr.k.210522.025.
- [3] Y. Mahendradhata *et al.*, "The Capacity of the Indonesian Healthcare System to Respond to COVID-19," *Front Public Health*, vol. 9, no. July, pp. 1–9, 2021, doi: 10.3389/fpubh.2021.649819.
- [4] K. E. Setiawan, A. Kurniawan, A. Chowanda, and D. Suhartono, "Clustering models for hospitals in Jakarta using fuzzy c-means and k-means," *Procedia Comput Sci*, vol. 216, pp. 356–363, 2023, doi: 10.1016/j.procs.2022.12.146.
- [5] H. Alashwal, M. El Halaby, J. J. Crouse, A. Abdalla, and A. A. Moustafa, "The application of unsupervised clustering methods to Alzheimer's disease," *Front Comput Neurosci*, vol. 13, no. May, pp. 1–9, 2019, doi: 10.3389/fncom.2019.00031.
- [6] E. M. Rochman, Miswanto, and H. Suprajitno, "COMPARISON OF CLUSTERING IN TUBERCULOSIS USING FUZZY C-MEANS AND K-MEANS METHODS," pp. 1–20, 2022, doi: 10.28919/cmbn/7335.
- [7] A. Ghosal, A. Nandy, A. K. Das, S. Goswami, and M. Panday, "A Short Review on Different Clustering Techniques and Their Applications," in *Advances in Intelligent Systems and Computing*, Springer Verlag, 2020, pp. 69–83. doi: 10.1007/978-981-13-7403-6_9.
- [8] C. Muehlethaler and R. Albert, "Collecting data on textiles from the internet using web crawling and web scraping tools," *Forensic Sci Int*, vol. 322, p. 110753, 2021, doi: 10.1016/j.forsciint.2021.110753.
- [9] J. R. Beattie and F. W. L. Esmonde-White, "Exploration of Principal Component Analysis: Deriving Principal Component Analysis Visually Using Spectra," *Appl Spectrosc*, vol. 75, no. 4, pp. 361–375, 2021, doi: 10.1177/0003702820987847.
- [10] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *J R Stat Soc Series B Stat Methodol*, vol. 61, no. 2, pp. 611–622, 2002, doi: 10.1111/1467-9868.00196.

- [11] M. Mahmudur, R. Khan, M. Abu Bakr Siddique, R. B. Arif, and M. R. Oishe, "ADBSCAN: Adaptive Density-Based Spatial Clustering of Applications with Noise for Identifying Clusters with Varying Densities." *Energy*, vol. 231, pp. 331–342, Dec. 2018, doi: 10.1016/j.apenergy.2018.09.050.
- [12] Y. Yang *et al.*, "An efficient DBSCAN optimized by arithmetic optimization algorithm with opposition-based learning," *Journal of Supercomputing*, vol. 78, no. 18, pp. 19566–19604, Dec. 2022, doi: 10.1007/s11227-022-04634-w.
- [13] K. Li, Z. Ma, D. Robinson, and J. Ma, "Identification of typical building daily electricity usage profiles using Gaussian mixture model-based clustering and hierarchical clustering," *Appl*
- [14] E. M. S. ROCHMAN *et al.*, "A combination of algorithm agglomerative hierarchical cluster (AHC) and K-means for clustering tourism in Madura-Indonesia," *Journal of Mathematical and Computational Science*, 2022, doi: 10.28919/jmcs/7086.
- [15] M. Naga Surya Lakshmi and Y. Radhika, "A comparative paper on measuring the performance of snort and suricata with variable packet sizes and speed," *Progress in Color, Colorants and Coatings*, vol. 8, no. 1, pp. 29–37, 2018, doi: 10.14419/ijet.v8i1.13971.



PENGEMBANGAN PROTOTIPE SISTEM S.K.P.I. UNTUK PENDAMPING IJAZAH BERBASIS *GOOGLE SUITE*

Nuraini Febrianti¹, Hasna Alaurrahman², Amalia Rahmah³

^{1,2,3} Sistem Informasi, Sekolah Tinggi Teknologi Terpadu Nurul Fikri
Depok, Jawa Barat, Indonesia 12640

nura20031si@student.nurulfikri.ac.id, hasn20161si@student.nurulfikri.ac.id, amaliamah2@gmail.com

Abstract

Diploma Supplement is a document containing information about fulfilling college graduates' competencies. The existence of the Merdeka Learning-Independence Campus (MBKM) program encourages students to acquire valuable competencies so that a portfolio of abilities and experiences can be accommodated in the Diploma Supplement. In this research, the Nurul Fikri Integrated Technology College still needs to implement Diploma Supplement because no procedures and information systems support Diploma Supplement management. Therefore, researchers will develop a prototype of the Diploma Supplement system for a Google Suite-based diploma companion that is comprehensive and can make it easier for the academic community. This research methodology uses the Scrum framework method. The entire work process for Diploma Supplement utilizes Google Suite, especially Google Drive, Google Docs, Google Forms, and Google Sheets with the AutoCrat add-on. This research successfully developed a prototype of the Diploma Supplement system for a Google Suite-based diploma companion. The results of this study can make it easier for students to process Diploma Supplement submissions because it is done online, BAAK also does not need to make Diploma Supplement manually because the system is already integrated, and campuses can save costs for creating, managing, and maintaining the system.

Keywords: AutoCrat, Google Form, Google Suite, MBKM, and SKPI

Abstrak

Surat Keterangan Pendamping Ijazah (SKPI) adalah dokumen yang memuat informasi tentang pemenuhan kompetensi lulusan perguruan tinggi. Adanya program Merdeka Belajar-Kampus Merdeka (MBKM) mendorong mahasiswa memperoleh kompetensi yang berguna sehingga portofolio kemampuan dan pengalaman dapat terwadahi di dalam SKPI. Pada penelitian ini Sekolah Tinggi Teknologi Terpadu Nurul Fikri belum mengimplementasikan SKPI karena belum adanya prosedur dan sistem informasi yang mendukung untuk pengelolaan SKPI. Oleh karena itu peneliti akan mengembangkan prototipe sistem SKPI untuk pendamping ijazah berbasis *Google Suite* yang komprehensif dan dapat memudahkan civitas akademika. Metodologi penelitian ini menggunakan metode kerangka kerja *Scrum*. Seluruh proses pengerjaan SKPI memanfaatkan *Google Suite*, terutama pemanfaatan *Google Drive*, *Google Docs*, *Google Form*, dan *Google Sheets* dengan *add-on AutoCrat*. Penelitian ini berhasil mengembangkan prototipe sistem SKPI untuk pendamping ijazah berbasis *Google Suite*. Hasil penelitian ini dapat memudahkan mahasiswa memproses pengajuan administrasi SKPI karena dilakukan secara *online*, BAAK juga tidak perlu membuat SKPI secara manual karena sistem sudah terintegrasi, serta kampus dapat menghemat biaya pengeluaran pembuatan, pengelolaan, dan pemeliharaan sistem.

Kata kunci: AutoCrat, Google Suite, Google Form, MBKM, dan SKPI

1. PENDAHULUAN

Surat Keterangan Pendamping Ijazah (SKPI) merupakan dokumen yang diterbitkan oleh perguruan tinggi yang memuat informasi tentang pemenuhan kompetensi lulusan pendidikan akademik dan vokasi [1]. Salah satu kebijakan pemerintah yang mendukung pemenuhan kompetensi lulusan yakni program Merdeka Belajar-Kampus Merdeka (MBKM). Sebagaimana Peraturan Menteri Pendidikan dan

Kebudayaan RI No. 3 Tahun 2020, mahasiswa diberikan hak untuk bebas mengambil SKS di luar program studi maksimal 3 semester, yaitu 1 semester di luar program studi di kampus yang sama dan 2 semester melaksanakan aktivitas pembelajaran di luar perguruan tinggi. Bentuk kegiatan belajar di luar perguruan tinggi, diantaranya; melakukan magang/praktek kerja di Industri, melaksanakan proyek pengabdian kepada masyarakat, mengajar di satuan

pendidikan, mengikuti pertukaran mahasiswa, melakukan penelitian, melakukan kegiatan kewirausahaan, membuat studi/proyek independen, dan mengikuti program kemanusiaan [2]. Adanya program MBKM bertujuan untuk mendorong mahasiswa menguasai berbagai keilmuan yang berguna untuk memasuki dunia kerja yang sangat kompetitif dan mendorong mahasiswa memperoleh pengalaman belajar, dengan berbagai kompetensi tambahan di luar program studi dan/atau di luar kampusnya. Tujuan utama MBKM adalah meningkatkan kompetensi lulusan, baik dalam *soft skills* maupun *hard skills*, untuk lebih sesuai dengan kebutuhan saat ini dan menghasilkan pemimpin masa depan yang berkualitas.

Kebijakan MBKM sejalan dengan Permendikbud No. 3 Tahun 2020 tentang Standar Nasional Pendidikan Tinggi. Perguruan tinggi, fakultas, program studi, mahasiswa, dan mitra terlibat dalam melaksanakan aturan ini. Perguruan tinggi perlu menyediakan opsi bagi mahasiswa untuk mengambil mata kuliah di luar program studi mereka, dan fakultas harus menyusun daftar mata kuliah lintas program studi serta kerja sama dengan mitra yang relevan [3]. Salah satu cara konkret untuk menggambarkan pencapaian pembelajaran mahasiswa adalah dengan digunakan penerbitan Surat Keterangan Pendamping Ijazah atau SKPI yang sesuai dengan ketentuan Permendikbud No. 73 Tahun 2013 [4]. Dengan adanya SKPI, rekam jejak mahasiswa selama mengikuti perkuliahan dan program MBKM dengan berbagai bentuk kegiatan pembelajaran yang menjadi portofolio kemampuan dan pengalaman dapat terwadahi di dalamnya.

Selain itu dengan menerbitkan SKPI maka implementasi kebijakan MBKM secara substansial dapat mendukung pengembangan dan peningkatan mutu pendidikan tinggi dalam menghasilkan lulusan yang memiliki kemampuan *softskill* dan *hardskill*, mumpuni dalam aspek *leadership* dan kepribadian. Beberapa kampus sudah menjalankan penerbitan SKPI, ada yang sudah berbasis teknologi sistem informasi dan ada pula yang masih manual menggunakan formulir cetak. Akan tetapi, Sekolah Tinggi Teknologi Terpadu Nurul Fikri (STT-NF) merupakan salah satu kampus yang tergolong belum menerbitkan SKPI karena merupakan kampus yang masih tergolong baru berdiri dan masih lebih fokus terhadap data akademik dibandingkan data non akademik.

Maka dari itu, sampai sekarang prestasi dan kegiatan non akademik mahasiswa belum tertata, selain itu kompetensi yang diperoleh dari program MBKM juga tidak terdokumentasi dengan baik, sehingga kampus belum bisa menerbitkan SKPI sebagai dokumen resmi yang mendukung kompetensi dan portofolio mahasiswa dalam waktu yang singkat. Adapun permasalahan lainnya yaitu berdasarkan informasi yang peneliti dapatkan melalui wawancara dengan staf Biro Administrasi Akademik dan Kemahasiswaan (BAAK) di STT-NF, bahwa STT-NF belum memiliki prosedur dan sistem informasi yang

mendukung untuk pengelolaan SKPI. Hal tersebut dikarenakan pembuatan sistem informasi berbasis *website* atau *mobile* membutuhkan waktu, sumber daya, dan pemeliharaan sistem dengan biaya yang tidak sedikit.

Untuk mengatasi permasalahan tersebut dibutuhkan suatu teknologi informasi digunakan dalam pelaksanaan tugas administrasi dengan tujuan mewujudkan tata kelola yang efisien (*good governance*) serta menyediakan pelayanan yang cepat, akurat, dan sederhana. Penggunaan teknologi informasi menjadi suatu keharusan dalam merancang sistem informasi yang bertujuan untuk mengelola data guna menghasilkan informasi berkualitas. Sistem informasi ini memungkinkan otomatisasi tugas dan pelayanan, termasuk dalam pengelolaan sumber daya manusia [5]. Sistem informasi SKPI yang komprehensif dan memudahkan pihak kampus dalam proses pengambilan data non akademik mahasiswa, validasi data, hingga penerbitan SKPI serta memudahkan mahasiswa dalam pemasukan data, *follow up* status pengajuan, sampai penerimaan SKPI yang efektif. Salah satu solusi yang dapat dilakukan yaitu pemaduan berbagai fitur *Google* yang gratis, mudah dipelajari, dan mudah diakses kapan pun dan di mana pun selama terkoneksi internet.

Maka dari itu, seluruh proses pengerjaan SKPI memanfaatkan *Google Suite*, terutama pemanfaatan *Google Drive*, *Google Docs*, *Google Form*, dan *Google Sheets* dengan *add-on AutoCrat*. *AutoCrat* adalah salah satu *add-on* yang tersedia di *Google Spreadsheet* untuk membuat laporan dengan mekanisme *mail merge*, yang dapat dikirim kepada responden dalam format *file word* atau *pdf*, baik langsung maupun dikirim melalui *email* [6]. Pemanfaatan *Google Suite* selain memudahkan pengelolaan SKPI juga dapat meminimalkan biaya pengembangan sistem. *Autocrat* telah sering dimanfaatkan untuk mendukung otomatisasi tugas di berbagai sektor seperti administratif dalam konteks pendidikan, pemerintahan, dan upaya pemasaran di sebuah perusahaan [7]. Oleh karena itu, penelitian ini akan mengembangkan prototipe sistem SKPI untuk pendamping ijazah berbasis *Google Suite* sebagai implementasi awal dan gambaran prototipe untuk kedepannya dapat diimplementasikan di sistem informasi akademik atau AIS STT-NF.

Surat Keterangan Pendamping Ijazah (SKPI) atau *diploma supplement* adalah dokumen resmi yang diberikan oleh perguruan tinggi, memuat informasi mengenai prestasi akademik atau kualifikasi dari lulusan pendidikan tinggi dengan gelar yang didapatkan oleh mahasiswa. Prestasi akademik lulusan dijelaskan dalam bentuk naratif deskriptif yang merinci pencapaian pembelajaran pada tingkat KKNI yang relevan, dengan format standar yang mudah dimengerti oleh masyarakat umum. SKPI tidak menggantikan fungsi ijazah maupun transkrip akademik, dan juga tidak secara otomatis memberikan pengakuan kepada pemegangnya [8]. Adapun manfaat dari penelitian

ini, lulusan STT-NF mendapatkan pengakuan kualifikasi yang ia miliki berdasarkan dokumen resmi yang memuat kompetensinya sehingga dapat lebih siap untuk bersaing di dunia kerja.

2. METODE PENELITIAN

Pada penelitian ini metode pengumpulan data berupa sumber data primer dan data sekunder. Sumber data primer berasal dari diskusi terfokus dengan dosen pembimbing dan kelompok, wawancara dengan Biro Administrasi Akademik dan Kemahasiswaan (BAAK), forum grup diskusi dengan Kepala Program Studi Sistem Informasi (SI) Teknik Informatika (TI), Bisnis Digital (BD) dan wawancara dengan Kemahasiswaan. Sumber data sekunder berasal dari hasil eksplorasi di internet.

Metode pengembangan yang digunakan adalah *Agile*. *Agile* adalah kerangka berfikir yang berfokus kepada pengembangan iteratif dan fleksibel mengikuti kebutuhan *customer*. Kerangka kerja untuk mengimplementasikan *Agile* menggunakan *Scrum*. *Scrum* ini menggunakan pendekatan iteratif dan inkremental untuk mengoptimalkan prediktabilitas serta pengendalian terhadap risiko [9]. Kerangka kerja ini dituangkan ke dalam tahapan penelitian sebagai berikut:

2.1. Identifikasi Masalah

Pada tahap identifikasi ini peneliti melakukan proses pendalaman terhadap permasalahan objek penelitian.

2.2. Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk menentukan komponen SKPI, terdiri dari:

- Identitas pemegang SKPI berisi data diri mahasiswa.
- Informasi tentang identitas penyelenggara program berisi informasi tentang sekolah tinggi dan program studi.
- Informasi tentang hasil pencapaian berisi tentang Capaian Pembelajaran Lulusan (CPL), capaian

keahlian, pengalaman organisasi, pengalaman non organisasi, prestasi/penghargaan, serta kegiatan MBKM yang didapat oleh mahasiswa selama perkuliahan.

- Informasi tambahan program studi berupa visi, misi, dan tujuan dari program studi yang ditempuh.

Selain itu, menganalisis aktor yang terlibat untuk dituangkan ke dalam *product backlog*.

2.3. Pengembangan Sistem

Perancangan dan Pengembangan Sistem SKPI yang akan dibangun memanfaatkan *Google Suite*, fitur yang digunakan adalah *Google Drive*, *Google Docs*, *Google Form*, *Google Sheets* dengan *add-on AutoCrat*. Hasil responden dari *Google Form* akan masuk ke *Google Sheets* untuk proses *data cleaning*, verifikasi bukti, dan validasi keseluruhan data. Apabila ada kesalahan data/bukti pada capaian mahasiswa, maka BAAK & Kemahasiswaan akan menginformasikan data yang perlu direvisi dan BAAK menginformasikan batas revisian. Ketika semua data sudah tervalidasi maka *Google Docs* dan *Google Sheets* diintegrasikan dengan fitur *add-on AutoCrat*. Sehingga SKPI akan otomatis tercetak dan BAAK memberikan ke Kepala Program Studi untuk pengesahan dokumen SKPI. Setelah itu BAAK menginformasikan jadwal pengambilan, apabila dokumen sudah diambil oleh mahasiswa BAAK mengubah status pengambilan. Mahasiswa dapat melihat secara berkala pengajuan dokumen SKPI nya pada *link* Status Pengajuan.

3. HASIL DAN PEMBAHASAN

3.1. Product Backlog dan Sprint Backlog

Hasil dari proses identifikasi masalah dan analisis kebutuhan, dibuat menjadi dokumen *product backlog*. Tabel 1 contoh *product backlog* berisi daftar pekerjaan yang dibutuhkan dalam pengembangan sistem.

Tabel 1. Tabel *Product Backlog*

Aktor	<i>I want to be able to ...</i>	<i>So that ...</i>	Priority	Sprint	Status
Mahasiswa	Mengisi <i>link form</i> pengajuan	Data tercatat dalam <i>Sheets</i> <i>must</i> SKPI	9		<i>DONE</i>
Mahasiswa	Mengupload bukti data nonakademik	Data nonakademik terpercaya <i>must</i>	9		<i>DONE</i>
Mahasiswa	Melihat informasi status pengajuan	Dapat segera memperbaiki data yang salah atau pengambilan dokumen yang sudah disetujui <i>must</i>	10		<i>DONE</i>
BAAK	Melakukan <i>data cleaning</i> form response	Menghindari terjadinya data yang tidak sesuai format, <i>duplicate</i> , dan data kosong <i>must</i>	9		<i>DONE</i>
BAAK	Melakukan proses verifikasi Identitas Pemegang SKPI bukti MBKM	Memastikan data sesuai dan keabsahan dari berkas MBKM <i>must</i>	9		<i>DONE</i>
BAAK	Melakukan validitas data dengan mengubah status validasi	Menandai data mana saja yang valid atau tidak valid <i>must</i>	9 & 15		<i>DONE</i>

Aktor	I want to be able to ...	So that ...	Priority	Sprint	Status
BAAK	Generate data menggunakan Template AutoCrat	SKPI akan terisi sesuai dengan data yang telah di generate untuk ditandatangani oleh KaProDi	must	9	DONE
Kemahasiswaan	Melakukan data cleaning form response	Menghindari terjadinya data yang tidak sesuai format, dan duplicate	must	16	DONE
Kemahasiswaan	Melakukan proses verifikasi berkas/bukti Sertifikat Keahlian, Prestasi/Penghargaan, Organisasi, dan Non-Organisasi	Memastikan keabsahan dari must	must	16	DONE
Kemahasiswaan	Mengisi dan mengedit kolom revisi	Memberikan catatan pada data yang perlu direvisi dan menghapusnya ketika data sudah direvisi	must	16	DONE

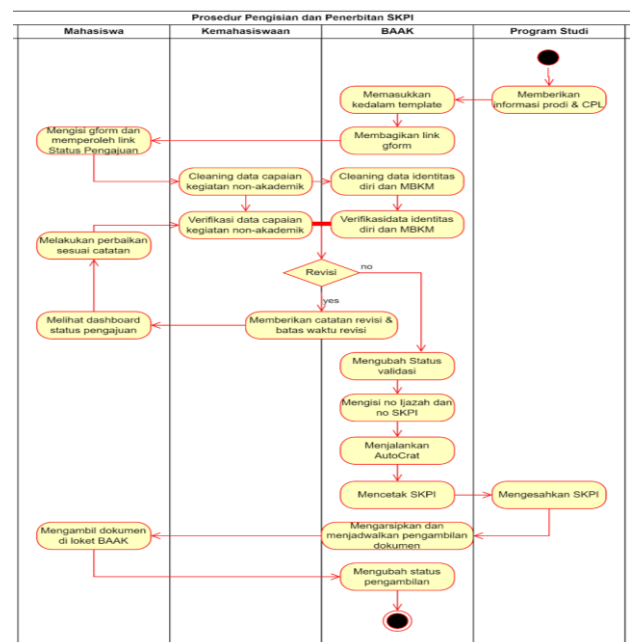
Kemudian pada Tabel 2 contoh *sprint backlog* yang berisi pembagian tugas untuk mengerjakan fitur-fitur yang dipilih dari *product backlog*.

Tabel 2. Tabel *Sprint Backlog*

Item Backlog	Task	PIC
Form Pengajuan	Membuat halaman isian Identitas Pemegang SKPI	Aini
	Melakukan uji coba <i>gform</i> dengan mengisi data diri dan prestasi	Aini & Hasna
File Pengajuan (Response)	Membuat sheet data untuk proses <i>data cleaning</i> yang dilakukan oleh BAAK	Hasna
	Mengintegrasikan template dokumen dengan data sheet SI & TI menggunakan <i>AutoCrat</i>	Hasna
	Mengintegrasikan template dokumen dengan data sheet BD menggunakan <i>AutoCrat</i>	Aini
File Status Pengajuan	Mengintegrasikan <i>file Pengajuan (Response)</i> dengan <i>file Status Pengajuan</i> untuk menampilkan data tertentu	Aini & Hasna
	Melakukan uji coba status pengajuan	Aini & Hasna

3.2. Prosedur pengisian dan penerbitan SKPI

Proses pengisian dan penerbitan SKPI dalam sistem yang dibangun berbasis *Google Suite* ini melibatkan mahasiswa, Kepala Program Studi (KaProDi), Kemahasiswaan, dan BAAK. Sistem ini memanfaatkan *add-on AutoCrat* untuk dapat menjalankan mekanisme *mail merge* dalam menghasilkan dokumen SKPI secara otomatis. Data dari *Google Form* akan disimpan pada sebuah *file Google Sheet*. Selanjutnya *AutoCrat* secara otomatis membuat *file merge* dari *template SKPI* menjadi sebuah *file doc*, hasilnya data akan berbeda-beda tergantung dari *file Google Sheet*. Berdasarkan Gambar 1, prosedur dimulai dari KaProDi memberikan CPL ke BAAK untuk dimasukkan ke dalam *template SKPI*. Proses pengisian SKPI diawali dengan BAAK membagikan *link Google Form* ke mahasiswa agar dapat mengisi data diri serta capaian sertifikat keahlian, prestasi/penghargaan, pengalaman baik organisasi maupun non organisasi, kegiatan MBKM beserta bukti/berkas pendukungnya.



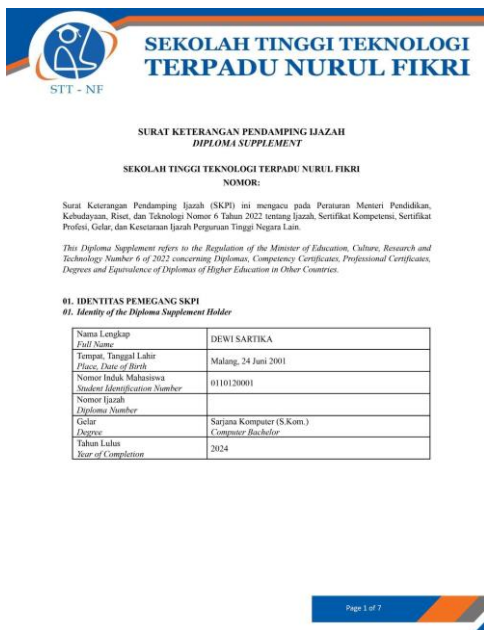
Gambar 1. Flowchart Prosedur Pengisian dan Penerbitan SKPI

Selanjutnya, hasil isian ini perlu diverifikasi oleh Kemahasiswaan dan BAAK. Data dinyatakan valid apabila tidak ada catatan revisi, sebaliknya jika ada catatan revisi maka data dianggap tidak valid dan harus direvisi oleh mahasiswa. Pemeriksaan hasil revisi akan dilakukan secara berkala sesuai dengan batas waktu revisi yang ditentukan oleh BAAK. BAAK menjalankan *AutoCrat* untuk memproses penerbitan SKPI pada data valid. Kemudian, dokumen SKPI dapat dicetak dan disahkan dengan cap basah serta tanda tangan dari KaProDi. BAAK menginformasikan jadwal pengambilan dokumen kepada mahasiswa. Selama proses tersebut mahasiswa dapat memantau proses SKPI pada *dashboard* status pengajuan.

3.3. Template SKPI

Proses pengembangan sistem dimulai dari perancangan desain *template* yang disesuaikan dengan komponen SKPI. Pada Gambar 2 berisi identitas pemegang SKPI. Pada Gambar 3 berisi informasi tentang capaian keahlian, pengalaman organisasi, pengalaman non organisasi,

prestasi/penghargaan, serta kegiatan MBKM yang didapat oleh mahasiswa selama berkuliah. *Template* akan dimasukkan ke dalam Sistem SKPI berbasis *Google Suite* untuk diproses secara otomatis agar dapat terisi dengan data mahasiswa calon lulusan.



Gambar 2. Komponen dari *Template* SKPI yang Akan Terisi Otomatis



Gambar 3. Komponen dari *Template* SKPI yang Akan Terisi Otomatis

3.4. Form Pengajuan Administrasi SKPI

Untuk menghasilkan dokumen SKPI yang sesuai dengan *template* SKPI, pihak kampus memerlukan dokumentasi data kegiatan non akademik yang mencakup sertifikat keahlian, prestasi/penghargaan, pengalaman baik organisasi maupun non organisasi, kompetensi dari program MBKM yang diperoleh mahasiswa selama berkuliah, dan bukti/berkas dari semua capaian tersebut. Hal tersebut menjadi alasan untuk membuat *form* menggunakan *Google Form* agar memudahkan pengambilan data dan pengajuan mahasiswa lebih efektif dan efisien.

3.5. Dashboard Pengajuan Administrasi SKPI

Setelah mahasiswa mengisi *form*, data akan masuk ke *dashboard* yang dibuat menggunakan *Google Sheets*. Di dalam *dashboard* ada beberapa *sheet*. *Sheet Form*, SI, TI, BD dapat diedit oleh BAAK, sedangkan *sheet* Kema & BAAK dapat diedit oleh Kemahasiswaan dan BAAK. *Dashboard* ini didesain untuk BAAK dan Kemahasiswaan. Proses *data cleaning*, verifikasi, dan validasi terdapat di *sheet Form* [BAAK] dan *sheet* Kema & BAAK. Setelah tervalidasi, akan masuk ke *sheet* prodi lalu diintegrasikan dengan *template* SKPI memanfaatkan *add-on AutoCrat* pada *Google Sheets*. Proses pembuatan SKPI menggunakan fitur *add-on AutoCrat* akan mengubah isian pada *template* dengan tanda kurung sudut (<< >>) menjadi data yang sesuai pada pengisian *form* setelah divalidasi.

3.6. Dashboard Status Pengajuan SKPI

Setelah mahasiswa mengisi *form*, mahasiswa dapat mengecek secara berkala di Status Pengajuan untuk progres pengajuan dokumen SKPI. Pada Gambar 5 berisi informasi NIM, Nama Lengkap, Tahun Lulus, Keterangan Revisi, Batas Revisi, Pengambilan Dokumen, dan Status Pengambilan.

	A	B	C	D	E	F	G
1		Nama Lengkap	Tahun Lulus	Keterangan Revisi	Batas Revisi	Pengambilan Dokumen	Status Pengambilan
2	0110120031	NURAINI FEBRIANTI	2024	-		Wednesday, 12 J	Monday, 17 Jun
4	0110120030	SHAKILA ZAYN	2024	bukti organisasi tidak lengkap	Friday, 3 May 2024	Friday, 14 June 2	Friday, 14 June
5	0110120001	AHMAD HAIKAL	2024	bukti MBKM belum valid	Friday, 3 May 2024		

Gambar 4. Dashboard Status Pengajuan SKPI Mahasiswa

3.7. Evaluasi

Evaluasi merupakan catatan berupa *feedback*/umpan balik yang diperoleh setelah melakukan pengujian dengan beberapa pihak terkait. Hasil pengujian dengan BAAK memperoleh beberapa *feedback* terhadap *form*, *sheets*, dan *Standard Operational Procedure (SOP)*. Tabel 3 merupakan daftar *feedback* dari BAAK yang berhasil diimplementasikan pada penelitian ini.

Tabel 3. Tabel Evaluasi dari BAAK

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
Menghilangkan nomor Ijazah pada <i>Gform</i>	Bisa disesuaikan di <i>Gform</i>	Sudah diperbaiki
Menambahkan deskripsi di section Identitas Pemegang SKPI bahwa data yang diinput harus sesuai dengan Ijazah Terakhir	Bisa disesuaikan di <i>Gform</i>	Sudah diperbaiki
Menambahkan nomor Whatsapp pada <i>Gform</i>	Bisa disesuaikan di <i>Gform</i>	Sudah diperbaiki
Mengganti format kolom status pengambilan dari teks dropdown menjadi tanggal	Bisa disesuaikan di <i>GSheets</i>	Sudah diperbaiki

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
Menambahkan kolom Nomor Ijazah dan Nomor SKPI di Sheets	Bisa disesuaikan di <i>GSheets</i>	Sudah diperbaiki
Menambahkan timeline pada SOP	Bisa disesuaikan di SOP	Sudah diperbaiki
Pencetakan dan pengesahan SKPI hanya dilakukan sekali disamakan di SOP seperti Ijazah dan Transkrip Akademik	Bisa disesuaikan di SOP	Sudah diperbaiki

Evaluasi dilakukan dengan pengujian Sistem Pengajaran SKPI dengan para KaProdi, yaitu dari Prodi Sistem Informasi, Teknik Informatika, dan Bisnis Digital, serta pihak Kemahasiswaan. Ada pun masukan dari KaProdi Sistem Informasi disampaikan Tabel 4. Beberapa masukan dapat langsung diimplementasikan pada penelitian ini, namun ada beberapa masukan yang dituangkan sebagai saran yang direkomendasikan untuk penelitian selanjutnya, seperti penggunaan *form* selain *Google Form*.

Tabel 4. Tabel Evaluasi dari Kepala Program Studi Sistem Informasi

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
CPL disatukan menjadi 14 poin	Bisa disesuaikan di <i>template</i> SKPI	sisa Digital karena masih dalam proses CPL nya
No. surat SKPI bisa dikeluarkan dari Prodi	Tidak ada perubahan dengan sebelumnya	-
Sertifikat keahlian, prestasi dan penghargaan ditambahkan menjadi 5 inputan	Bisa disesuaikan di <i>GForm</i>	Sudah diperbaiki
Mengimplementasi Form langsung dari Google Sheet bukan menggunakan Google Form, untuk memudahkan format penulisan Capaian Aktivitas, Prestasi, dan Penghargaan	Bisa diimplementasikan di penelitian selanjutnya	-
Menambahkan deskripsi kewirausahaan dengan format [Jenis Kegiatan - Judul Bisnis - Nama Penyelenggara - Periode] di Form MBKM	Bisa disesuaikan di <i>Gform</i>	Sudah diperbaiki

Hasil pengujian dengan KaProdi Teknik Informatika memperoleh beberapa *feedback* terhadap *template*, SOP, dan Prosedur yang berhasil diimplementasikan pada penelitian ini. Tabel 5 merupakan daftar *feedback* KaProdi Teknik Informatika dan respon peneliti.

Tabel 5. Tabel Evaluasi dari Kepala Program Studi Teknik Informatika

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
Mengkonfirmasi bagian KKNI dan menambahkan KKNI berbasis OBE	Bisa disesuaikan di <i>template</i>	dan menghapuskan dari <i>template</i> karena CPL sudah berbasis KKNI OBE
Mengkonfirmasi batasan lisensi <i>AutoCrat</i>	Bisa dikonfirmasi	Tergantung dari akun Google yang dipakai
Durasi revisi mahasiswa ditambahkan menjadi 7 hari	Bisa disesuaikan di SOP	Sudah diperbaiki
Durasi <i>cleaning</i> menjadi 7 hari	kedua Bisa disesuaikan di SOP	Sudah diperbaiki
Narasi prosedur dan detail terutama yang perlu dipastikan oleh aktor/unit tertentu	diperjelas Bisa disesuaikan di SOP	Sudah diperbaiki
Bagan alir dibuat landscape agar lebih jelas	landscape Bisa disesuaikan di SOP	di Sedang diperbaiki
Standar mutu di SOP ditanyakan ke LPPMI	perlu Menunggu konfirmasi	Peraturan dari kampus belum ada
Mengikutsertakan Kemahasiswaan validator data mahasiswa	unit sebagai wawancara organisasi mahasiswa	Berdasarkan hasil wawancara dengan data organisasi mahasiswa cukup data dibuktikan dengan sertifikat dan divalidasi oleh BAAK

Hasil pengujian dengan KaProdi Bisnis Digital memperoleh beberapa *feedback* terhadap *template* dan SOP. Tabel 6 merupakan daftar *feedback* dari KaProdi BD dan dapat diimplementasikan pada penelitian ini. Namun ada saran yang peneliti rekomendasikan untuk penelitian selanjutnya.

Tabel 6. Tabel Evaluasi dari Kepala Program Studi Bisnis Digital

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
Informasi tentang Kualifikasi.... Prestasi/Penghargaan	diganti menjadi Informasi Lulusan dan	Diubah menjadi Informasi Tentang Hasil Pencapaian
Informasi tentang Sistem Pendidikan Tinggi .. menjadi Informasi tentang Prodi Sistem Informasi (per prodi) STT Terpadu Nurul Fikri (deskripsi dan visi) atau dihapuskan	diganti ..	Bisa disesuaikan di <i>template</i>
Template KKNI dihapuskan		Bisa disesuaikan di <i>template</i>
Judul pengesahan dihapuskan, lalu dan di cek kembali alasan ttd harus di	langsung tanda tangan di kanan atau di kiri	Sudah disesuaikan di <i>template</i> SKPI. Tanda tangan dipindah jadi di kanan karena menyamakan dengan Transkrip Akademik dan sejalan dengan PerMen Riset, teknologi, dan Pendidikan Tinggi RI No. 59 Tahun 2018 Pasal 11 Ayat 2 Huruf B

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
MBKM dijadikan bab terpisah	Bisa diimplementasikan di penelitian selanjutnya	-
Menambahkan 1 bab tambahan di SOP tentang template SKPI (kalau bisa dijelaskan isinya)	Penjelasan mengenai isi SKPI terdapat pada bab 2 tentang SKPI STT-NF bagian komponen SKPI	Sudah diperbaiki
Standar Mutu di SOP: isi permenristek dikti dan peraturan Kepala STT Terpadu Nurul Fikri	Bisa disesuaikan di SOP	Sudah diperbaiki
Mengikutsertakan unit Kemahasiswaan di dalam (validator data prestasi/penghargaan dan organisasi mahasiswa) atau di luar (hanya menambahkan deskripsi bahwa inputan data prestasi/penghargaan dan organisasi berdasarkan data yang sudah terdata di Kemahasiswaan) SKPI	Bisa disesuaikan agar Kemahasiswaan dapat terlibat di dalam proses penerbitan SKPI	Sudah diperbaiki. Kemahasiswaan dilibatkan untuk memverifikasi/validasi data nonakademik

Hasil pengujian dengan Kemahasiswaan memperoleh beberapa feedback terhadap *template* dan *Google Form*. Tabel 7 merupakan daftar masukan dari Kemahasiswaan. Beberapa dapat diimplementasikan pada penelitian ini, namun saran untuk membuat infografis SKPI direkomendasikan pada penelitian selanjutnya.

Tabel 7. Tabel Evaluasi dari Kemahasiswaan

Saran Perbaikan	Respon Perbaikan	Status Perbaikan
Menambahkan kemampuan bahasa section dengan mencantumkan skor TOEFL/IELTS/dsb	Bisa disesuaikan di <i>gform</i>	Sudah diperbaiki
Mahasiswa tidak perlu mengkonfirmasi kembali ke validator(kemas/baak), melihat status pengajuan secara berkala	Bisa diterapkan cukup	Sudah diterapkan
Membuat infografis mengenai SOP SKPI	Bisa diimplementasikan di penelitian selanjutnya	-
Mengonfirmasi kembali ke BAAK mengenai durasi prosedur SKPI	Sudah dikonfirmasi	Sudah dikonfirmasi. BAAK akan meninjau ulang kembali mengenai fiksasi timeline SKPI.

Untuk mendukung penerapan Sistem Pengajuan SKPI, maka diperlukan adanya *Standard Operational Procedure* (SOP) SKPI, yang merupakan pedoman dalam melaksanakan fungsi dan tugas yang berkaitan dengan alur kerja penerbitan SKPI beserta bagian-bagian yang berperan di dalamnya. SOP SKPI mencakup bagian definisi, tujuan, standar mutu, lingkup prosedur, komponen, ketentuan, durasi, prosedur pengisian dan penerbitan, bagan alir, dan template SKPI yang telah dicetak. Dokumen ini dibuat agar pihak yang terkait dapat mengetahui tata cara pengisian SKPI dan menjamin proses kerja penerbitan SKPI di Sekolah Tinggi Teknologi Terpadu Nurul Fikri dapat berjalan dengan baik.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan prototipe sistem SKPI untuk pendamping ijazah berbasis *Google Suite*. Pada

pengembangan sistem sederhana ini dapat memudahkan mahasiswa memproses pengajuan administrasi SKPI karena dilakukan secara *online*. BAAK juga tidak perlu membuat SKPI secara manual karena sudah terintegrasi antara *Google Drive*, *Google Doc*, *Google Form*, dan *Google Sheet* dengan bantuan formula dan *add-on AutoCrat*. Selain itu kampus STT-NF dapat menghemat biaya pengeluaran pembuatan, pengelolaan, dan pemeliharaan sistem.

Berdasarkan hasil evaluasi pengujian, sistem SKPI berbasis *Google Suite* sudah baik untuk dapat diterapkan di STT-NF, namun ada beberapa *feedback* yang direkomendasikan untuk diimplementasikan di penelitian selanjutnya. Adapun harapan peneliti terhadap sistem pengajuan administrasi SKPI ini dapat dikembangkan lebih lanjut, dan saran dari peneliti agar sistem ini diimplementasikan pada Sistem Informasi Akademik (AIS) di STT-NF.

DAFTAR PUSTAKA

- [1] Menteri Pendidikan, Kebudayaan, Riset, dan Teknologi, "Peraturan Nomor 6 Tahun 2022 Tentang Ijazah, Sertifikat Kompetensi, Sertifikat Profesi, Gelar, dan Kesetaraan Ijazah Perguruan Tinggi Negara Lain", 2022.
- [2] Menteri Pendidikan dan Kebudayaan, "Peraturan No. 3 Tahun 2020 Tentang Standar Nasional Pendidikan Tinggi.", 2020.
- [3] T. Marjan Fuadi, "Konsep Merdeka Belajar-Kampus Merdeka (MbkM): Aplikasinya Dalam Pendidikan Biologi," Pros. Semin. Nas. Biot., vol. 9, no. 2, p. 38, 2022, doi: 10.22373/pbio.v9i2.11594.
- [4] Z. Zulhalim, A. Z. Sianipar, and E. Witono, "Perancangan Aplikasi Surat Keterangan Pendamping Ijazah Berbasis Web Pada Stmik Jayakarta," *J. Inf. Syst. Informatics Comput.*, vol. 4, no. 2, p. 17, 2020, doi: 10.52362/jisicom.v4i2.316.
- [5] R.A.SYAEFUDIN and W. H. Suseno, "Penggunaan Pengaya (Add-on) Autocrat Pada Google Sheets Dalam Aplikasi Sering Cuti," *J. SAINTEKOM*, vol. 10, no.1, p.37, 2020, doi: 10.33020/saintekom.v10i1.129.
- [6] S. Wulandari, A. D. Lestari, O. Fajarianto, and P. K. Sari, "Optimalisasi Fitur AutoCrat Google Sheet Untuk Membuat E-Certificate Pada Sekolah",

- Abdimas Awang Long*, vol. 5, no. 2, p. 44, 2022, doi: 10.56301/awal.v5i1.438.
- [7] R.A.SYAEFUDIN and W. H. Suseno, “Penggunaan Pengaya (Add-on) Autocrat Pada Google Sheets Dalam Aplikasi Sering Cuti”, *J. SAINTEKOM*, vol. 10, no.1, p.37, 2020, doi: 10.33020/saintekom.v10i1.129.
- [8] D. Bagaskara, A. Lestari and F. Sylviana, “Perancangan Aplikasi Surat Keterangan Pendamping Ijazah (Skpi) Berbasis Website,” *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 16, no. 2, pp. 201–208, 2022, doi: 10.47111/jti.v16i2.5419.
- [9] K. Schwaber and J. Sutherland, “Panduan Scrum”, *Scrum Guide*, November 2020, [Online]. Available: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Indonesian.pdf>.

Jurnal Informatika Terpadu
Volume. 9 No. 2 Tahun 2023
Daftar Isi

Pengembangan Sistem Informasi Manajemen Keuangan Sekolah berbasis Web menggunakan <i>Whatsapp Gateway</i>	74
Nur Abidin, Anwar Aini, Muhammad Izzuddin	
UMMIBOT sebagai Media Layanan Informasi Penerimaan Mahasiswa Baru Universitas Muhammadiyah Sukabumi	82
M. Rizky Suherlan, Asriyanik, Agung Pambudi	
Pengembangan <i>REST API</i> dengan menggunakan <i>Express JS</i> untuk Mencari Mentor Pribadi	92
Nasrul, Ahmad Izhar	
Penyusunan Dokumentasi Sistem pada Perancangan Web <i>Backoffice</i> Aplikasi <i>Mobile Telemedicine</i>	103
Laily Maghfira, Esi Putri Silmina	
Rancang Bangun Sistem Informasi <i>Back Office</i> Aplikasi <i>Telemedicine</i> sebagai Media Pembelajaran	119
Anggit Widcaksono, Esi Putri Silmina	
Analisis Kriptografi Algoritma <i>Blowfish</i> pada Keamanan Data menggunakan <i>Dart</i>	126
Yoga Pratama, Tata Sutabri	
Analisa Kinerja Aplikasi Digital Forensik <i>Autopsy</i> untuk Pengembalian Data menggunakan Metode <i>NIST SP 800-86</i>	136
Dedek Julian, Tata Sutabri	
Perancangan Alat Tekanan Darah Otomatis secara Berkala menggunakan Sensor <i>MPX5050DP</i> untuk Pasien Hipertensi	143
Afdhal Kurniawan, Karli Eka Setiawan	
Pengelompokan Rumah Sakit di Jakarta Menggunakan Model <i>DBSCAN</i>, <i>Gaussian Mixture</i>, dan <i>Hierarchical Clustering</i>	149
Karli Eka Setiawan, Afdhal Kurniawan	
Pengembangan Prototipe Sistem S.K.P.I. untuk Pendamping Ijazah Berbasis <i>Google Suite</i>	157
Nuraini Febrianti, Hasna Alaurrahman, Amalia Rahmah	

Published by:

LPPM STT Terpadu Nurul Fikri

Jln. Raya Lenteng Agung, no. 20, Srengseng Sawah,
Jagakarsa, Jakarta Selatan, DKI Jakarta 12640

Telp. 021 - 786 3191 WhatsApp. 0851 7444 3360

Email : journal@nurulfikri.ac.id

Website : <https://journal.nurulfikri.ac.id/index.php/jit>

