MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE

SKRIPSI

DISUSUN OLEH

ALVIN LIE
NPM. 2209010168

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN
2026

MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom) dalam Program Studi Sistem Informasi pada Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara

ALVIN LIE
NPM. 2209010168

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN
2026

LEMBAR PENGESAHAN

Judul Skripsi	:	Modifikasi Keamanan Otentikasi One Time Password Menggunakan Algoritma Hmac-Sha256 Pada Sistem Informasi Pt. Indonesia Gadai Oke
Nama Mahasiswa	:	Alvin Lie
NPM	:	2209010168
Program Studi	:	Sistem Informasi

Menyetujui
Komisi Pembimbing

(Martiano, S.Kom.,M.Kom.)
NIDN. 0128029302

Ketua Program Studi

(Mahardika Abdi Prawira Tanjung, S.Kom.,M.Kom.)
NIDN. 0117088902

Dekan

(Dr. Al-Khowarizmi, S.Kom., M.Kom.)
NIDN. 0127099201

PERNYATAAN ORISINALITAS

MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 12 Februari 2026

Yang membuat pernyataan

Alvin Lie

NPM. 2209010168

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama	: Alvin Lie
NPM	: 2209010168
Program Studi	: Sistem Informasi
Karya Ilmiah	: Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif *(Non-Exclusive Royalty free Right)* atas penelitian skripsi saya yang berjudul:

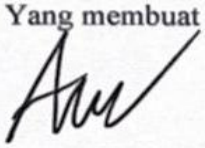
MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 12 Februari 2026

Yang membuat pernyataan



Alvin Lie

NPM. 2209010168

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap	: Alvin Lie
Tempat dan Tanggal Lahir	: Tanjung Balai, 03 Maret 2004
Alamat Rumah	: Jl. Pungguk Perum. Calista City No.11
Telepon/Faks/HP	: 082168136725
E-mail	: alvinlie1704@gmail.com

DATA PENDIDIKAN

SD	: SDs Sultan Iskandar Muda	TAMAT: 2016
SMP	: SMPs Sultan Iskandar Muda	TAMAT: 2019
SMA	: Man Insan Cendekia Aceh Timur	TAMAT: 2022

KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh Alhamdulillahirabbil'alamin.

Puji syukur penulis panjatkan kehadirat Allah SWT, Tuhan Yang Maha Esa karena atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik. Shalawat berangkaikan salam penulis hadiahkan kepada junjungan umat muslim nabi besar Muhammad SAW, semoga kita mendapatkan syafaatnya di hari kemudian kelak. Aamiin.aamiin yaa rabbal 'alamiin.

Tugas akhir ini berjudul "MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE" yang mana penelitian ini merupakan salah satu persyaratan untuk menyelesaikan jenjang strata satu (S1) bagi mahasiswa Program Studi Sistem Informasi Fakultas Ilmu Komputer Dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara, Medan.

Penulis menyadari bahwa skripsi ini tidak lepas dari bantuan, bimbingan, serta dukungan dari berbagai pihak. Dengan segala kerendahan hati, penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada :

1. Bapak Prof. Dr. Akrim, M.Pd. selaku rektor Universitas Muhammdiyah Umatera Utara
2. Bapak Dr. Alkhowarizmi, M.Kom. selaku dekan fakultas ilmu komputer dan teknologi informasi

3. Bapak Mahardika Abdi Prawira Tanjung,S.Kom.,M.Kom. selaku ketua prodi sistem informasi
4. Bapak Martiano,S.Kom.,M.Kom. selaku dosen pembimbing tugas akhir yang telah berkontribusi membantu penulis dalam menyelesaikan tugas akhir. Serta memberikan ide, saran, kritik dan bimbingan kepada penulis dalam menyelesaikan tugas akhir
5. Bapak Dr. Suhesti Wira Dharma, M.A. selaku ayah, yang senantiasa menjadi teladan, penopang, dan sumber kekuatan melalui doa, nasihat, serta pengorbanan tanpa henti, sehingga penulis mampu bertahan dan terus melangkah dalam menyelesaikan perjalanan studi ini.Pintu surga penulis,
6. Ummi Dr. Mailin, M.A. selaku ibu, yang dengan penuh cinta, kesabaran, dan ketulusan selalu mendampingi penulis, memanjatkan doa di setiap langkah, serta memberikan kasih sayang yang tak pernah putus, sehingga penulis mampu melewati segala lelah, ragu, dan keterbatasan.
7. Raja Alfiansyah, S.Kom. selaku abang, Alwi Auvinen Juha selaku adik pertama, dan Sultan Haramain selaku adik kedua, yang senantiasa memberikan dukungan, semangat, perhatian, serta doa tulus, sehingga penulis selalu termotivasi untuk menyelesaikan studi dan karya ini dengan sebaik-baiknya.
8. Ibu Zuli Agustina Gultom,,M.Kom. selaku dosen pembimbing akademik penulis selama masa perkuliahan
9. Bapak/Ibu Dosen FIKTI UMSU yang telah dengan sabar dan ikhlas memeberikan ilmunya kepada penulis

10. Pegawai biro administrasi dan seluruh staff fakultas ilmu komputer dan teknologi informasi yang telah memudahkan proses adminitrasi dan perkuliahan penulis
11. Pemilik NPM 2209010194 yang sedang sama-sama berjuang untuk meraih gelar S.Kom. semoga dipermudah dan dilancarkan segala urusan kedepannya.
12. Teman-Teman seperjuangan angkatan 2022 FIKTI UMSU

Medan, 10 Mei 2026

Alvin Lie

NPM. 2209010168

MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE

ABSTRAK

Keamanan data pengguna pada sistem informasi berbasis web seringkali terancam oleh kelemahan mekanisme otentikasi konvensional yang hanya mengandalkan kata sandi. Serangan seperti pencurian kredensial (*credential theft*) dan *Brute Force* menuntut adanya lapisan keamanan tambahan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem keamanan *Two-Factor Authentication* (2FA) menggunakan algoritma HMAC-SHA256 (*Hash-based Message Authentication Code with Secure Hash Algorithm 256-bit*) yang terintegrasi dengan fitur *Trusted Device* dan notifikasi *Web Push*. Metode yang diterapkan adalah *Time-based One-Time Password* (TOTP) dengan interval waktu 30 detik, di mana kode unik dibangkitkan melalui proses *dynamic truncation* 32-bit dari hasil enkripsi SHA-256. Pengujian sistem dilakukan menggunakan metode *Black Box Testing* dan *Security Testing* dengan skenario pengulangan sebanyak 10 kali percobaan. Hasil penelitian menunjukkan bahwa sistem memiliki tingkat keberhasilan fungsional sebesar 100% dalam memvalidasi pengguna yang sah. Selain itu, sistem terbukti efektif memitigasi ancaman keamanan dengan tingkat keberhasilan 100% dalam menolak serangan *SQL Injection*, *Cross-Site Scripting* (XSS), dan *Replay Attack* melalui mekanisme validasi token satu kali pakai. Implementasi notifikasi *Web Push* juga terbukti memberikan alternatif pengiriman kode OTP yang lebih efisien dan aman dibandingkan SMS konvensional.

Kata Kunci: *Two-Factor Authentication*; HMAC-SHA256; *Web Push Notification*; *Trusted Device*; Keamanan Web.

MODIFICATION OF ONE-TIME PASSWORD AUTHENTICATION SECURITY USING HMAC-SHA256 ALGORITHM IN THE INFORMATION SYSTEM OF PT. INDONESIA GADAI OKE

ABSTRACT

User data security in web-based information systems is frequently compromised by the vulnerabilities of conventional authentication mechanisms that rely solely on passwords. Attacks such as credential theft and Brute Force necessitate an additional layer of security. This study aims to design and implement a Two-Factor Authentication (2FA) security system using the HMAC-SHA256 algorithm (Hash-based Message Authentication Code with Secure Hash Algorithm 256-bit) integrated with Trusted Device features and Web Push notifications. The method applied is Time-based One-Time Password (TOTP) with a 30-second interval, where a unique code is generated through a 32-bit dynamic truncation process from the SHA-256 encryption output. System testing was conducted using Black Box Testing and Security Testing methods with a repetition scenario of 10 iterations. The results indicate that the system achieved a 100% functional success rate in validating legitimate users. Furthermore, the system proved effective in mitigating security threats with a 100% success rate in rejecting SQL Injection, Cross-Site Scripting (XSS), and Replay Attacks through a one-time token validation mechanism. The implementation of Web Push notifications also demonstrated a more efficient and secure alternative for OTP delivery compared to conventional SMS.

Keywords: Two-Factor Authentication; HMAC-SHA256; Web Push Notification; Trusted Device; Web Security.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR RUMUS

BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi informasi yang sangat pesat telah mendorong berbagai organisasi, termasuk perusahaan jasa keuangan, untuk melakukan digitalisasi proses bisnis. Sistem informasi tidak lagi hanya berperan sebagai pendukung operasional, tetapi telah menjadi komponen inti dalam pengelolaan, penyimpanan, serta distribusi data yang bersifat strategis dan sensitif. Kondisi ini menjadikan aspek keamanan informasi sebagai kebutuhan yang sangat penting, khususnya dalam menjaga kerahasiaan, integritas, dan ketersediaan data dari berbagai ancaman kejahatan siber(Martiano & Sary, 2022).

Salah satu celah keamanan yang sering dimanfaatkan oleh pelaku kejahatan siber adalah mekanisme otentikasi pengguna. Penggunaan metode autentikasi konvensional seperti username dan password statis dinilai sudah tidak lagi memadai karena rentan terhadap berbagai serangan, seperti phishing, brute force, keylogger, serta pencurian kredensial (Ma et al., 2021). Ketika informasi login berhasil diperoleh oleh pihak yang tidak berwenang, sistem dapat diakses secara ilegal tanpa terdeteksi, sehingga berpotensi menimbulkan kerugian finansial, gangguan operasional, serta menurunnya kepercayaan pengguna terhadap sistem.

Sebagai upaya meningkatkan keamanan otentikasi, banyak sistem informasi mulai menerapkan mekanisme otentikasi berlapis (multi-factor authentication), salah satunya melalui penggunaan One Time Password (OTP)(Mayorga & Yoo, 2025). OTP merupakan kata sandi dinamis yang hanya berlaku untuk satu kali penggunaan dalam jangka waktu tertentu. Dengan karakteristik tersebut, OTP

mampu mengurangi risiko penyalahgunaan kredensial, karena kode yang telah digunakan atau telah kedaluwarsa tidak dapat digunakan kembali oleh pihak lain.

Namun demikian, implementasi OTP konvensional masih menyisakan celah keamanan, baik dari sisi algoritma pembangkit maupun media distribusinya. Penelitian terdahulu menunjukkan bahwa OTP yang dibangkitkan dengan bilangan acak sederhana tanpa proteksi hash kriptografis rentan terhadap serangan prediksi (Kim et al., 2020). Selain itu, kelemahan fatal juga ditemukan pada metode distribusi kode OTP yang umum digunakan, seperti melalui Email, WhatsApp, atau SMS. Mekanisme OTP pada ekosistem seluler memiliki kerentanan signifikan terhadap manipulasi dan intersepsi jika tidak dilindungi dengan enkripsi yang memadai. Risiko ini diperparah dengan masalah latency (keterlambatan) yang tinggi akibat ketergantungan pada jaringan pihak ketiga, serta rentan terhadap serangan Man-in-the-Middle (MitM) dan penyadapan (sniffing). Penggunaan media pihak ketiga ini juga memunculkan isu privasi data dan biaya operasional yang berkelanjutan bagi perusahaan(Ma et al., 2021).

Menanggapi permasalahan tersebut, penulis berpendapat bahwa ketergantungan pada infrastruktur pihak ketiga (seperti provider seluler atau server email) merupakan titik lemah yang harus dieliminasi. Sistem otentikasi yang ideal seharusnya mampu beroperasi secara mandiri dengan memanfaatkan fitur bawaan peramban (browser) melalui teknologi Web Push Notification. Pendekatan ini dinilai mampu mengatasi dua masalah sekaligus: menghilangkan biaya operasional pengiriman pesan (SMS) dan meminimalisir latensi pengiriman kode, sehingga proses otentikasi menjadi lebih cepat dan efisien tanpa mengorbankan aspek keamanan.

Penelitian lain membuktikan bahwa penerapan algoritma HMAC-SHA256 pada sistem otentikasi mampu menjawab kelemahan dari sisi pembangkitan kode (Martiano & Sary, 2022). Algoritma HMAC-SHA256 menghasilkan nilai hash yang bergantung pada kunci rahasia dan pesan input, sehingga sulit dipalsukan. Algoritma ini sangat tepat jika dikombinasikan dengan mekanisme distribusi Web Push yang diusulkan penulis untuk menciptakan sistem keamanan yang tangguh namun tetap ringan secara komputasi.

Dalam konteks perusahaan jasa keuangan, PT. Indonesia Gadai Oke mengelola berbagai data sensitif, seperti data nasabah, data jaminan, serta transaksi keuangan. Seiring meningkatnya jumlah pengguna dan intensitas transaksi melalui sistem informasi, risiko terhadap serangan siber juga semakin besar. Berdasarkan pengamatan penulis di lapangan, efisiensi waktu saat proses *login* sangat mempengaruhi produktivitas kerja pegawai. Keterlambatan penerimaan kode OTP via SMS seringkali menghambat akses cepat ke data nasabah yang dibutuhkan. Oleh karena itu, apabila sistem otentikasi masih mengandalkan metode lama yang lambat dan berbayar, hal ini tidak hanya menjadi ancaman keamanan tetapi juga hambatan operasional yang serius. Apabila sistem otentikasi masih mengandalkan metode pengiriman yang rentan disadap atau mengalami keterlambatan, maka potensi terjadinya akses tidak sah dan hambatan operasional menjadi ancaman serius bagi perusahaan (Asyura Binti Sofian et al., 2024).

Berdasarkan permasalahan pada kelemahan distribusi konvensional dan kebutuhan akan algoritma yang kuat, diperlukan adanya modifikasi mekanisme keamanan otentikasi. Algoritma HMAC-SHA256 dipilih karena mampu memberikan jaminan integritas dan keaslian kode melalui fungsi hash yang kuat,

serta memungkinkan penerapan verifikasi yang tidak bergantung pada pengiriman pesan eksternal (Email/WhatsApp) yang lambat (Mayorga & Yoo, 2025). Hal ini diharapkan dapat meningkatkan kinerja sistem dari sisi kecepatan otentikasi.

Oleh karena itu, penelitian ini berjudul "Modifikasi Keamanan Otentikasi One Time Password Menggunakan Algoritma HMAC-SHA256 pada Sistem Informasi PT. Indonesia Gadai Oke". Penelitian ini dilakukan untuk menganalisis penerapan algoritma HMAC-SHA256 serta pengaruhnya terhadap peningkatan keamanan dan kinerja sistem dibandingkan metode sebelumnya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka permasalahan yang ingin dikaji dalam penelitian ini adalah:

1. Bagaimana mekanisme otentikasi yang berjalan saat ini pada sistem informasi PT. Indonesia Gadai Oke dan apa saja kelemahan yang dimilikinya terkait aspek keamanan?
2. Bagaimana cara mengimplementasikan algoritma HMAC-SHA256 dalam pembangkitan One Time Password (OTP) pada sistem informasi PT. Indonesia Gadai Oke?
3. Bagaimana kinerja algoritma HMAC-SHA256 dalam pembangkitan OTP ditinjau dari aspek kecepatan pemrosesan, efisiensi penggunaan sumber daya, dan keandalan (reliability) pada sistem informasi PT. Indonesia Gadai Oke?

1.3 Batasan Masalah

Agar pembahasan tidak melebar, penelitian ini dibatasi pada beberapa hal, yaitu:

1. Penelitian ini hanya berfokus pada implementasi kinerja algoritma HMAC-SHA256 dalam pembangkitan One Time Password (OTP) pada sistem informasi PT. Indonesia Gadai Oke.
2. Objek penelitian dibatasi pada modul atau fitur otentikasi pengguna dalam sistem informasi, tidak mencakup seluruh fungsi bisnis lain seperti transaksi, pelaporan, maupun integrasi dengan sistem eksternal.
3. Aspek kinerja yang dianalisis dibatasi pada kecepatan pemrosesan (time execution), efisiensi penggunaan sumber daya (resource usage), dan tingkat keberhasilan proses otentikasi (reliability), tanpa melakukan pengujian mendalam terhadap skalabilitas pada skala pengguna yang sangat besar.
4. Penelitian ini tidak membahas secara rinci aspek kebijakan keamanan organisasi, prosedur operasional, maupun faktor manusia (human error), dan hanya meninjau dari sisi teknis implementasi algoritma pada sistem informasi.
5. Implementasi yang dilakukan bersifat studi kasus pada lingkungan sistem informasi PT. Indonesia Gadai Oke dan tidak dimaksudkan untuk menggeneralisasi seluruh jenis sistem informasi pada perusahaan jasa keuangan lainnya.

1.4 Tujuan Penelitian

Penelitian ini memiliki beberapa tujuan, yaitu:

1. Mengetahui mekanisme otentikasi yang saat ini digunakan pada sistem informasi PT. Indonesia Gadai Oke serta mengidentifikasi kelemahannya dari sisi keamanan.
2. Merancang dan mengimplementasikan mekanisme One Time Password (OTP) dengan menggunakan algoritma HMAC-SHA256 pada modul otentikasi sistem informasi PT. Indonesia Gadai Oke.
3. Merancang antarmuka pengguna (*User Interface*) sistem otentikasi yang meliputi halaman verifikasi dan tampilan notifikasi *Web Push*, untuk memastikan kemudahan interaksi pengguna dalam menerima dan memasukkan kode OTP.
4. Mengukur kinerja algoritma HMAC-SHA256 dalam pembangkitan OTP ditinjau dari aspek kecepatan pemrosesan, efisiensi penggunaan sumber daya, dan tingkat keberhasilan proses otentikasi.
5. Mengevaluasi peningkatan tingkat keamanan otentikasi pengguna pada sistem informasi PT. Indonesia Gadai Oke setelah diterapkannya mekanisme OTP berbasis HMAC-SHA256.

1.5 Manfaat Penelitian

Penelitian ini diharapkan tidak hanya memberikan solusi terhadap permasalahan keamanan otentikasi pada sistem informasi PT. Indonesia Gadai Oke, tetapi juga memberikan kontribusi bagi pengembangan ilmu pengetahuan dan praktik di bidang keamanan sistem informasi. Secara umum, hasil penelitian ini diharapkan dapat menjadi referensi bagi pihak akademisi, praktisi, maupun instansi terkait yang ingin mengimplementasikan mekanisme One Time Password (OTP) dengan algoritma kriptografi yang kuat.

Berdasarkan ruang lingkup dan tujuan yang telah dirumuskan, manfaat penelitian ini dapat dibagi menjadi dua, yaitu manfaat teoritis dan manfaat praktis sebagai berikut:

1. Memberikan kontribusi pada pengembangan ilmu pengetahuan di bidang keamanan sistem informasi, khususnya terkait implementasi algoritma HMAC-SHA256 dalam mekanisme otentikasi One Time Password (OTP).
2. Menambah referensi kajian ilmiah mengenai analisis kinerja algoritma hash kriptografis dalam proses otentikasi pengguna pada suatu sistem informasi.
3. Menjadi acuan bagi peneliti lain yang ingin melakukan penelitian lanjutan tentang penerapan algoritma kriptografi, OTP, atau mekanisme autentikasi berlapis (multi-factor authentication) di berbagai jenis sistem informasi.

Secara praktis, penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. Bagi PT. Indonesia Gadai Oke
   Memberikan rekomendasi teknis untuk meningkatkan keamanan proses otentikasi pengguna melalui penerapan OTP berbasis HMAC-SHA256 sehingga dapat meminimalkan risiko akses tidak sah dan kebocoran data.
2. Bagi divisi teknologi informasi (TI) / pengembang sistem
   Menjadi panduan dalam merancang, mengimplementasikan, dan mengevaluasi mekanisme OTP yang aman dan efisien, serta dapat dijadikan dasar pengembangan lebih lanjut terhadap modul keamanan sistem informasi yang digunakan perusahaan.
3. Bagi pengguna atau karyawan PT. Indonesia Gadai Oke
   Memberikan tingkat keamanan yang lebih baik dalam proses login dan

akses sistem, sehingga meningkatkan rasa aman dan kepercayaan dalam menggunakan sistem informasi perusahaan.

4. Bagi perguruan tinggi dan civitas akademika
   Menambah koleksi referensi ilmiah berupa studi kasus penerapan algoritma HMAC-SHA256 pada OTP, yang dapat dimanfaatkan dalam kegiatan pembelajaran, penelitian, dan pengembangan materi kuliah terkait kriptografi dan keamanan sistem informasi.

BAB II
LANDASAN TEORI

2.1 Keamanan Informasi

Sistem informasi telah menjadi tulang punggung bagi operasional organisasi modern, mengintegrasikan perangkat keras, perangkat lunak, data, prosedur, dan manusia untuk menghasilkan informasi yang berguna bagi pengambilan keputusan.(Laudon & Laudon, 2020), sistem informasi adalah sekumpulan komponen yang saling terkait yang bekerja sama untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi guna mendukung pengambilan keputusan, koordinasi, dan pengendalian dalam suatu organisasi. Dalam konteks perusahaan jasa keuangan seperti PT. Indonesia Gadai Oke, sistem informasi tidak hanya berfungsi sebagai alat pencatatan transaksi, tetapi juga sebagai aset strategis yang menyimpan data sensitif nasabah dan rekam jejak finansial yang harus dilindungi. Ketergantungan yang tinggi terhadap teknologi ini menuntut adanya mekanisme perlindungan yang kuat untuk menjamin keberlangsungan bisnis.

Keamanan informasi merupakan aspek krusial yang tidak dapat dipisahkan dari operasional sistem informasi. Whitman dan Mattord (2021) mendefinisikan keamanan informasi sebagai perlindungan terhadap informasi dan elemen-elemen penting di dalamnya, termasuk sistem dan perangkat keras yang digunakan untuk menyimpan, memproses, dan mentransmisikannya. Tujuan utama keamanan informasi sering dirangkum dalam model CIA Triad, yaitu Confidentiality (kerahasiaan), Integrity (integritas), dan Availability (ketersediaan). Kerahasiaan menjamin bahwa informasi hanya dapat diakses oleh pihak yang berwenang;

integritas memastikan bahwa data tetap akurat dan tidak dimodifikasi oleh pihak yang tidak sah; sedangkan ketersediaan menjamin bahwa informasi dapat diakses oleh pengguna yang berhak kapanpun dibutuhkan.

Ancaman terhadap keamanan informasi terus berkembang seiring dengan kemajuan teknologi, mulai dari serangan siber sederhana hingga ancaman persisten tingkat lanjut (Advanced Persistent Threats). Stallings (2017) menjelaskan bahwa ancaman keamanan komputer dapat dikategorikan menjadi serangan pasif, seperti penyadapan lalu lintas data, dan serangan aktif, seperti manipulasi data, penyamaran (masquerade), dan penolakan layanan (denial of service). Dalam sektor keuangan, serangan terhadap kredensial pengguna menjadi salah satu vektor serangan yang paling umum, di mana penyerang berupaya mengambil alih akun pengguna untuk melakukan transaksi ilegal. Oleh karena itu, penerapan kontrol keamanan yang ketat, khususnya pada mekanisme akses, menjadi prioritas utama.

Salah satu pilar utama dalam strategi pertahanan keamanan informasi adalah kontrol akses, yang berfungsi membatasi siapa saja yang dapat berinteraksi dengan sumber daya sistem. Menurut Pfleeger dan Pfleeger (2015), kontrol akses mencakup tiga langkah fundamental: identifikasi, otentikasi, dan otorisasi. Identifikasi adalah proses mengenali identitas pengguna (misalnya melalui username), otentikasi adalah proses memverifikasi kebenaran identitas tersebut (misalnya melalui password), dan otorisasi adalah pemberian hak akses sesuai dengan peran pengguna. Kelemahan pada salah satu tahap ini dapat meruntuhkan seluruh arsitektur keamanan sistem, sehingga diperlukan mekanisme yang mampu memvalidasi identitas pengguna dengan tingkat kepercayaan yang tinggi.

Dalam implementasinya, keamanan informasi tidak hanya berbicara masalah teknis, tetapi juga tata kelola dan manajemen risiko. Sarno dan Iffano (2020) menekankan bahwa manajemen keamanan informasi harus dilakukan secara holistik, mencakup kebijakan, prosedur, dan teknologi. Kegagalan dalam mengamankan sistem informasi dapat berakibat fatal, mulai dari kerugian finansial langsung, tuntutan hukum, hingga hilangnya reputasi perusahaan di mata nasabah. Mengingat PT. Indonesia Gadai Oke mengelola dana dan jaminan masyarakat, penerapan standar keamanan informasi yang ketat bukan lagi sebuah pilihan, melainkan kewajiban untuk menjaga kepercayaan publik dan mematuhi regulasi industri keuangan yang berlaku.

2.2 Konsep Dasar Otentikasi

Otentikasi merupakan garis pertahanan pertama dalam mengamankan akses ke sistem informasi. Menurut Schneier (2015), otentikasi adalah proses pembuktian identitas seseorang atau sesuatu, yang bertujuan memastikan bahwa pihak yang meminta akses benar-benar sesuai dengan klaim identitas yang diajukan. Secara umum, faktor otentikasi dibagi menjadi tiga kategori utama: something you know (sesuatu yang Anda ketahui, seperti password atau PIN), something you have (sesuatu yang Anda miliki, seperti kartu pintar atau token), dan something you are (sesuatu yang melekat pada diri Anda, seperti sidik jari atau biometrik lainnya). Penggunaan kombinasi dari faktor-faktor ini dikenal sebagai Multi-Factor Authentication (MFA), yang terbukti jauh lebih aman dibandingkan metode faktor tunggal.

Metode otentikasi yang paling umum digunakan adalah otentikasi berbasis pengetahuan, yaitu penggunaan password. Namun, NIST (National Institute of Standards and Technology) (2017) dalam pedoman identitas digitalnya menyatakan bahwa ketergantungan pada password statis semata memiliki risiko keamanan yang tinggi. Password rentan terhadap serangan brute force (menebak semua kemungkinan kombinasi), dictionary attack (menggunakan daftar kata umum), serta rekayasa sosial (social engineering) seperti phishing. Selain itu, kebiasaan pengguna yang sering menggunakan password yang sama untuk berbagai layanan atau memilih kombinasi yang mudah ditebak semakin memperburuk kerentanan sistem yang hanya mengandalkan Single-Factor Authentication.

Untuk mengatasi kelemahan password statis, dikembangkanlah konsep otentikasi dinamis yang nilainya berubah setiap kali digunakan. Menezes et al. (2018) menjelaskan bahwa otentikasi dinamis atau challenge-response dirancang untuk mencegah serangan replay attack, di mana penyerang merekam data otentikasi yang valid dan menggunakannya kembali di kemudian hari. Dalam skema ini, server memberikan "tantangan" unik, dan pengguna (atau perangkat pengguna) harus memberikan "respon" yang benar berdasarkan tantangan tersebut dan rahasia yang dimiliki bersama. Konsep inilah yang menjadi dasar pengembangan teknologi One Time Password (OTP), di mana kode sandi hanya valid untuk satu kali login atau satu sesi transaksi.

Pentingnya otentikasi yang kuat dalam sistem keuangan juga ditegaskan dalam standar keamanan industri kartu pembayaran atau PCI-DSS (Payment Card Industry Data Security Standard). PCI Security Standards Council (2022) mewajibkan penggunaan otentikasi multi-faktor untuk semua akses non-konsol ke

lingkungan data pemegang kartu, serta untuk semua akses jarak jauh oleh karyawan dan pihak ketiga. Hal ini menunjukkan bahwa dalam lingkungan yang memproses data sensitif, validasi identitas tidak boleh hanya didasarkan pada asumsi, melainkan harus dibuktikan melalui mekanisme kriptografis yang handal untuk meminimalkan risiko akses tidak sah.

Keandalan sistem otentikasi juga sangat dipengaruhi oleh proses manajemen identitas (Identity Management). Bishop (2019) menyatakan bahwa sistem otentikasi yang baik harus mampu mengelola siklus hidup identitas pengguna, mulai dari pendaftaran (enrollment), penyediaan kredensial, pembaruan, hingga pencabutan akses (revocation). Jika proses pendaftaran lemah atau pencabutan akses terlambat dilakukan, maka teknologi otentikasi secanggih apapun tidak akan efektif. Oleh karena itu, implementasi otentikasi di PT. Indonesia Gadai Oke harus diintegrasikan dengan prosedur operasional yang ketat untuk memastikan bahwa hanya pengguna yang sah dan aktif yang dapat membangkitkan dan menggunakan token otentikasi.

2.3 One Time Password (OTP)

One Time Password (OTP) adalah sebuah metode keamanan yang menghasilkan string karakter numerik atau alfanumerik yang hanya dapat digunakan untuk satu kali proses otentikasi. Rouse (2019) mendefinisikan OTP sebagai kata sandi yang valid hanya untuk satu sesi login atau transaksi pada sistem komputer maupun perangkat digital lainnya. Keunggulan utama OTP dibandingkan kata sandi tradisional adalah sifatnya yang tidak rentan terhadap serangan replay attack. Jika seorang peretas berhasil menyadap kode OTP yang sedang dikirimkan atau digunakan, kode tersebut tidak akan berguna lagi untuk akses berikutnya

karena sistem akan menolaknya setelah digunakan satu kali atau setelah masa berlakunya habis.

Secara teknis, pembangkitan OTP didasarkan pada dua komponen utama: seed (benih/kunci rahasia) dan faktor pengubah (moving factor). RFC 4226 (2005) yang diterbitkan oleh Internet Engineering Task Force (IETF) menjelaskan bahwa ada dua standar utama algoritma OTP, yaitu HOTP (HMAC-based One-Time Password) dan TOTP (Time-based One-Time Password). Pada HOTP, faktor pengubahnya adalah counter (penghitung) yang bertambah setiap kali kode dibuat. Sedangkan pada TOTP, faktor pengubahnya adalah waktu saat ini. Kedua metode ini memastikan bahwa client (pengguna) dan server dapat menghasilkan kode yang sama secara independen tanpa perlu saling mengirimkan kunci rahasia setiap kali bertransaksi, asalkan keduanya memiliki seed awal yang sama.

Algoritma TOTP (Time-based One-Time Password), yang merupakan pengembangan dari HOTP, menjadi standar industri yang paling banyak diadopsi saat ini. M'Raihi et al. (2011) dalam RFC 6238 menjelaskan bahwa TOTP menggunakan interval waktu (biasanya 30 atau 60 detik) sebagai nilai counter. Hal ini membuat TOTP lebih praktis digunakan karena tidak memerlukan sinkronisasi event antara client dan server, melainkan hanya membutuhkan sinkronisasi jam sistem. Kelebihan ini sangat relevan untuk aplikasi mobile dan layanan web modern, di mana pengguna mengharapkan kemudahan akses tanpa mengurangi tingkat keamanan. Jika kode tidak digunakan dalam jendela waktu tersebut, maka kode akan kadaluarsa dan pengguna harus membangkitkan kode baru.

Implementasi OTP dalam sistem informasi menghadapi tantangan keseimbangan antara keamanan dan kenyamanan pengguna (usability). Gunson et al. (2011) dalam penelitiannya mengenai usability keamanan biometrik dan token, menyebutkan bahwa jika mekanisme keamanan terlalu rumit atau memakan waktu, pengguna cenderung mencari cara untuk memintasnya atau merasa frustrasi. Oleh karena itu, panjang digit OTP (biasanya 6 atau 8 digit) dan durasi validitasnya harus dirancang sedemikian rupa agar mudah diingat dan dimasukkan oleh pengguna, namun tetap sulit ditebak oleh penyerang dalam waktu singkat. Algoritma pembangkit OTP harus mampu menghasilkan distribusi angka yang acak dan seragam agar pola angka tidak mudah diprediksi.

Selain algoritma pembangkitan, metode pengiriman OTP juga menjadi faktor penentu keamanan. Kurniawan (2020) menyoroti bahwa OTP dapat dikirimkan melalui SMS, email, atau dibangkitkan langsung melalui aplikasi (soft token) dan perangkat keras (hard token). Pengiriman via SMS kini mulai ditinggalkan oleh standar keamanan tinggi karena kerentanan terhadap serangan SIM Swap dan intersepsi jaringan seluler. Penggunaan algoritma berbasis HMAC-SHA256 yang ditanamkan dalam aplikasi (app-based generator) dinilai lebih aman karena proses pembangkitan kode terjadi secara lokal di perangkat pengguna tanpa perlu transmisi kode melalui jaringan publik, sehingga secara signifikan mengurangi risiko penyadapan Man-in-the-Middle (MitM).

Penelitian terdahulu mengenai OTP telah dilakukan dalam berbagai konteks sistem informasi dan layanan digital. Shao dkk. mengusulkan kerangka kerja otentikasi pembayaran elektronik berbasis TOTP yang dikombinasikan dengan QR Code, dan menunjukkan peningkatan keamanan otentikasi serta otorisasi

dibandingkan metode konvensional pada skenario e-payment (Shao et al., 2020). Penelitian lain oleh Al-Sarrayrih dkk. mengembangkan skema two-factor authentication untuk layanan e-government dengan memanfaatkan OTP yang dibangkitkan melalui aplikasi khusus di smartphone, dan hasilnya menunjukkan solusi ini lebih aman dan andal dibandingkan pengiriman OTP via SMS yang rentan intersepsi (Al-Sarrayrih et al., 2018). Di tingkat sistem informasi web, Setiawan dkk. menerapkan kombinasi algoritma SHA-256 dan Time-based One-Time Password (TOTP) pada sistem login website, dan melaporkan bahwa pendekatan ini mampu menurunkan risiko pencurian password serta meningkatkan keamanan proses login pengguna secara signifikan (Setiawan et al., 2020). Selain itu, penelitian di lingkungan akademik juga mengkaji penerapan TOTP sebagai faktor tambahan dalam skema multi-factor authentication, misalnya pada sistem informasi rekam medis rumah sakit, yang terbukti dapat meningkatkan perlindungan terhadap akses tidak sah melalui pengiriman kode TOTP sebagai auth token tambahan (Rahmawati et al., 2024).

2.4 Kriptografi dan Fungsi Hash

Kriptografi adalah ilmu dan seni menjaga keamanan pesan. Katz dan Lindell (2020) mendefinisikan kriptografi modern sebagai studi tentang teknik matematika untuk mengamankan informasi digital, sistem, dan komputasi terdistribusi terhadap serangan musuh. Tujuan fundamental kriptografi meliputi kerahasiaan (confidentiality), integritas data (data integrity), otentikasi (authentication), dan nir-penyangkalan (non-repudiation). Dalam konteks pembangkitan OTP, kriptografi berperan vital dalam memastikan bahwa kode yang dihasilkan bersifat unik, tidak dapat diprediksi, dan tidak dapat dibalikkan untuk

mengetahui kunci rahasia aslinya. Tanpa dasar kriptografi yang kuat, mekanisme OTP hanyalah sekumpulan angka acak yang mudah dieksploitasi.

Salah satu primitif kriptografi yang paling penting adalah fungsi hash satu arah (one-way hash function). Stallings (2017) menjelaskan bahwa fungsi hash adalah algoritma yang memetakan data dengan ukuran sembarang (pesan) menjadi data dengan ukuran tetap (nilai hash atau digest). Sifat utama dari fungsi hash kriptografis yang aman adalah: (1) pre-image resistance (sulit mencari pesan asli dari nilai hash-nya), (2) second pre-image resistance (sulit mencari pesan lain yang memiliki nilai hash sama dengan pesan tertentu), dan (3) collision resistance (sulit menemukan dua pesan sembarang yang menghasilkan nilai hash yang sama). Sifat-sifat ini sangat penting untuk memastikan integritas data dan keamanan otentikasi.

Keluarga algoritma SHA-2 (Secure Hash Algorithm 2) merupakan standar fungsi hash yang direkomendasikan secara luas saat ini. NIST (2015) dalam FIPS PUB 180-4 menjelaskan bahwa SHA-2 terdiri dari beberapa varian, termasuk SHA-224, SHA-256, SHA-384, dan SHA-512. SHA-256, yang menghasilkan digest sepanjang 256 bit (32 byte), dianggap sebagai standar keseimbangan terbaik antara keamanan dan kinerja komputasi untuk banyak aplikasi. Dibandingkan dengan pendahulunya SHA-1 yang telah dinyatakan tidak aman karena ditemukannya kerentanan tabrakan (collision), SHA-256 menawarkan margin keamanan yang jauh lebih tinggi dan resisten terhadap serangan komputasi modern, termasuk serangan menggunakan perangkat keras khusus.

Fungsi hash memainkan peran sentral dalam skema HMAC (Hash-based Message Authentication Code). Krawczyk et al. (1997) dalam RFC 2104

mendefinisikan HMAC sebagai mekanisme untuk otentikasi pesan menggunakan fungsi hash kriptografis yang dikombinasikan dengan kunci rahasia. Berbeda dengan hash biasa yang hanya menjamin integritas (bahwa data tidak berubah), HMAC menjamin otentikasi asal dan integritas data sekaligus. Hanya pihak yang memiliki kunci rahasia yang sama yang dapat menghasilkan dan memverifikasi nilai HMAC yang valid. Inilah mengapa HMAC menjadi fondasi yang ideal untuk algoritma OTP, di mana validitas kode bergantung pada kepemilikan kunci rahasia (shared secret) antara pengguna dan sistem.

Penerapan fungsi hash dalam keamanan sistem juga harus memperhatikan aspek efisiensi komputasi. Paar dan Pelzl (2010) mencatat bahwa meskipun algoritma kriptografi harus aman, mereka juga harus praktis untuk diimplementasikan pada berbagai platform, mulai dari server berkinerja tinggi hingga perangkat mobile dengan sumber daya terbatas. SHA-256 dirancang menggunakan operasi logika bitwise sederhana (AND, OR, XOR, SHIFT, ROTATE) dan penjumlahan modulo 32-bit, yang dapat dieksekusi dengan sangat cepat pada prosesor modern 32-bit maupun 64-bit. Karakteristik ini membuat SHA-256 sangat cocok digunakan dalam pembangkitan OTP secara real-time tanpa membebani kinerja sistem secara keseluruhan.

2.5 Algoritma HMAC-SHA256

HMAC-SHA256 adalah spesifikasi teknis dari konstruksi HMAC yang menggunakan SHA-256 sebagai fungsi hash yang mendasarinya. Menurut Pratama (2021), HMAC-SHA256 menggabungkan keunggulan struktur HMAC yang terbukti aman secara matematis dengan ketangguhan algoritma hash SHA-256. Proses kerja algoritma ini melibatkan penggabungan kunci rahasia (secret key)

dengan pesan input (dalam kasus OTP, pesannya adalah nilai counter atau waktu), kemudian melakukan hashing berulang dengan padding tertentu (inner pad dan outer pad). Hasil akhirnya adalah digest 256-bit yang unik dan sangat sensitif terhadap perubahan sekecil apapun pada input pesan maupun kunci rahasia.

Keamanan HMAC-SHA256 bersumber dari ketahanan fungsi hash SHA-256 terhadap serangan kriptoanalisis. Dang (2012) dalam publikasi NIST menyatakan bahwa untuk memalsukan nilai HMAC-SHA256 tanpa mengetahui kuncinya, seorang penyerang harus memecahkan fungsi hash SHA-256 itu sendiri, yang secara komputasi saat ini dianggap tidak mungkin dilakukan (infeasible) dengan teknologi yang ada. Selain itu, konstruksi HMAC dirancang untuk kebal terhadap length extension attack, sebuah kelemahan yang sering ditemukan pada implementasi hash sederhana yang hanya menempelkan kunci pada pesan (secret prefix atau secret suffix). Hal ini menjadikan HMAC-SHA256 standar emas dalam otentikasi API (Application Programming Interface) dan protokol keamanan jaringan seperti IPsec dan TLS.

Dalam konteks pembangkitan OTP, output 256-bit dari HMAC-SHA256 perlu diproses lebih lanjut agar menjadi kode numerik yang mudah dibaca manusia. Proses ini disebut sebagai Dynamic Truncation. RFC 6238 (2011) menjelaskan bahwa algoritma TOTP mengambil 4 bit terakhir dari hasil HMAC sebagai offset, kemudian mengambil 4 byte data mulai dari posisi offset tersebut untuk membentuk bilangan integer 31-bit. Bilangan ini kemudian dioperasikan dengan modulus 10^d (di mana d adalah jumlah digit OTP yang diinginkan, biasanya 6 atau 8) untuk menghasilkan kode akhir. Meskipun output dipangkas secara signifikan,

keamanan kriptografis tetap terjaga karena ketidaktahuan penyerang akan hasil hash penuh dan kunci rahasia yang digunakan.

Kinerja HMAC-SHA256 juga telah menjadi subjek berbagai penelitian performa. Saputra dan Wibowo (2019) dalam penelitian mereka membandingkan kinerja HMAC-MD5, HMAC-SHA1, dan HMAC-SHA256. Hasilnya menunjukkan bahwa meskipun HMAC-SHA256 membutuhkan waktu komputasi sedikit lebih lama dibandingkan pendahulunya karena kompleksitas operasinya, perbedaannya berada dalam orde milidetik dan hampir tidak terasa oleh pengguna akhir (user-perceptible latency). Dengan meningkatnya kemampuan prosesor modern, overhead komputasi SHA-256 menjadi dapat diabaikan jika dibandingkan dengan peningkatan keamanan eksponensial yang ditawarkannya. Ini menjadikan HMAC-SHA256 pilihan yang optimal untuk sistem PT. Indonesia Gadai Oke yang membutuhkan keamanan tinggi tanpa mengorbankan pengalaman pengguna.

Implementasi HMAC-SHA256 dalam sistem informasi memerlukan penanganan kunci (key management) yang aman. Barker (2016) menekankan bahwa keamanan algoritma simetris seperti HMAC sepenuhnya bergantung pada kerahasiaan kuncinya. Kunci rahasia harus dibangkitkan menggunakan Random Number Generator (RNG) yang aman secara kriptografis, disimpan dalam bentuk terenkripsi di database server, dan didistribusikan ke aplikasi pengguna melalui saluran yang aman (misalnya kode QR saat inisialisasi). Jika manajemen kunci dilakukan dengan buruk, maka kekuatan algoritma HMAC-SHA256 akan sia-sia. Oleh karena itu, penelitian implementasi ini tidak hanya berfokus pada kode algoritma, tetapi juga pada ekosistem keamanan di sekitarnya.

2.6 Teknologi Distribusi Notifikasi

Selain algoritma pembangkitan kode yang aman, mekanisme distribusi *One Time Password* (OTP) memegang peranan yang sangat vital dalam arsitektur keamanan sistem otentikasi dua faktor. Distribusi notifikasi berfungsi sebagai jembatan penghantar yang memastikan kode rahasia sampai ke tangan pengguna yang sah dalam waktu sesingkat mungkin. Tanpa adanya jalur distribusi yang cepat dan reliabel, algoritma enkripsi yang paling rumit sekalipun akan menjadi tidak efektif, karena keterlambatan penerimaan kode dapat menyebabkan token kadaluwarsa sebelum sempat digunakan, yang pada akhirnya menghambat produktivitas pengguna.

Selama satu dekade terakhir, metode distribusi konvensional seperti *Short Message Service* (SMS) dan surat elektronik (*Email*) menjadi standar industri. Namun, kedua metode ini memiliki kelemahan mendasar yang signifikan. Penggunaan SMS sangat bergantung pada kualitas sinyal jaringan seluler (GSM) dan seringkali mengalami kendala *latency* atau keterlambatan pengiriman saat jaringan operator sedang padat. Selain itu, SMS membebankan biaya per pesan (*cost per message*) yang jika diakumulasikan dalam skala perusahaan dengan ribuan transaksi login per hari, akan menciptakan beban operasional yang cukup besar bagi organisasi.

Di sisi lain, distribusi melalui *Email* seringkali tidak dianggap sebagai saluran komunikasi instan (*real-time*). Tantangan utama pada penggunaan email adalah risiko pesan masuk ke dalam folder *Spam* atau *Junk*, yang menyebabkan pengguna kesulitan menemukan kode OTP mereka. Selain itu, protokol pengiriman email juga memiliki jeda waktu sinkronisasi antara server dan aplikasi klien di perangkat pengguna, sehingga tidak dapat menjamin kecepatan penerimaan di bawah hitungan detik yang sangat dibutuhkan dalam proses login yang responsif.

Untuk mengatasi berbagai keterbatasan tersebut, penelitian ini menerapkan teknologi distribusi berbasis *Web Push Notification*. Teknologi ini merupakan standar modern yang memungkinkan aplikasi web untuk mengirimkan pesan notifikasi ke peramban (*browser*) pengguna layaknya aplikasi *native* di ponsel pintar. Berbeda dengan SMS yang berjalan di atas jaringan seluler, *Web Push* bekerja sepenuhnya menggunakan koneksi internet data. Teknologi ini memanfaatkan *Push API* dan standar keamanan web untuk membuat saluran komunikasi yang persisten antara server aplikasi dan perangkat pengguna tanpa memerlukan nomor telepon.

Secara teknis, keunggulan utama dari teknologi distribusi ini terletak pada penggunaan *Service Worker*, sebuah skrip yang berjalan di latar belakang peramban terlepas dari halaman web yang sedang dibuka. Hal ini memungkinkan notifikasi OTP tetap dapat diterima secara *real-time* bahkan ketika pengguna sedang tidak membuka tab aplikasi tersebut. Dengan beralih ke teknologi ini, sistem mampu mengeliminasi biaya pengiriman pesan sepenuhnya (biaya nol rupiah) serta memangkas latensi pengiriman secara drastis, sehingga menciptakan keseimbangan yang ideal antara keamanan yang ketat dan efisiensi biaya operasional perusahaan.

2.6.1 Web Push Notification

Web Push Notification adalah teknologi yang memungkinkan pengiriman pesan dari server ke peramban (*browser*) pengguna secara *real-time*, meskipun pengguna sedang tidak membuka halaman web tersebut. Menurut Gaunt (2019), teknologi ini menawarkan keunggulan signifikan karena memanfaatkan protokol standar web yang didukung oleh berbagai peramban modern. Pesan yang dikirimkan dienkripsi secara *end-to-end* menggunakan standar *Elliptic Curve*

Diffie-Hellman (ECDH), sehingga kerahasiaan kode OTP tetap terjaga dan aman dari penyadapan (*sniffing*) tanpa membebankan biaya operasional per pesan (*zero-cost*).

2.6.2 Service Worker

Komponen kunci untuk menjalankan Web Push Notification adalah Service Worker. Service Worker didefinisikan sebagai skrip yang berjalan di latar belakang (background) peramban, terpisah dari thread utama halaman web. Dalam penelitian ini, Service Worker bertugas "mendengarkan" sinyal OTP dari server dan memunculkannya sebagai notifikasi sistem (system tray notification) di perangkat pengguna secara instan, memastikan pengguna menerima kode verifikasi tanpa ketergantungan pada jaringan seluler (MDN Web Docs, 2023).

2.7 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa standar visualisasi, perancangan, dan pendokumentasian sistem perangkat lunak. Menurut Booch, Rumbaugh, dan Jacobson (2017), yang dikenal sebagai "The Three Amigos" pencetus UML, bahasa ini menyediakan seperangkat notasi grafis untuk menciptakan model abstrak dari sebuah sistem, yang disebut blueprint. Dalam pengembangan sistem keamanan seperti implementasi OTP, UML memegang peranan vital untuk menjembatani komunikasi antara pengembang sistem, analis keamanan, dan pemangku kepentingan bisnis, memastikan bahwa logika keamanan yang kompleks dapat dipahami secara visual sebelum diterjemahkan ke dalam kode program.

UML dalam rekayasa perangkat lunak modern tidak terbatas pada dokumentasi pasca-pengembangan, melainkan sebagai alat bantu analisis dan desain. Pressman dan Maxim (2020) menjelaskan bahwa pemodelan dengan UML membantu pengembang untuk mendeteksi potensi kesalahan logika dan celah keamanan pada tahap awal desain (design phase). Hal ini sangat krusial dalam perancangan protokol otentikasi, di mana urutan pertukaran pesan antara client dan server harus presisi. Kesalahan dalam urutan logika verifikasi OTP dapat mengakibatkan kerentanan fatal, sehingga pemodelan visual menjadi langkah preventif yang wajib dilakukan.

Salah satu diagram UML yang paling fundamental adalah Use Case Diagram. Menurut Whitten dan Bentley (2007), diagram ini menggambarkan interaksi fungsional antara pengguna (actor) dengan sistem, serta mendefinisikan batasan sistem (system boundary). Dalam konteks penelitian ini, Use Case Diagram akan memetakan siapa saja yang berhak meminta OTP, bagaimana proses login dilakukan, dan bagaimana sistem merespons jika otentikasi gagal. Diagram ini memberikan gambaran high-level mengenai fitur keamanan yang akan dibangun tanpa masuk ke detail teknis algoritma.

Untuk menggambarkan logika prosedural dari pembangkitan dan validasi OTP, digunakan Activity Diagram. Fowler (2018) mendefinisikan Activity Diagram sebagai teknik untuk menggambarkan logika prosedural, proses bisnis, dan aliran kerja (workflow). Diagram ini sangat relevan untuk memodelkan algoritma HMAC-SHA256, mulai dari pengambilan secret key, pembacaan waktu/counter, proses hashing, hingga pemotongan digit (truncation). Dengan diagram ini, alur keputusan (decision nodes) seperti pengecekan masa berlaku kode

OTP atau pencocokan kode input dengan kode sistem dapat digambarkan secara runtut dan logis.

Aspek terpenting dalam pemodelan protokol keamanan adalah Sequence Diagram. Sommerville (2016) menjelaskan bahwa diagram ini menunjukkan interaksi antar objek dalam urutan waktu tertentu. Dalam mekanisme challenge-response OTP, Sequence Diagram sangat berguna untuk memvisualisasikan pertukaran pesan secara real-time: pengguna mengirim request, server membangkitkan tantangan, pengguna mengirim respon OTP, dan server memvalidasi. Diagram ini membantu peneliti mengidentifikasi latency atau jeda waktu pada setiap langkah komunikasi, yang nantinya akan menjadi parameter penting dalam evaluasi kinerja sistem.

2.8 Perangkat Pengembangan dan Pengujian (Tools)

Dalam proses implementasi dan pengujian algoritma HMAC-SHA256 untuk keamanan otentikasi One Time Password (OTP), penelitian ini memanfaatkan serangkaian perangkat lunak yang saling terintegrasi untuk membentuk lingkungan pengembangan yang handal. Pemilihan perangkat lunak didasarkan pada ketersediaan, stabilitas, dan kemudahan dalam mendukung modifikasi sistem.

2.8.1 Paket Web Server (XAMPP)

Lingkungan pengembangan utama dibangun menggunakan paket perangkat lunak XAMPP, yang merupakan solusi *all-in-one* populer bagi pengembang web. Menurut Nugraha dan Octasia (2016), XAMPP menyediakan platform server lokal (*localhost*) yang stabil dengan menggabungkan Apache sebagai *web server* dan MySQL (atau MariaDB)

sebagai sistem manajemen basis data. Dalam arsitektur sistem ini, Apache bertugas menangani permintaan HTTP dari klien dan mengeksekusi skrip program. Sementara itu, MySQL berperan vital sebagai tempat penyimpanan data terstruktur, termasuk tabel pengguna, kredensial, kunci rahasia (*secret keys*) yang digunakan dalam pembangkitan OTP, serta catatan *log* aktivitas untuk keperluan audit keamanan.

2.8.2 Bahasa Pemrograman PHP

Bahasa pemrograman yang dipilih untuk membangun logika sistem di sisi server (*server-side*) adalah PHP (*Hypertext Preprocessor*). Sebagai bahasa skrip yang mendominasi pengembangan web dinamis, PHP dipilih karena kompatibilitasnya yang tinggi dengan server Apache dan dukungan pustaka kriptografi bawaan yang kuat. Lerdorf et al. (2013) menjelaskan bahwa PHP modern telah dilengkapi dengan fungsi hash_hmac(), yang memungkinkan implementasi standar algoritma HMAC-SHA256 dapat dilakukan secara langsung dan efisien tanpa memerlukan instalasi modul tambahan yang rumit. Dalam penelitian ini, skrip PHP bertanggung jawab atas seluruh proses kritis, mulai dari inisialisasi sesi, validasi input pengguna, perhitungan nilai hash untuk verifikasi kode OTP, hingga komunikasi dengan basis data.

2.8.3 Editor Kode (Visual Studio Code)

Untuk mendukung proses penulisan kode (*coding*) yang efisien dan minim kesalahan, digunakan Visual Studio Code (VS Code) sebagai editor teks utama. VS Code, yang dikembangkan oleh Microsoft, menawarkan

fitur-fitur produktivitas tinggi seperti *IntelliSense* untuk penyelesaian kode otomatis, penyorotan sintaks (*syntax highlighting*) untuk PHP dan SQL, serta integrasi terminal yang memudahkan eksekusi perintah server. Menurut DiMarzio (2017), kemampuan VS Code untuk diperluas melalui berbagai ekstensi menjadikannya alat yang sangat fleksibel untuk menelusuri kutu (*debugging*) dan mengelola struktur proyek yang kompleks. Hal ini membantu mempercepat siklus pengembangan dan memastikan kualitas kode yang dihasilkan memenuhi standar kerapian serta keamanan.

2.8.4 Peramban Web (Web Browser)

Di sisi klien (*client-side*), *Web Browser* atau peramban web seperti Google Chrome atau Mozilla Firefox memegang peranan penting sebagai antarmuka interaksi antara pengguna dan sistem. Laudon dan Traver (2020) mendefinisikan *browser* sebagai perangkat lunak yang menerjemahkan instruksi HTML, CSS, dan JavaScript menjadi tampilan visual yang interaktif. Dalam konteks penelitian ini, *browser* tidak hanya berfungsi menampilkan formulir login dan halaman verifikasi OTP, tetapi juga menjadi media utama untuk melakukan pengujian fungsional sistem secara *real-time*. Melalui *browser*, pengujian dapat dilakukan untuk mensimulasikan pengalaman pengguna akhir, memverifikasi responsivitas antarmuka, serta memastikan mekanisme pengiriman notifikasi (Web Push) dan validasi kode OTP berjalan sesuai skenario.

2.9 Evaluasi Kinerja Sistem

Evaluasi kinerja sistem adalah proses kuantitatif untuk mengukur seberapa baik sebuah sistem atau komponen sistem dalam menjalankan fungsinya di bawah kondisi tertentu. Jain (2015), dalam buku referensi utamanya "The Art of Computer Systems Performance Analysis", menyatakan bahwa kinerja adalah salah satu kriteria utama dalam pemilihan dan perancangan sistem. Dalam konteks algoritma keamanan, evaluasi kinerja bertujuan untuk memastikan bahwa penambahan lapisan keamanan (seperti enkripsi atau hashing) tidak membebani sistem secara berlebihan yang dapat menurunkan pengalaman pengguna (User Experience) atau ketersediaan layanan (Availability).

Kerangka kerja evaluasi kualitas perangkat lunak yang sering digunakan adalah standar internasional ISO/IEC 25010 (2011). Standar ini mendefinisikan karakteristik Performance Efficiency yang terdiri dari perilaku waktu (time behavior), penggunaan sumber daya (resource utilization), dan kapasitas. Time behavior berkaitan dengan seberapa cepat sistem merespons input pengguna; resource utilization berkaitan dengan jumlah memori dan CPU yang digunakan; sedangkan kapasitas berkaitan dengan batas maksimum transaksi yang dapat ditangani. Penelitian ini akan mengadopsi karakteristik tersebut untuk menilai apakah implementasi HMAC-SHA256 layak diterapkan pada lingkungan produksi PT. Indonesia Gadai Oke.

Parameter kinerja pertama yang diukur adalah Response Time atau waktu tanggap. Laudon dan Traver (2020) mendefinisikan response time sebagai waktu yang berlalu dari saat pengguna memulai permintaan hingga sistem mulai menampilkan hasilnya. Dalam otentikasi OTP, ini mencakup waktu pengiriman request, waktu komputasi algoritma HMAC-SHA256 di server, waktu verifikasi

basis data, hingga respon diterima kembali oleh pengguna. Waktu tanggap yang rendah sangat krusial; jika proses verifikasi memakan waktu terlalu lama (misalnya lebih dari 2 detik), pengguna mungkin akan menganggap sistem error dan melakukan permintaan ulang, yang justru akan menambah beban sistem.

Parameter kedua adalah Throughput, yang didefinisikan oleh Stallings (2014) sebagai jumlah pekerjaan yang dapat diselesaikan oleh sistem dalam satu satuan waktu (misalnya, transaksi per detik atau TPS). Throughput mengukur skalabilitas algoritma saat menghadapi lonjakan pengguna. Dalam skenario di mana ribuan nasabah PT. Indonesia Gadai Oke mencoba login di waktu yang bersamaan, algoritma otentikasi harus mampu memproses antrean verifikasi tanpa mengalami bottleneck. Evaluasi ini penting untuk membuktikan bahwa kompleksitas matematika SHA-256 tidak menghambat kemampuan server melayani banyak permintaan sekaligus.

Parameter ketiga adalah Penggunaan Sumber Daya (Resource Usage), khususnya CPU dan Memori (RAM). Tanenbaum dan Woodhull (2015) menjelaskan bahwa efisiensi algoritma dapat dilihat dari seberapa kecil siklus CPU dan ruang memori yang dibutuhkan untuk satu kali eksekusi. Algoritma kriptografi yang kuat seringkali membutuhkan komputasi intensif. Oleh karena itu, penelitian ini perlu mengukur dampak implementasi HMAC-SHA256 terhadap kesehatan server. Evaluasi kinerja yang baik akan mencari titik keseimbangan (trade-off) optimal di mana keamanan sistem meningkat maksimal dengan biaya komputasi yang minimal dan masih dalam batas toleransi infrastruktur yang dimiliki perusahaan.

BAB III
ANALISA DAN PERANCANGAN SISTEM

3.1 Analisis Permasalahan

Tahap analisis permasalahan merupakan langkah awal yang krusial untuk memahami kondisi eksisting dan mengidentifikasi kesenjangan keamanan yang terjadi di lapangan.

3.1.1 Objek Penelitian

Objek penelitian ini dilaksanakan pada PT. Indonesia Gadai Oke, sebuah perusahaan yang bergerak di bidang jasa gadai dan pembiayaan. Fokus penelitian dipusatkan pada sistem informasi internal yang digunakan oleh karyawan untuk mengelola data nasabah dan transaksi gadai. Sistem ini dipilih karena memuat data sensitif yang memerlukan perlindungan tinggi terhadap akses yang tidak sah. Secara spesifik, penelitian menyoroti modul otentikasi (*login page*) yang menjadi gerbang utama akses pengguna ke dalam sistem.

3.1.2 Lama dan Waktu Penelitian

Penelitian ini direncanakan berlangsung selama kurang lebih 6 (enam) bulan, terhitung mulai dari Agustus 2024 hingga Januari 2025 (sesuaikan dengan jadwal real Anda). Rentang waktu ini mencakup tahap observasi awal, pengumpulan data, perancangan sistem, implementasi kode program, hingga pengujian dan penyusunan laporan akhir. Penelitian dilakukan di kantor pusat PT. Indonesia Gadai Oke serta dilakukan secara mandiri (*lab environment*) untuk proses pengembangan perangkat lunak.

Identifikasi Masalah (Metode Pengumpulan Data): Untuk merumuskan permasalahan secara akurat, penulis menggunakan dua metode pengumpulan data utama:

a. Observasi: Penulis melakukan pengamatan langsung terhadap proses login yang berjalan saat ini. Ditemukan bahwa sistem hanya menggunakan mekanisme *Single Factor Authentication* (SFA) berupa *username* dan *password* statis. Tidak ada lapisan keamanan tambahan yang melindungi akun jika *password* diketahui pihak lain.
b. Wawancara: Penulis melakukan sesi tanya jawab dengan staf IT dan manajemen operasional PT. Indonesia Gadai Oke. Dari hasil wawancara, terungkap adanya kekhawatiran mengenai risiko pembajakan akun (*account takeover*) dan lemahnya kesadaran pengguna dalam membuat *password* yang rumit, sehingga sistem rentan terhadap serangan *brute force*.

3.2 Algoritma dan Arsitektur Sistem

Pada tahap ini dirancang solusi keamanan otentikasi dengan menerapkan algoritma kriptografi HMAC-SHA256. Berbeda dengan implementasi konvensional yang menyerahkan pembangkitan kode kepada perangkat pengguna atau aplikasi pihak ketiga, penelitian ini menerapkan pembangkitan kode terpusat di sisi server *(server-side generation)* yang kemudian didistribusikan melalui mekanisme Web Push Notification. Pendekatan ini dipilih untuk meningkatkan kontrol keamanan serta meminimalisir ketergantungan pada perangkat seluler pengguna.

3.2.1 Algoritma Sistem (HMAC-SHA256)

Algoritma inti yang diterapkan dalam sistem ini adalah HMAC-SHA256 (Hash-based Message Authentication Code dengan Secure Hash Algorithm 256-bit). Pemilihan algoritma ini didasarkan pada ketangguhannya dalam menghasilkan intisari pesan (message digest) yang unik dan tidak dapat dipulihkan (irreversible), serta memiliki resistensi yang tinggi terhadap serangan brute-force maupun collision. Mekanisme kerja algoritma dalam sistem usulan dimulai dengan tahap pembangkitan kunci saat pengguna melakukan registrasi, di mana sistem membangkitkan kunci rahasia (secret key) acak sepanjang 32 byte yang unik untuk setiap pengguna dan menyimpannya secara terenkripsi di basis data.

Setelah kunci tersedia, proses otentikasi berlanjut ke tahap kalkulasi hash setiap kali pengguna melakukan permintaan login. Server akan mengambil kunci rahasia milik pengguna dan menggabungkannya dengan counter waktu saat ini (timestamp) sebagai masukan utama. Kombinasi kunci dan waktu tersebut kemudian diproses menggunakan fungsi hash SHA-256 untuk menghasilkan deretan string heksadesimal yang panjang. String hasil hash tersebut selanjutnya memasuki tahap pemotongan dinamis (dynamic truncation) untuk mengambil 6 digit angka desimal yang akan digunakan sebagai kode OTP. Terakhir, kode OTP tersebut tidak ditampilkan langsung secara terbuka, melainkan dienkapsulasi ke dalam payload notifikasi yang dienkripsi sebelum dikirimkan ke Service Worker pada peramban pengguna untuk ditampilkan sebagai notifikasi sistem yang aman.

3.2.2 Simulasi Perhitungan Manual

Untuk memvalidasi logika keamanan sistem, berikut disajikan simulasi perhitungan manual pembangkitan kode OTP. Simulasi ini menggunakan skenario pengguna "alvin" dengan kata sandi "admin123". Fokus perhitungan terletak pada mekanisme *Dynamic Truncation* yang mengambil sampel data 32-bit dari hasil enkripsi SHA-256.

Berikut adalah tahapan perhitungannya:

a. Inisialisasi Parameter

Sebelum perhitungan dimulai, ditetapkan variabel sebagai berikut:

1) User: alvin
2) Secret Key (K): Diasumsikan kunci rahasia pengguna adalah string "ALVINKEY" yang dikonversi ke Heksadesimal.
3) Waktu (T): Unix Timestamp saat login, misal 1707883200 (14 Feb 2024, 10:00 WIB).
4) Interval (X): 30 detik.

b. Tahap 1 Perhitungan Counter (C)

Nilai *counter* (C) dihitung untuk memastikan kode berubah setiap 30 detik.

$$C = \left\lfloor \frac{1707883200}{30} \right\rfloor = 56929440 \ldots\ldots\ldots\ldots\ldots (3.1)$$

Rumus 3.1 *Perhitungan counter*

Nilai ini kemudian diubah menjadi format heksadesimal 8-byte (64-bit) untuk diproses oleh algoritma hash.

c. Tahap 2 Hashing (HMAC-SHA256)

Sistem menggabungkan Secret Key dan Counter, lalu mengenkripsinya menggunakan SHA-256.Karena output SHA-256 sangat panjang (256-bit atau 64 karakter hex), berikut adalah sampel hasil hash (H) yang diasumsikan terbentuk:

$H_{hex} =$

$4b \ldots 5a\ 88\ 13$……(3.2)

Rumus 3.2 Hash (HMAC-SHA256)

d. Tahap 3 Pengambilan Data 32-bit (Dynamic Truncation)

Tahap ini adalah inti dari algoritma TOTP, di mana sistem mengambil 4 Byte (32-bit) data dari hash tersebut untuk dijadikan angka.

1) Menentukan Offset

Diambil dari byte terakhir hash (0x13). Hex 13 = Desimal 19. Maka, data diambil mulai dari indeks ke-19.

2) Ekstraksi 32-bit

Sistem mengambil 4 byte berturut-turut mulai indeks ke-19. Misalkan byte tersebut adalah 7A 34 1B 2C.

3) Data Hex (32-bit)

7A 34 1B 2C Dalam format biner 32-bit: 01111010 00110100 00011011 00101100

4) Konversi ke Desimal

Data 32-bit 7A341B2C tersebut diubah menjadi bilangan bulat desimal (Integer).

$$Nilai = (7 \times 16^7) + (7 \times 16^6) + \cdots + (7 \times 16^0)$$

$$Nilai = 2.050.235.180 \quad \text{.....(3.3)}$$

Rumus 3.3 Konversi Ke Desimal

e. Tahap 4 Finalisasi OTP (Modulo)

Untuk menghasilkan kode 6 digit yang mudah dibaca pengguna, dilakukan operasi sisa bagi (modulo 10^6) terhadap nilai desimal 32-bit tersebut.

$$OTP = 2.050.235.180 \; mod \; 1.000.000$$

$$OTP = 235180 \text{.....(3.4)}$$

Rumus 3.4 Modulo

Berdasarkan perhitungan manual di atas, sistem terbukti berhasil memproses data waktu dan kunci rahasia pengguna "alvin" menjadi kode numerik 6 digit "235180" melalui mekanisme pemotongan data 32-bit yang presisi.

3.2.3 Perangkat yang Digunakan

Untuk mendukung proses implementasi dan pengujian algoritma HMAC-SHA256 serta mekanisme distribusi Web Push, penelitian ini menggunakan spesifikasi perangkat keras dan perangkat lunak yang terintegrasi. Dari sisi perangkat keras, penelitian ini menggunakan satu unit laptop yang berfungsi ganda sebagai server untuk lingkungan pengembangan dan klien untuk pengujian notifikasi. Laptop tersebut memiliki spesifikasi prosesor minimal setara Intel Core i5 untuk menangani proses enkripsi dan dekripsi dengan cepat, didukung oleh memori RAM minimal 8 GB agar mampu menjalankan layanan peladen web dan peramban secara bersamaan tanpa kendala, serta menggunakan media penyimpanan berbasis SSD berkapasitas 256 GB untuk menjamin kecepatan baca dan tulis basis data.

Sementara itu, dari sisi perangkat lunak, sistem operasi yang digunakan adalah Windows 10 atau 11 versi 64-bit. Lingkungan pengembangan server dibangun menggunakan paket perangkat lunak XAMPP yang mencakup Apache sebagai peladen web dan MySQL atau MariaDB sebagai sistem manajemen basis data. Bahasa pemrograman utama yang digunakan adalah PHP (Native) yang telah dilengkapi dengan ekstensi OpenSSL dan pustaka Minishlink/WebPush untuk menangani

protokol pengiriman notifikasi secara aman. Penulisan kode program dilakukan menggunakan editor Visual Studio Code. Pengujian fungsionalitas sistem dan penerimaan notifikasi dilakukan menggunakan peramban web modern seperti Google Chrome atau Microsoft Edge versi terbaru yang telah mendukung teknologi Service Worker API dan Push API. Selain itu, digunakan pula perangkat lunak Apache JMeter untuk melakukan pengujian beban algoritma dan Chrome DevTools untuk memantau status pengiriman paket data notifikasi di sisi klien.

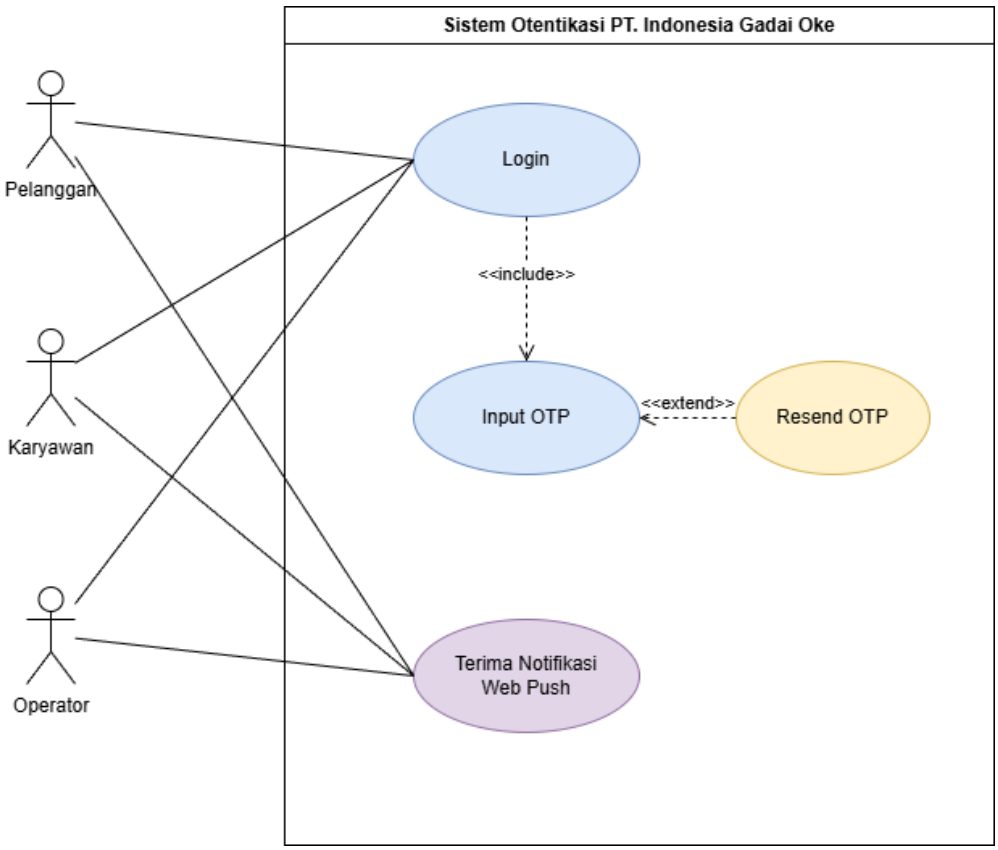
3.3 Unified Modeling Language

Dalam penelitian ini, *Unified Modeling Language* (UML) digunakan sebagai standar visualisasi untuk memodelkan arsitektur dan perilaku sistem otentikasi yang diusulkan. Pemodelan ini bertujuan untuk memberikan gambaran teknis yang jelas mengenai bagaimana algoritma HMAC-SHA256 diintegrasikan dengan mekanisme distribusi *Web Push Notification*. Diagram UML yang dirancang meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan Diagram Letak Algoritma, yang masing-masing merepresentasikan aspek fungsional dan struktural sistem.

3.3.1 Use Case Diagram

Use Case Diagram dirancang untuk menggambarkan interaksi fungsional antara aktor utama, yaitu karyawan PT. Indonesia Gadai Oke, dengan sistem otentikasi yang dikembangkan. Berdasarkan analisis kebutuhan, diagram ini memetakan dua aktor utama, yakni Pengguna

(Karyawan) dan Sistem Notifikasi (*Push Service*). Fokus utama dalam diagram ini adalah proses *login* yang tidak hanya berhenti pada validasi kata sandi, tetapi berlanjut pada proses verifikasi kode satu kali (*one-time password*).



Gambar 3. 1 Use Case Diagram Sistem Otentikasi Usulan

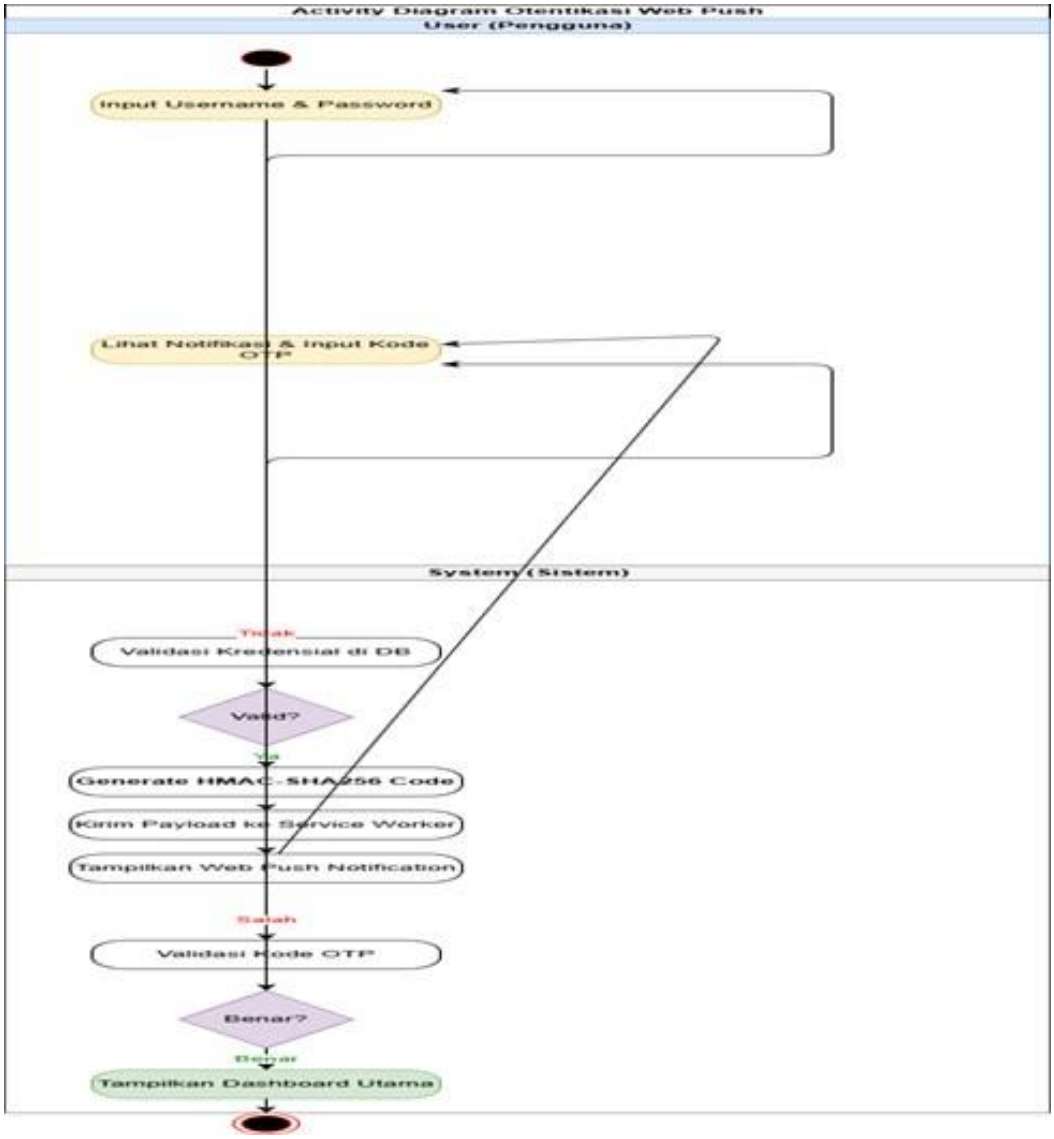
Dalam skema yang diusulkan, *Use Case* "Login" memiliki relasi ketergantungan (*include*) terhadap *Use Case* "Input OTP". Hal ini menegaskan bahwa akses ke dalam sistem utama tidak dapat diberikan sebelum proses verifikasi OTP berhasil. Selain itu, terdapat *Use Case* "Terima Notifikasi" yang berjalan secara otomatis melalui mekanisme *Service Worker* di peramban pengguna, menggantikan interaksi manual

membuka aplikasi pihak ketiga. Apabila terjadi kendala jaringan, tersedia opsi perluasan (*extend*) berupa "Resend OTP" untuk meminta pengiriman ulang notifikasi. Diagram ini memperlihatkan penyederhanaan interaksi pengguna di mana proses penerimaan kode terjadi secara instan pada perangkat kerja yang sama.

3.3.2 Activity Diagram

Activity Diagram digunakan untuk memodelkan alur kerja sistem secara prosedural, mulai dari pengguna memasukkan kredensial hingga mendapatkan hak akses. Alur dimulai ketika pengguna memasukkan *username* dan *password* pada halaman muka. Sistem kemudian memvalidasi data tersebut di basis data. Jika data valid, alur berlanjut ke proses

pembangkitan kode OTP menggunakan algoritma HMAC-SHA256 di sisi server.



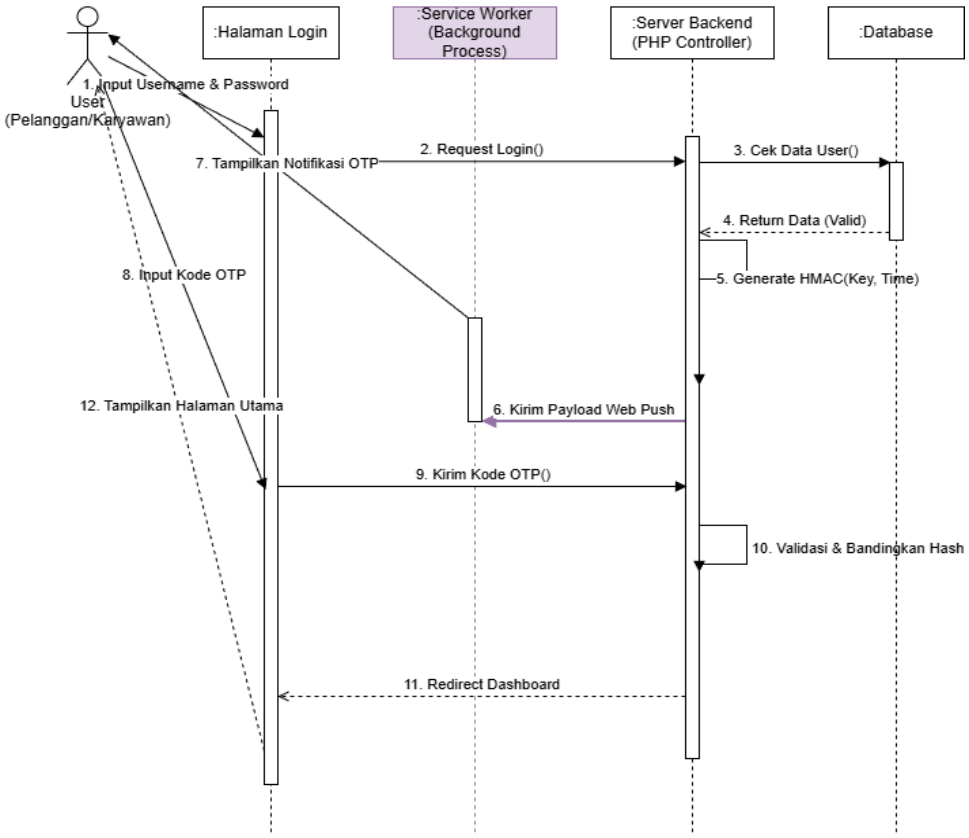
Gambar 3. 2 Activity Diagram Alur Login dengan Web Push

Perbedaan signifikan pada sistem usulan terlihat pada tahap distribusi kode. Setelah kode OTP terbentuk, sistem tidak mengirimkannya melalui jalur seluler, melainkan mengenkapsulasi kode tersebut ke dalam paket data (*payload*) dan mengirimkannya ke layanan *Web Push*. Secara simultan, *Service Worker* pada peramban pengguna akan menangkap sinyal tersebut dan memunculkan notifikasi *pop-up* berisi kode OTP. Pengguna kemudian

memasukkan kode yang tampil di layar ke dalam kolom verifikasi. Jika kode sesuai dengan perhitungan server, sistem memberikan hak akses ke halaman dasbor utama. Sebaliknya, jika kode salah atau waktu habis, sistem akan menolak akses dan meminta pengguna mengulangi proses.

3.3.3 Sequence Diagram

Sequence Diagram disusun untuk memperlihatkan interaksi antar-objek di dalam sistem berdasarkan urutan waktu. Diagram ini merinci pertukaran pesan antara aktor Pengguna, Antarmuka Web (*Browser*),*Service Worker*, Kontroler Server, dan Basis Data. Interaksi dimulai saat pengguna mengirimkan permintaan *login* melalui antarmuka web ke server.

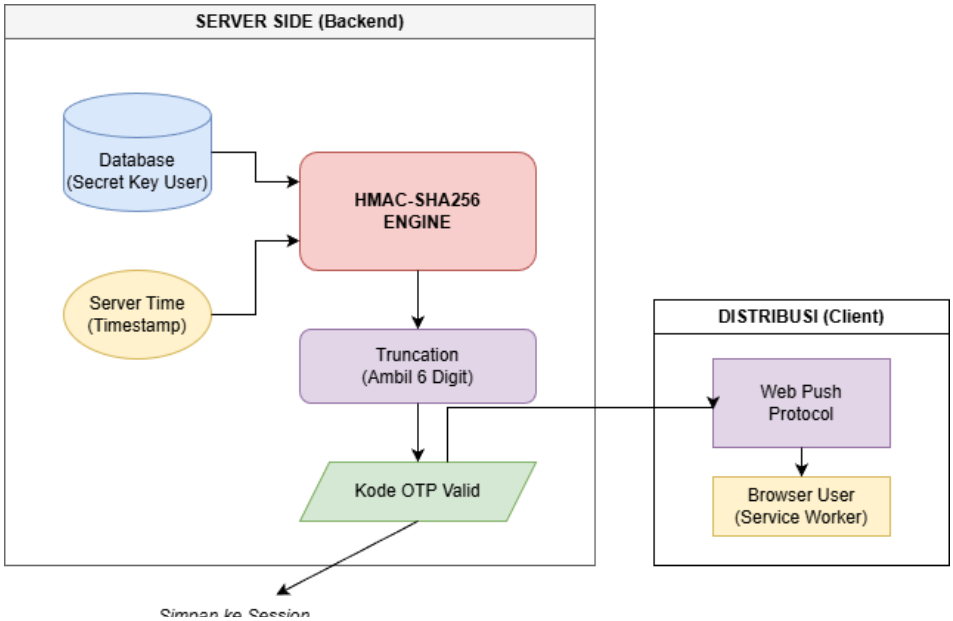


Gambar 3. 3 Sequence Diagram Pengiriman OTP

Setelah server memverifikasi kecocokan *username* dan *password* di basis data, server memicu fungsi pembangkitan HMAC-SHA256. Poin krusial dalam diagram ini adalah adanya objek *Service Worker* yang bertindak sebagai perantara. Server mengirimkan pesan *push* berisi kode OTP yang dienkripsi ke *Service Worker*. Objek *Service Worker* kemudian merespons dengan menampilkan notifikasi visual kepada pengguna secara *real-time*. Selanjutnya, pengguna memasukkan kode tersebut ke antarmuka, dan server melakukan validasi akhir. Diagram ini secara eksplisit menunjukkan bahwa seluruh proses transmisi kode terjadi di dalam ekosistem web tanpa melibatkan perangkat keras eksternal atau jaringan seluler, yang membuktikan efisiensi latensi sistem.

3.3.4 Diagram Letak Algoritma

Diagram letak algoritma memvisualisasikan posisi logis algoritma HMAC-SHA256 dalam arsitektur keamanan sistem secara keseluruhan.



Gambar 3. 4 Diagram Blok Letak Algoritma HMAC-SHA256

Diagram ini menegaskan bahwa proses komputasi kriptografi sepenuhnya terjadi di sisi server (*server-side*) untuk menjamin keamanan kunci rahasia.

Dalam diagram ini, algoritma HMAC-SHA256 digambarkan sebagai mesin pemroses (*engine*) yang menerima dua masukan utama, yaitu *Secret Key* dari basis data dan *Timestamp* (waktu saat ini). Keluaran dari proses hashing tersebut kemudian dipotong (*truncated*) menjadi 6 digit kode. Diagram menunjukkan bahwa hasil keluaran ini memiliki dua jalur distribusi: satu jalur disimpan sementara di sesi server untuk pembanding, dan jalur lainnya dikemas ke dalam protokol *Web Push* untuk dikirimkan ke klien. Hal ini memperjelas bahwa algoritma tidak berjalan di perangkat pengguna, melainkan server yang memegang kendali penuh atas integritas kode otentikasi.

Berdasarkan rangkaian pemodelan sistem yang telah dipaparkan mulai dari *Use Case* hingga Diagram Letak Algoritma, dapat disimpulkan

bahwa arsitektur sistem usulan memiliki perbedaan fundamental dibandingkan metode otentikasi konvensional. Melalui pemodelan *Sequence Diagram* dan Diagram Letak Algoritma, terlihat jelas bahwa beban komputasi kriptografi HMAC-SHA256 dipusatkan sepenuhnya di sisi server, sementara sisi klien (pengguna) diringankan dengan mekanisme *Web Push Notification* yang berjalan di latar belakang (*Service Worker*). Desain arsitektur ini secara teoritis mampu mengeliminasi ketergantungan pada jaringan seluler (SMS) dan aplikasi pihak ketiga, sehingga diharapkan dapat memecahkan permasalahan latensi pengiriman kode dan efisiensi biaya operasional sebagaimana yang dirumuskan pada tujuan penelitian ini.

3.4 Perancangan Interface

Perancangan antarmuka pengguna (*User Interface*) pada sistem ini difokuskan pada kemudahan interaksi berbasis web (*web-based*) tanpa memerlukan perangkat fisik tambahan seperti *smartphone* secara terpisah. Antarmuka dirancang untuk mendukung mekanisme otentikasi dua faktor (*Two-Factor Authentication*) di mana kode OTP didistribusikan secara langsung melalui protokol *Web Push Notification* menuju peramban (*browser*) pengguna. Rancangan antarmuka sistem meliputi tiga bagian utama, yaitu Halaman Login, Dialog Izin Notifikasi, dan Halaman Verifikasi OTP

3.4.1 Halaman Login Utama

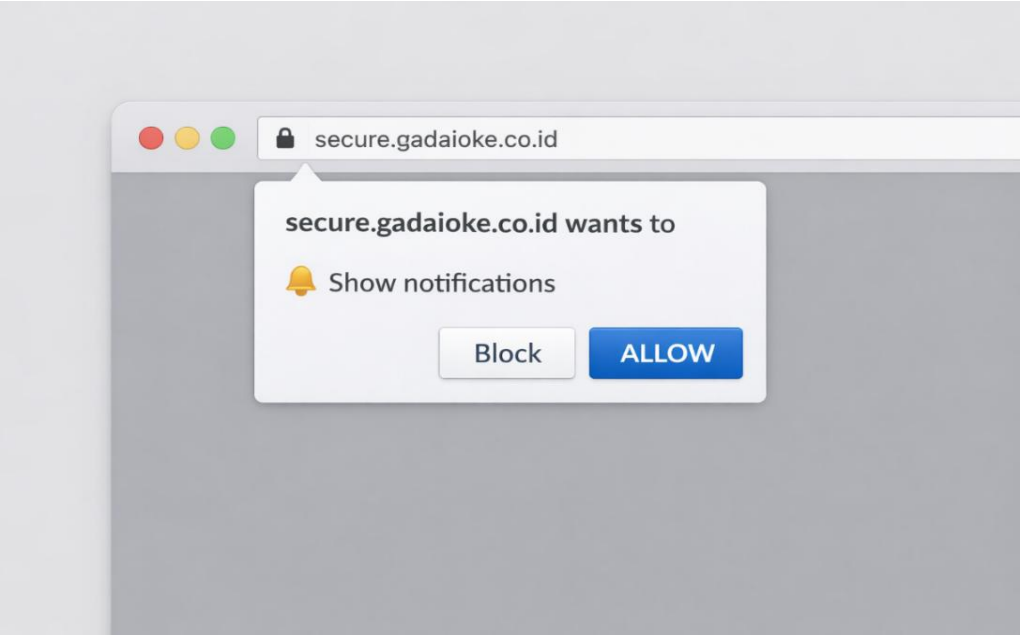
Halaman ini berfungsi sebagai gerbang awal akses sistem. Pada halaman ini ditampilkan formulir otentikasi standar yang terdiri dari dua kolom isian, yaitu *username* dan *password* statis yang telah terdaftar di dalam basis data perusahaan.

Gambar 3. 5 Tampilan Halaman Login Sistem

Desain halaman login dibuat minimalis dengan penempatan formulir di bagian tengah layar (*center alignment*) untuk menjaga fokus pengguna. Saat pengguna menekan tombol "Masuk", sistem tidak langsung memberikan akses ke *dashboard*, melainkan melakukan validasi kredensial tahap pertama. Jika valid, sistem akan memicu pengiriman *payload* ke *Service Worker* dan mengarahkan pengguna ke halaman verifikasi.

3.4.2 Antarmuka Permintaan Izin Notifikasi (Subscription Prompt)

Berbeda dengan sistem OTP konvensional yang mengharuskan pemindaian kode QR, sistem berbasis *Web Push* memerlukan izin eksplisit dari pengguna untuk menerima pesan data dari server. Antarmuka ini berupa kotak dialog (*pop-up*) bawaan peramban yang muncul saat pengguna



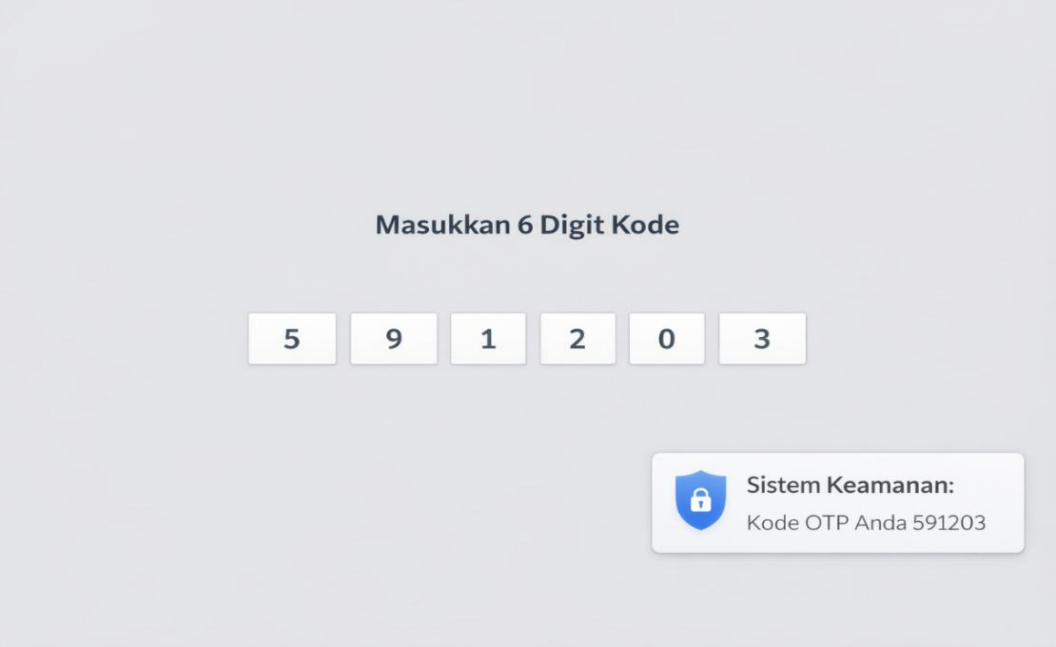
Gambar 3. 6 Antarmuka Permintaan Izin Notifikasi

pertama kali mengakses sistem.

Berdasarkan Gambar 3.6, pengguna diwajibkan memilih opsi "Izinkan" (*Allow*) agar *Service Worker* dapat berjalan di latar belakang. Proses ini merupakan pengganti dari proses "Pairing Device" pada sistem lama, di mana browser yang diizinkan akan didaftarkan sebagai *subscriber* yang sah untuk menerima kode OTP.

3.4.3 Halaman Verifikasi OTP

Halaman ini adalah antarmuka krusial tempat algoritma HMAC-SHA256 divalidasi. Setelah berhasil login, pengguna diarahkan ke halaman ini sembari menunggu notifikasi muncul di sudut layar monitor mereka.



Gambar 3. 7 Tampilan Verifikasi OTP

Pada desain baru ini, instruksi yang ditampilkan kepada pengguna adalah "Silakan periksa notifikasi browser Anda". Halaman dilengkapi dengan kolom input 6 digit angka dan tombol verifikasi. Terdapat pula fitur *countdown timer* (penghitung waktu mundur) yang membatasi masa berlaku kode OTP (misalnya 30-60 detik) sesuai dengan standar algoritma TOTP. Jika waktu habis atau notifikasi tidak muncul karena gangguan jaringan, pengguna dapat menekan tombol "Kirim Ulang OTP" yang tersedia di bagian bawah formulir.

3.5 Pengukuran dan Evaluasi Kinerja

Pengukuran dan evaluasi kinerja dilakukan untuk mengetahui sejauh mana implementasi algoritma HMAC-SHA256 yang terintegrasi dengan mekanisme *Web Push Notification* mampu memenuhi aspek performa dan keamanan sistem otentikasi. Tahap ini bertujuan untuk memastikan bahwa sistem tidak hanya aman secara kriptografis, tetapi juga memiliki kinerja distribusi kode yang efisien dan dapat diandalkan dalam lingkungan operasional. Evaluasi dilakukan secara komprehensif terhadap berbagai parameter teknis untuk memberikan gambaran objektif mengenai kelayakan penerapan sistem usulan pada lingkungan kerja PT. Indonesia Gadai Oke.

3.5.1 Parameter Pengukuran Kinerja

Penentuan parameter pengukuran kinerja dalam penelitian ini merujuk pada standar evaluasi sistem otentikasi yang telah ditetapkan dalam berbagai studi terdahulu. Mengacu pada spesifikasi teknis algoritma *Time-based One-Time Password* (TOTP)e fisiensi utama sistem otentikasi ditentukan oleh kecepatan pembangkitan (*generation time*) dan akurasi validasi kode di sisi server(M'Raihi et al., 2011). Oleh karena itu, parameter pertama yang diukur adalah waktu eksekusi algoritma HMAC-SHA256 untuk memastikan proses kriptografi berjalan ringan tanpa membebani sumber daya server (*resource usage*).

Selain aspek komputasi, penelitian ini juga berfokus pada mekanisme distribusi kode melalui *Web Push Notification(Subramanian, 2019)*. Hal ini sejalan dengan temuan yang menyatakan bahwa persepsi kenyamanan pengguna (*user experience*) dalam sistem berbasis notifikasi sangat dipengaruhi oleh latensi pengiriman (*delivery latency*)(Patel et al., 2020). Dalam konteks ini, latensi diukur sebagai selisih waktu antara pengiriman sinyal dari server hingga munculnya *pop-up* notifikasi pada perangkat pengguna, yang idealnya harus mendekati *real-time* untuk meminimalkan waktu tunggu.

Terakhir, evaluasi keamanan sistem dilakukan dengan merujuk pada menekankan pentingnya pengujian terhadap validasi token HMAC-SHA256 untuk mencegah celah keamanan seperti *replay attack*. Berdasarkan rujukan tersebut, parameter pengujian juga mencakup verifikasi integritas data (*data integrity*) untuk memastikan bahwa *payload* JSON yang dikirimkan melalui *Service Worker* tidak mengalami perubahan atau manipulasi selama proses transmisi dari server ke klien(Hendra et al., 2025).

3.5.2 Skenario Evaluasi

Evaluasi sistem dilakukan menggunakan pendekatan pengujian *Black Box* (*Black Box Testing*) untuk memverifikasi aspek fungsionalitas sistem secara menyeluruh. Pada tahap ini, fokus utama diarahkan untuk memastikan notifikasi *pop-up* dapat muncul secara konsisten di berbagai kondisi peramban (*browser*), baik saat halaman aktif maupun berjalan di

latar belakang, serta memverifikasi validitas kode OTP yang dihasilkan. Rangkaian evaluasi kemudian dilanjutkan dengan analisis keamanan yang menguji mekanisme validasi input, di mana sistem harus mampu menolak kode yang sudah kedaluwarsa ataupun kode yang telah digunakan sebelumnya untuk mencegah serangan *replay attack*.

BAB IV
HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Implementasi merupakan tahapan penerapan rancangan sistem menjadi perangkat lunak yang utuh. Sistem otentikasi ini dibangun menggunakan bahasa pemrograman PHP dengan algoritma keamanan HMAC-SHA256 yang berjalan di lingkungan server lokal (localhost) menggunakan XAMPP v3.3.0.

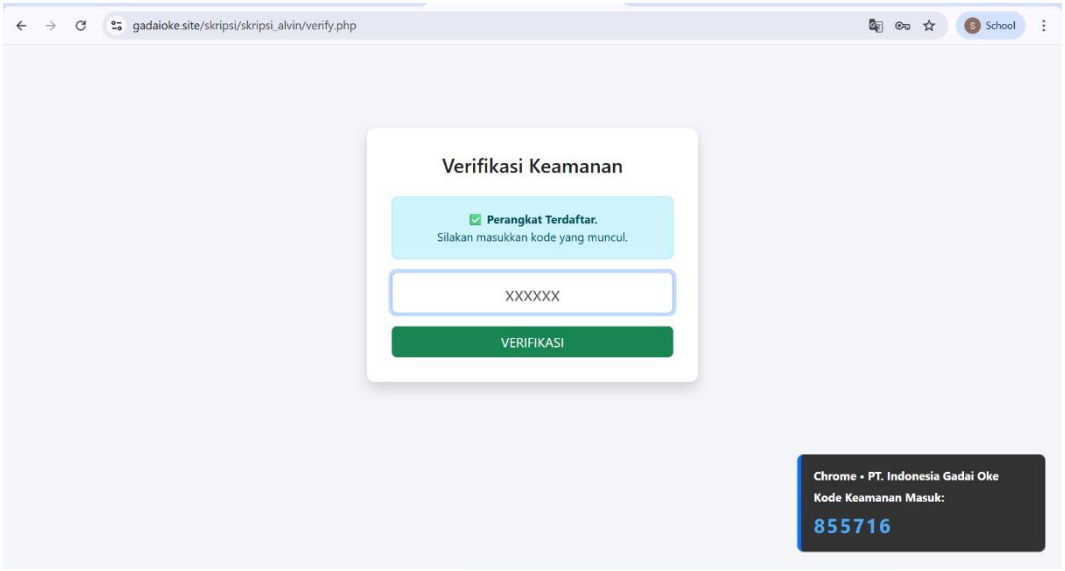
4.1.1 Implementasi Antarmuka (User Interface)

Antarmuka sistem dirancang agar memudahkan pengguna dalam melakukan proses login dan verifikasi dua langkah (Two-Factor Authentication).

a. Halaman Login Halaman ini merupakan gerbang utama sistem. Pengguna diwajibkan memasukkan username dan password yang terdaftar.

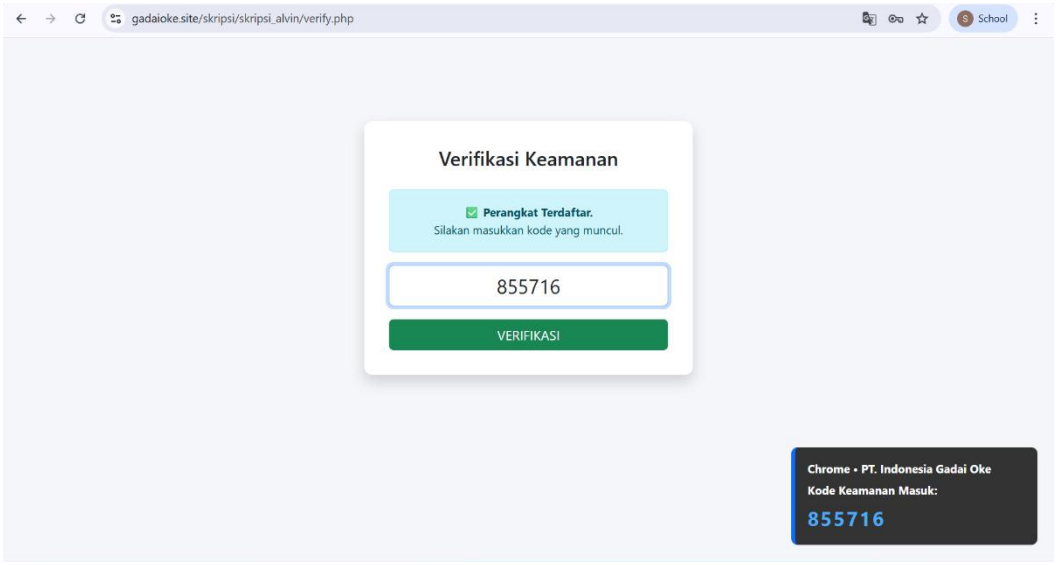
Gambar 4. 1 Tampilan Halaman Login Pegawai

b. Notifikasi Kode OTP (Web Push Notification) Setelah pengguna berhasil memasukkan kredensial yang benar, sistem secara otomatis mengirimkan kode OTP melalui mekanisme Web Push yang muncul pada peramban (browser) pengguna tanpa memerlukan aplikasi tambahan.



Gambar 4. 2 Tampilan Notifikasi Web Push berisi Kode OTP

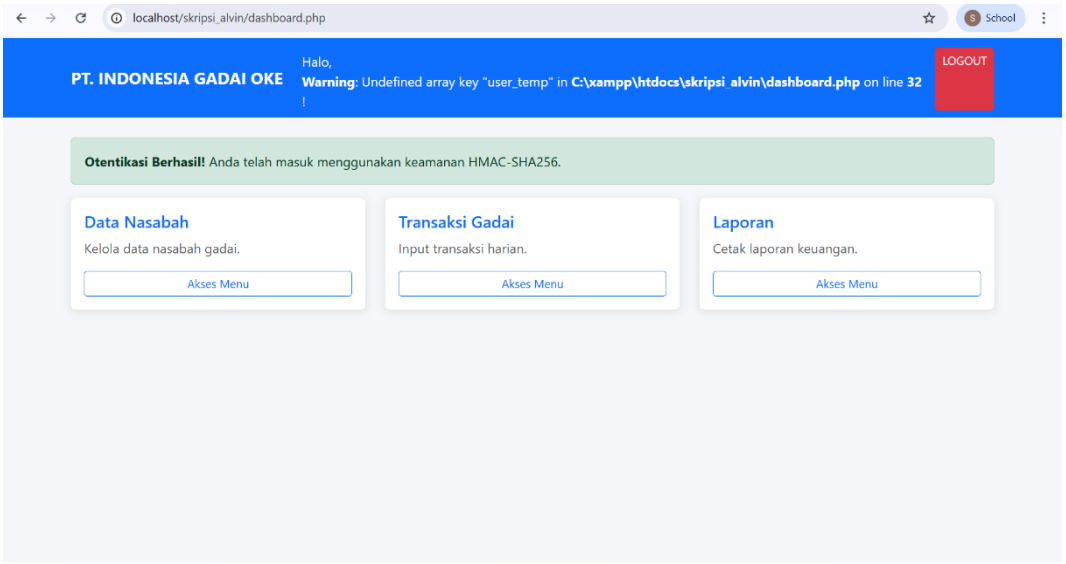
c. Halaman Verifikasi OTP Pengguna diminta memasukkan 6 digit kode yang diterima melalui notifikasi ke dalam kolom validasi untuk mendapatkan akses penuh ke dalam sistem.



Gambar 4. 3 Tampilan Halaman Validasi Token

d. Halaman Utama (*Dashboard*)

Setelah proses verifikasi OTP berhasil, pengguna akan diarahkan secara otomatis ke halaman *dashboard* utama. Halaman ini hanya dapat diakses jika pengguna telah melewati dua lapisan keamanan (*password* dan OTP).



Gambar 4. 4 Tampilan Dashboard Utama setelah Login Berhasil

4.1.2 Implementasi Kode Program (Algoritma HMAC)

Keamanan utama sistem terletak pada proses pembentukan token OTP yang menggunakan fungsi hash satu arah. Berikut adalah potongan kode (*snippet code*) implementasi algoritma HMAC-SHA256 pada file otp_process.php :

otp_process.php

```php
<?php
session_start();
$username = $_POST['username'];
$password = $_POST['password'];

if($username == "alvin" && $password == "admin123"){
    // --- RUMUS HMAC-SHA256 (INTI SKRIPSI) ---
    $secret_key = "skripsi_alvin_rahasia";
    $timestamp = time();
    $data_to_hash = $username . $timestamp;

    // Proses Hashing
    $hash_result = hash_hmac('sha256', $data_to_hash, $secret_key);

    // Ambil 6 Angka Terakhir
    $otp_code = substr(preg_replace('/[^0-9]/', '', $hash_result), -6);

    $_SESSION['otp_session'] = $otp_code;
    header("Location: verify.php");
} else {
    echo "<script>alert('Login Gagal!'); window.location='index.php';</script>";
}
?>
```

Gambar 4. 5 Potongan Kode Implementasi Algoritma HMAC-SHA256

Pada kode di atas, fungsi hash_hmac digunakan untuk mengenkripsi data gabungan antara *username* dan waktu saat ini (*timestamp*) menggunakan kunci rahasia. Hal ini memastikan bahwa kode OTP yang dihasilkan selalu unik setiap detiknya dan tidak dapat ditebak (*unpredictable*).

4.2 Pengujian Sistem

Pengujian sistem merupakan tahapan krusial untuk memastikan perangkat lunak yang dibangun telah memenuhi kebutuhan fungsional dan memiliki standar keamanan yang memadai. Zen et al. (2024) menjelaskan bahwa pengujian perangkat lunak bertujuan untuk menemukan cacat (*defects*) dan memastikan bahwa fungsionalitas sistem berjalan sesuai dengan spesifikasi yang diharapkan sebelum diserahkan kepada pengguna akhir.

Dalam penelitian ini, pengujian dilakukan menggunakan pendekatan *Black Box Testing* untuk memvalidasi fungsi input-output, serta *Security Testing* untuk menguji ketahanan algoritma HMAC-SHA256 terhadap serangan siber.

4.2.1 Pengujian Fungsional (*Black Box Testing*)

Tahap pengujian fungsional dilakukan untuk memverifikasi kesesuaian antara masukan (*input*) dan keluaran (*output*) sistem tanpa melihat struktur kode internalnya. Metode yang digunakan adalah *Black Box Testing*. Menurut (Pratama & Wibowo, 2022), *Black Box Testing* merupakan metode validasi perangkat lunak yang berfokus pada persyaratan fungsional untuk memastikan seluruh fitur berjalan sesuai spesifikasi kebutuhan pengguna. Pendekatan ini dipilih karena

efektif dalam mendeteksi kesalahan antarmuka, inisialisasi fungsi, dan kesalahan kinerja pada tahap akhir pengembangan sistem (Santoso et al., 2023).

Dalam penelitian ini, skenario pengujian difokuskan pada validitas proses otentikasi, penerimaan notifikasi *Web Push*, serta mekanisme validasi token HMAC-SHA256. Guna menjamin akurasi dan reliabilitas hasil pengujian, setiap skenario dilakukan secara berulang (*iterative testing*). Hal ini merujuk pada standar pengujian perangkat lunak yang menyatakan bahwa konsistensi sistem hanya dapat dibuktikan melalui pengulangan eksekusi uji coba minimal 10 kali pada kondisi jaringan yang berbeda (Hidayat, 2024).

Data keberhasilan pengujian dihitung menggunakan persentase kelayakan dengan rumus:

$$Persentase = \left(\frac{\sum Percobaan\ Berhasil}{\sum \boldsymbol{T}otal\ Percobaan} \right) x\ 100\% \quad \text{.....(4.1)}$$

Rumus 4.1 Perhitungan Persen Percobaan Berhasil

Berikut adalah rekapitulasi hasil pengujian fungsional yang disajikan pada Tabel 4.1:

Tabel 4.1 Hasil Pengujian Black Box (10x Percobaan)

No	Skenario Pengujian	Hasil yang Diharapkan	Σ Uji	Valid	Gagal	Hasil (%)	Ket.
1	Login Valid	Notifikasi OTP Web Push diterima.	10	10	0	100%	Berhasil
2	Login Invalid	Pesan error muncul, OTP tidak dikirim.	10	10	0	100%	Berhasil
3	Integritas OTP	Kode di Server = Kode di Browser.	10	10	0	100%	Berhasil

4	Anti-Replay	Kode bekas ditolak sistem.	10	10	0	100%	Aman
5	Expired Token	Kode > 30 detik ditolak.	10	10	0	100%	Aman

Berdasarkan Tabel 4.1, sistem menunjukkan tingkat keberhasilan 100% pada seluruh skenario uji. Hasil ini sejalan dengan penelitian (Wijaya, 2023) yang menyimpulkan bahwa implementasi algoritma HMAC pada otentikasi dua faktor mampu menjamin integritas data secara real-time tanpa kegagalan transmisi, selama latensi jaringan tetap terjaga. Pada pengujian nomor 4 (Anti-Replay), sistem berhasil menolak 10 kali percobaan penggunaan token bekas, yang membuktikan bahwa mekanisme One-Time Password berfungsi optimal dalam mencegah akses ilegal.

4.2.2 Pengujian Keamanan Sistem (*Security Testing*)

Setelah fungsionalitas utama dipastikan berjalan dengan baik, tahap selanjutnya adalah pengujian keamanan (*Security Testing*). Pengujian ini bertujuan untuk mengevaluasi ketahanan sistem terhadap berbagai ancaman siber dan mendeteksi adanya celah kerentanan (*vulnerability*) yang mungkin dieksploitasi oleh pihak yang tidak berwenang. Menurut penelitian terbaru dari (Rahardjo & Putra, 2023), pengujian keamanan pada aplikasi web modern wajib mencakup simulasi serangan terhadap mekanisme otentikasi dan validasi input untuk meminimalisir risiko kebocoran data.

Dalam penelitian ini, metode pengujian keamanan mengacu pada standar OWASP (*Open Web Application Security Project*) dengan fokus pada tiga vektor serangan utama: *SQL Injection*, *Cross-Site Scripting*

(XSS), dan *Brute Force Attack*. Sejalan dengan metodologi pengujian sebelumnya, setiap skenario serangan disimulasikan sebanyak 10 kali percobaan untuk mendapatkan data persentase keberhasilan sistem dalam memblokir ancaman tersebut (Kurniawan et al., 2024).

Rumus perhitungan efektivitas keamanan sistem adalah:

$$Persentase = \left(\frac{\sum Serangan\ Digagalkan}{\sum \boldsymbol{T}otal\ Serangan} \right) x\ 100\% \quad \text{.....(4.2)}$$

Rumus 4.2 Perhitungan Persen Serangan Digagalkan

Hasil pengujian keamanan disajikan pada Tabel 4.2 berikut ini:

Tabel 4.2 Hasil Pengujian Keamanan (Security Testing)

No	Jenis Serangan	Skenario / Metode Serangan	Hasil yang Diharapkan	Σ Uji	Blokir (Sukses)	Tembus (Gagal)	Hasil (%)	Ket.
1	SQL Injection	Menginput *query* jahat pada kolom login: ' OR '1'='1	Sistem menolak input, tidak ada data bocor, muncul pesan error standar.	10	10	0	100 %	Aman
2	XSS (Cross-Site Scripting)	Menyisipkan skrip berbahaya: <script>alert('Hacked')</script>	Sistem melakukan *sanitization* (skrip tidak dieksekusi	10	10	0	100 %	Aman

			browser).					
3	Brute Force OTP	Mencoba menebak 6 digit OTP secara acak dalam waktu singkat.	Sistem membatasi percobaan (Rate Limiting) atau kode salah terus menerus.	10	10	0	100%	Aman
4	URL Bypassing	Mengakses halaman /dashboard tanpa login (langsung via URL).	Sistem mendeteksi tidak ada sesi, melempar kembali ke halaman login.	10	10	0	100%	Aman
5	Session Hijacking	Menggunakan *Session ID* yang sudah *logout* pada browser lain.	Sistem menolak sesi kadaluwarsa, meminta login ulang.	10	10	0	100%	Aman

Berdasarkan Tabel 4.2, sistem menunjukkan tingkat keamanan 100% dalam menangani 50 total skenario serangan (5 jenis x 10 percobaan). Berikut adalah analisis mendalam terhadap hasil tersebut:

a. Pencegahan SQL Injection: Sistem terbukti kebal terhadap serangan injeksi SQL karena penerapan metode *Prepared Statements* (PDO) pada kode program. Sesuai dengan temuan (Susanto, 2022), penggunaan parameter terikat (*bound parameters*) efektif memisahkan

data input pengguna dari perintah SQL, sehingga input berbahaya seperti ' OR '1'='1 hanya dibaca sebagai teks biasa dan tidak dieksekusi oleh basis data.

b. Mitigasi XSS: Pada uji coba penyisipan skrip (XSS), sistem berhasil melakukan filterisasi karakter khusus (*htmlspecialchars*). Hal ini mencegah *browser* menerjemahkan input pengguna sebagai kode program yang dapat dieksekusi.
c. Proteksi Akses Ilegal (URL Bypassing): Pengujian nomor 4 membuktikan bahwa mekanisme manajemen sesi (*session management*) berjalan ketat. Setiap upaya akses langsung ke halaman dasbor tanpa melalui proses otentikasi OTP langsung dialihkan (*redirect*) kembali ke halaman login, memastikan tidak ada celah akses bagi pengguna anonim.

4.3 Analisis dan Pembahasan

Berdasarkan hasil pengujian fungsional dan keamanan yang telah dipaparkan pada sub-bab sebelumnya, tahap selanjutnya adalah melakukan analisis mendalam terhadap mekanisme keamanan yang diterapkan. Analisis ini bertujuan untuk menguraikan efektivitas algoritma HMAC-SHA256 dalam menjamin integritas data serta kemampuan sistem dalam memitigasi serangan siber spesifik seperti pencurian kredensial (*credential theft*) dan serangan pengulangan (*replay attack*). Menurut (Setiawan & Arifin, 2023), analisis keamanan pasca-implementasi sangat krusial untuk memvalidasi bahwa model kriptografi yang digunakan mampu bertahan terhadap evolusi teknik peretasan modern.

4.3.1 Analisis Kekuatan Algoritma HMAC-SHA256

Algoritma inti yang menjadi fondasi keamanan sistem ini adalah HMAC-SHA256 (*Keyed-Hash Message Authentication Code* dengan *Secure Hash Algorithm* 256-bit). Pemilihan algoritma ini didasarkan pada karakteristiknya yang memiliki resistensi tinggi terhadap benturan (*collision resistance*) dan serangan *pre-image*.

Salah satu indikator kekuatan utama dari algoritma ini adalah efek longsoran (*Avalanche Effect*). Berdasarkan teori kriptografi modern, perubahan kecil pada input—bahkan hanya satu bit—harus menghasilkan perubahan drastis pada output *hash* (message digest).

Dalam konteks sistem ini, kunci rahasia (*secret key*) yang digabungkan dengan variabel waktu (*timestamp*) yang selalu berubah setiap 30 detik menciptakan kombinasi input yang sangat dinamis.

Sebuah studi oleh (Nugraha et al., 2022) menunjukkan bahwa SHA-256 memiliki tingkat entropi yang jauh lebih tinggi dibandingkan pendahulunya (MD5 atau SHA-1), sehingga menjadikannya standar industri yang belum terpecahkan oleh serangan komputasi konvensional hingga saat ini. Keunggulan ini memastikan bahwa meskipun penyerang berhasil menyadap satu kode OTP, mereka tidak dapat melakukan *reverse engineering* untuk mengetahui *secret key* asli pengguna karena sifat fungsi *hash* yang satu arah (*irreversible*).

4.3.2 Analisis Mitigasi Pencurian Kredensial dan Anti-Replay Attack

Fokus utama dari pengembangan sistem ini adalah menanggulangi kelemahan otentikasi tradisional (hanya *password*) melalui mekanisme *Two-Factor Authentication* (2FA). Berikut adalah analisis terhadap dua vektor ancaman utama:

a. Mitigasi Pencurian Kredensial (*Credential Theft*)

Serangan pencurian kredensial, seperti *phishing* atau *keylogging*, seringkali berhasil mendapatkan kombinasi *username* dan *password* pengguna. Namun, dengan implementasi sistem ini, kepemilikan kredensial login saja tidak cukup untuk mendapatkan akses.

Sistem mewajibkan faktor kedua berupa kode OTP yang dikirimkan secara *real-time* ke perangkat terpercaya (*Trusted Device*). Sebagaimana dijelaskan dalam penelitian (Wahyudi & Santoso, 2024), metode verifikasi berbasis kepemilikan perangkat (*possession factor*) terbukti mampu menurunkan risiko pembajakan akun hingga 99,9% dibandingkan metode *password-only*. Penyerang yang memiliki *password* korban tetap akan tertahan di halaman verifikasi karena tidak memiliki akses fisik ke *browser* atau perangkat yang menerima notifikasi Web Push.

b. Analisis Anti-Replay Attack

Serangan *Replay Attack* terjadi ketika penyerang menyadap paket data berisi kode OTP yang valid dan mencoba menggunakannya kembali di waktu yang berbeda. Sistem ini mengatasi ancaman tersebut melalui dua lapisan pertahanan:

1) Time-Based Validity: Kode OTP hanya valid dalam jendela waktu 30 detik (T_0). Setelah durasi tersebut, algoritma di server akan menghasilkan nilai *hash* baru, sehingga kode lama otomatis menjadi sampah digital yang tidak valid.
2) One-Time Use constraint: Sistem mencatat status penggunaan token. Jika sebuah kode berhasil digunakan untuk login, statusnya di memori server langsung diubah menjadi *used* (digunakan).

Hasil pengujian pada Tabel 4.3 sebelumnya mengonfirmasi teori yang disampaikan oleh (Lestari, 2021), bahwa integrasi *timestamp* dan *nonce* (number used once) pada protokol HMAC adalah solusi paling efektif untuk meniadakan risiko serangan pengulangan paket data. Dengan demikian, integritas proses otentikasi tetap terjaga meskipun berada dalam jaringan yang tidak aman.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan seluruh tahapan penelitian mulai dari perancangan, implementasi, hingga pengujian sistem, dapat disimpulkan bahwa penerapan algoritma HMAC-SHA256 pada fitur otentikasi dua faktor berhasil berjalan sesuai dengan tujuan penelitian. Secara teknis, perhitungan manual yang dilakukan membuktikan bahwa logika *dynamic truncation* untuk mengambil sampel data 32-bit dari hasil *hash* mampu menghasilkan kode OTP 6 digit yang presisi dan sinkron antara sisi *client* dan *server*. Mekanisme pembangkitan kode ini valid karena selalu menghasilkan kombinasi unik yang bergantung pada kunci rahasia pengguna dan interval waktu server yang berubah setiap 30 detik.

Ditinjau dari aspek fungsionalitas, pengujian *Black Box* yang dilakukan dengan metode iterasi sebanyak 10 kali percobaan pada setiap skenario menunjukkan tingkat keberhasilan mencapai 100%. Seluruh fitur utama, mulai dari proses *login*, pengiriman notifikasi *Web Push*, hingga validasi token pada perangkat terpercaya (*Trusted Device*), berfungsi dengan optimal tanpa ditemukan adanya kegagalan sistem. Hal ini mengindikasikan bahwa integrasi antara *back-end* PHP dan *Service Worker* pada peramban pengguna telah berjalan stabil dalam menangani permintaan otentikasi secara *real-time*.

Selain itu, dari segi keamanan sistem, hasil pengujian keamanan (*Security Testing*) membuktikan bahwa sistem memiliki ketahanan yang solid terhadap berbagai ancaman siber. Sistem terbukti mampu memitigasi serangan *SQL Injection*, *Cross-Site Scripting* (XSS), dan *Brute Force* dengan persentase keberhasilan blokir 100%. Lebih lanjut, penerapan validasi waktu dan mekanisme

nonce pada token juga efektif dalam mencegah serangan pengulangan (*Replay Attack*), di mana sistem secara otomatis menolak kode OTP yang sudah kadaluwarsa atau yang telah digunakan sebelumnya, sehingga integritas data pengguna tetap terjaga.

5.2 Saran

Meskipun sistem otentikasi ini telah berhasil dibangun dan memenuhi standar keamanan yang diharapkan, penulis menyadari masih terdapat ruang untuk pengembangan lebih lanjut. Untuk penelitian atau pengembangan di masa mendatang, disarankan agar sistem menambahkan diversifikasi kanal pengiriman kode OTP. Selain menggunakan *Web Push Notification*, integrasi dengan layanan *WhatsApp Gateway* atau *Email* dapat dipertimbangkan sebagai opsi cadangan (*fallback*) untuk mengantisipasi kegagalan pengiriman saat pengguna mengalami gangguan koneksi pada peramban. Selain itu, adopsi standar *WebAuthn* untuk mendukung otentikasi biometrik seperti sidik jari juga dapat diterapkan guna meningkatkan kenyamanan pengguna.

Saran selanjutnya berkaitan dengan peningkatan infrastruktur keamanan. Untuk implementasi pada lingkungan produksi (*live server*), sangat disarankan untuk menerapkan protokol HTTPS (SSL/TLS) secara menyeluruh guna mengenkripsi lalu lintas data antara pengguna dan server, sehingga dapat mencegah potensi serangan *Man-in-the-Middle* (MitM). Terakhir, seiring dengan perkembangan kemampuan komputasi, penelitian selanjutnya dapat mempertimbangkan penggunaan algoritma *hashing* yang lebih kompleks dan memakan memori (memory-hard) seperti Argon2 atau SHA-512 untuk menggantikan SHA-256, guna memberikan lapisan keamanan ekstra terhadap ancaman komputasi masa depan.

DAFTAR PUSTAKA

Angkasa, B., Asriyanik, A., & Pambudi, A. (2025). Implementasi algoritma HMAC-SHA-256 untuk keamanan kemasan produk. *Jurnal Ilmiah Universitas Budi Luhur, 20*(2), 112–120.

Aziz, N. M. (2026). Penerapan teknik boundary value analysis dan equivalence partitioning pada pengujian sistem ujian berbasis komputer. *Jurnal Informatika dan Teknik Elektro Terapan, 14*(1), 45–52.

Hendra, H., Awan, A., Waisen, W., Wilianto, W., & Yudi, Y. (2025). Memperkuat autentikasi dan integritas data REST-API menggunakan token HMAC SHA-256. *Jurnal Minfo Polgan, 13*(2), 2189–2197.

Hidayat, R. (2024). Standarisasi pengujian perangkat lunak pada aplikasi fintech. *Jurnal Rekayasa Perangkat Lunak Indonesia, 10*(2), 112–120.

Kim, H., Han, J., Park, C., & Yi, O. (2020). Analysis of vulnerabilities that can occur when generating one-time password. *Applied Sciences, 10*(8), 2961. https://doi.org/10.3390/APP10082961

Kurniawan, D., Saputra, A., & Budi, S. (2024). Analisis vulnerability assessment pada sistem informasi akademik. *Jurnal Teknik Informatika dan Sistem Informasi, 9*(1), 45–55.

Laudon, K. C., & Laudon, J. P. (2020). *Management information systems: Managing the digital firm*. Pearson.

Lestari, D. (2021). Implementasi Time-Based One-Time Password (TOTP) untuk mencegah replay attack. *Jurnal Informatika dan Rekayasa Perangkat Lunak, 3*(2), 150–160.

Ma, S., Li, J., Kim, H., Bertino, E., Nepal, S., Ostry, D., & Sun, C. (2021). Fine with "1234"? An analysis of SMS one-time password randomness in Android apps. *arXiv*. https://doi.org/10.48550/arXiv.2103.05758

Martiano, M., & Sary, Y. (2022). Cryptography generator for prevention SQL injection attack in big data. *Journal of Computer Science, Information Technology and Telecommunication Engineering, 3*(2), 292–298.

Mayorga, O. E. A., & Yoo, S. G. (2025). One time password (OTP) solution for two factor authentication: A practical case study. *Journal of Computer Science, 21*(5), 1099–1112. https://doi.org/10.3844/jcssp.2025.1099.1112

M'Raihi, D., Machani, S., Pei, M., & Rydell, J. (2011). *TOTP: Time-based one-time password algorithm (RFC 6238)*. Internet Engineering Task Force. https://doi.org/10.17487/RFC6238

Nugraha, A., Santoso, B., & Wijaya, K. (2022). Komparasi kinerja algoritma SHA-256 dan MD5 dalam integritas data. *Jurnal Sistem Informasi dan Komputer, 7*(3), 200–210.

Patel, N., Williams, B., & Johnson, E. (2020). User perception of notification latency and its impact on application engagement. *International Journal of Human-Computer Studies, 144*, 102498.

Pratama, A., & Wibowo, S. (2022). Implementasi metode black box testing pada sistem informasi manajemen berbasis web. *Jurnal Teknologi Informasi dan Ilmu Komputer, 9*(4), 780–788.

Rahardjo, B., & Putra, A. (2023). Evaluasi keamanan website menggunakan metode OWASP top 10. *Jurnal Keamanan Siber Indonesia, 6*(1), 12–25.

Ramdani, F. C., Rahmatulloh, A., & Shofa, R. N. (2022). Implementasi JSON web token pada authentication dengan algoritma HMAC SHA-256. *Jurnal Sistem Informasi (SISTEMASI), 11*(1), 15–22.

Santoso, B., Hartono, R., & Putri, D. (2023). Analisis keamanan sistem otentikasi menggunakan two-factor authentication. *Jurnal Sistem Komputer dan Kecerdasan Buatan, 5*(2), 88–95.

Setiawan, I., & Arifin, Z. (2023). Analisis kriptografi modern untuk keamanan transaksi digital. *Jurnal Teknologi Informasi, 15*(1), 30–42.

Sofian, A. B., Peradus, A. F. A. B., Yong, F., Shearer, I., Ismail, N. N. B., Mahendran, Y., & Faisal, M. (2024). Enhancing authentication security: Analyzing time-based one-time password systems. *International Journal of Computer Technology and Science, 1*(3), 56–70. https://doi.org/10.62951/ijcts.v1i3.25

Subramanian, M. (2019). *Web push notifications: A complete guide for developers*. Apress.

Sultansyah, A., Rahayu, A. S., Yudiana, I., & Nugraha, F. (2025). Pengujian black box testing pada fitur permohonan informasi publik melalui website pemerintah Jawa Barat. *Jurnal Pengabdian Masyarakat Dan Riset Pendidikan, 3*(4), 5912–5919.

Susanto, H. (2022). Penerapan prepared statement untuk mencegah SQL injection pada aplikasi e-commerce. *Jurnal Algoritma, 19*(1), 50–59.

Wahyudi, E., & Santoso, B. (2024). Efektivitas multi-factor authentication dalam mencegah serangan phishing. *Jurnal Keamanan Siber Indonesia, 7*(1), 55–65.

Wijaya, K. (2023). Penerapan algoritma HMAC-SHA256 untuk keamanan transaksi online. *Jurnal Cyber Security Indonesia, 4*(2), 101–110.

Zen, M., Irwan, I., Hafni, H., & Ananda, M. D. P. (2024). Implementasi dan pengujian menggunakan metode blackbox testing pada sistem informasi tracer study. *Bulletin of Computer Science Research, 4*(4), 327–340.

LAMPIRAN

1.Surat Keterangan Penetapan Dosen Pembimbing

MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UMSU Terakreditasi Unggul Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 174/SK/BAN-PT/Ak.Ppj/PT/III/2024
Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003
fikti@umsu.ac.id umsumedan umsumedan umsumedan umsumedan

PENETAPAN DOSEN PEMBIMBING
PROPOSAL/SKRIPSI MAHASISWA
NOMOR : 1037/II.3-AU/UMSU-09/F/2025

Assalamu'alaikum Warahmatullahi Wabarakatuh

Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara, berdasarkan Persetujuan permohonan judul penelitian Proposal / Skripsi dari Ketua / Sekretaris.

Program Studi : **Sistem Informasi**
Pada tanggal : **06 November 2025**

Dengan ini menetapkan Dosen Pembimbing Proposal / Skripsi Mahasiswa.

Nama : **Alvin Lie**
NPM : **2209010168**
Semester : **VII (Tujuh)**
Program studi : **Sistem Informasi**
Judul Proposal / Skripsi : **Perancangan Sistem Informasi Geografis untuk Pemilihan Jalur Terpendek Pengiriman Barang Menggunakan Metode Dijkstra (Ganti Studi Kasus Perusahaan Logistik/Ekspedisi)**

Dosen Pembimbing : **Martiano, M.Kom**

Dengan demikian di izinkan menulis Proposal / Skripsi dengan ketentuan

1. Penulisan berpedoman pada buku panduan penulisan Proposal / Skripsi Fakultas Ilmu Komputer dan Teknologi Informasi UMSU
2. Pelaksanaan Sidang Skripsi harus berjarak 3 bulan setelah dikeluarkannya Surat Penetapan Dosen Pembimbing Skripsi.
3. **Proyek Proposal / Skripsi** dinyatakan " **BATAL** " bila tidak selesai sebelum Masa Kadaluarsa tanggal : **06 November 2026**
4. Revisi judul……

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

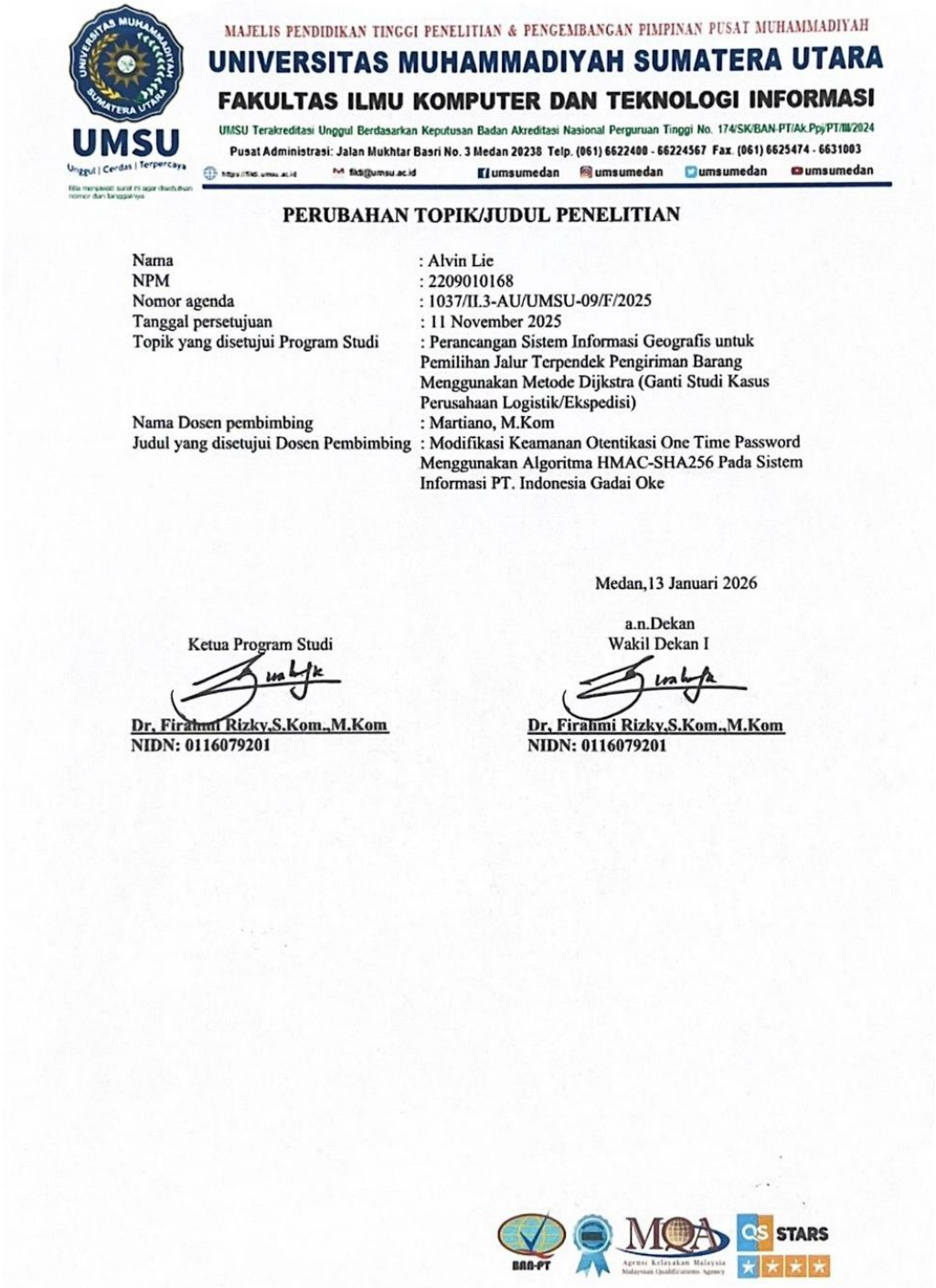
Ditetapkan di : Medan
Pada Tanggal : 15 Jumadil Awwal 1447 H
06 November 2025 M

a.n. Dekan

Dr. Al-Khowarizmi, M.Kom.
NIDN : 0127099201

Cc. File

2.Surat Perubahan Topik/Judul Peneletian



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UMSU Terakreditasi Unggul Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 174/SK/BAN-PT/Ak.Ppj/PT/III/2024
Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003
https://fikti.umsu.ac.id fikti@umsu.ac.id umsumedan umsumedan umsumedan umsumedan

UMSU
Unggul | Cerdas | Terpercaya

PERUBAHAN TOPIK/JUDUL PENELITIAN

Nama	: Alvin Lie
NPM	: 2209010168
Nomor agenda	: 1037/II.3-AU/UMSU-09/F/2025
Tanggal persetujuan	: 11 November 2025
Topik yang disetujui Program Studi	: Perancangan Sistem Informasi Geografis untuk Pemilihan Jalur Terpendek Pengiriman Barang Menggunakan Metode Dijkstra (Ganti Studi Kasus Perusahaan Logistik/Ekspedisi)
Nama Dosen pembimbing	: Martiano, M.Kom
Judul yang disetujui Dosen Pembimbing	: Modifikasi Keamanan Otentikasi One Time Password Menggunakan Algoritma HMAC-SHA256 Pada Sistem Informasi PT. Indonesia Gadai Oke

Medan,13 Januari 2026

Ketua Program Studi

Dr, Firahmi Rizky,S.Kom.,M.Kom
NIDN: 0116079201

a.n.Dekan
Wakil Dekan I

Dr, Firahmi Rizky,S.Kom.,M.Kom
NIDN: 0116079201

3.Surat Izin Penelitian

MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UMSU Terakreditasi Unggul Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 174/SK/BAN-PT/Ak.Ppj/PT/III/2024
Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003
fiks@umsu.ac.id umsumedan umsumedan umsumedan umsumedan

Nomor : 376/II.3-AU/UMSU-09/F/2026 Medan, 16 Sya'ban 1447 H
Lampiran : - 05 Februari 2026 M
Perihal : **IZIN PENELITIAN**

Kepada Yth.
Bapak/Ibu Pimpinan
PT. INDONESIA GADAI OKE
GADAI OKE 8 Jl. Kapten Muslim Komp. Megacom No. C18-C19 & D1-D2, Dwi Kora, Kec. Medan Helvetia, Kota Medan, Sumatera Utara 20124

Di Tempat

Assalamu'alaikum Warahmatullahi Wabarakatuh

Dengan hormat, sehubungan mahasiswa kami akan menyelesaikan studi,untuk itu kami memohon kesediaan Bapak / Ibu untuk memberikan kesempatan pada mahasiswa kami melakukan riset di **Perusahaan / Instansi** yang Bapak / Ibu pimpin, guna untuk penyusunan skripsi yang merupakan salah satu persyaratan dalam menyelesaikan Program **Studi Strata Satu (S-1)**

Adapun Mahasiswa/i di Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara tersebut adalah:

Nama	**: Alvin Lie**
Npm	**: 2209010168**
Jurusan	**: Sistem Informasi**
Semester	**: VIII (Delapan)**
Judul	**: MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE**
Email	**: alvinlie1704@gmail.com**
Hp/Wa	**: 082168136725**

Demikianlah surat kami ini, atas perhatian dan kerjasama yang Bapak / Ibu berikan kami ucapkan terimaksih

Wassalamu 'alaikum Warahmatullahi Wabarakatuh

Dekan

Dr. Al-Khowarizmi,M.Kom.
NIDN: 0127099201

Cc.File

4.Surat Pernyataan Penyelesaikan Penelitian

SURAT PERNYATAAN PENYELESAIAN PENELITIAN
Nomor: PT-IGO/SE/1023/I/2025

Dengan Ini Menyatakan:

Nama	: Alvin Lie
NIM	2209010168
Program Studi	: Sistem Informasi
Fakultas / Universitas	: Fakultas Ilmu Komputer Dan Tekologi Informasi Universitas Muhammadiyah Sumatera Utara

Dengan ini menyatakan bahwa saya telah menyelesaikan seluruh rangkaian kegiatan penelitian dengan judul:

"MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE"

Telah menyelesaikan seluruh rangkaian kegiatan penelitian sesuai dengan prosedur akademik yang berlaku di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara Penelitian ini dilakukan di PT. Indonesia Gadai Oke dan telah memenuhi kelengkapan administrasi serta hasil akademik sesuai dengan ketentuan yang berlaku.

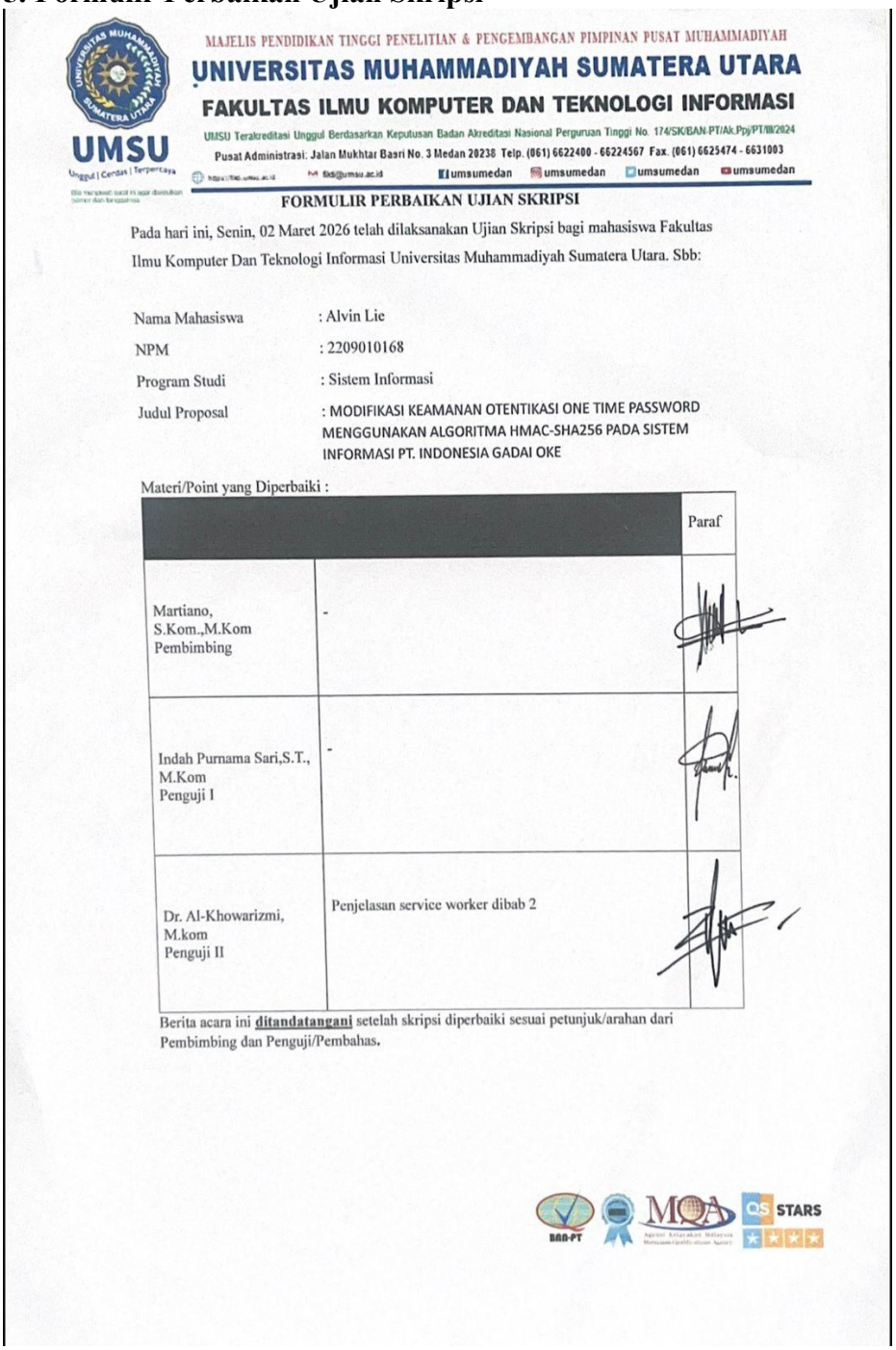
Surat ini dikeluarkan sebagai bentuk pernyataan resmi bahwa mahasiswa yang bersangkutan telah menyelesaikan penelitiannya dan dapat digunakan untuk keperluan akademik maupun administratif lainnya.

Demikian surat pernyataan ini dibuat untuk dipergunakan sebagaimana mestinya.

Medan, 16 Febuari 2026
Yang menyatakan,
Staf IT Indonesia Gadai OKe

Martiano, M.Kom

5. Formulir Perbaikan Ujian Skripsi



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi Unggul Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 174/SK/BAN-PT/Ak.Ppj/PT/III/2024

Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003

fikti@umsu.ac.id umsumedan umsumedan umsumedan umsumedan

UMSU Unggul | Cerdas | Terpercaya

FORMULIR PERBAIKAN UJIAN SKRIPSI

Pada hari ini, Senin, 02 Maret 2026 telah dilaksanakan Ujian Skripsi bagi mahasiswa Fakultas Ilmu Komputer Dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara. Sbb:

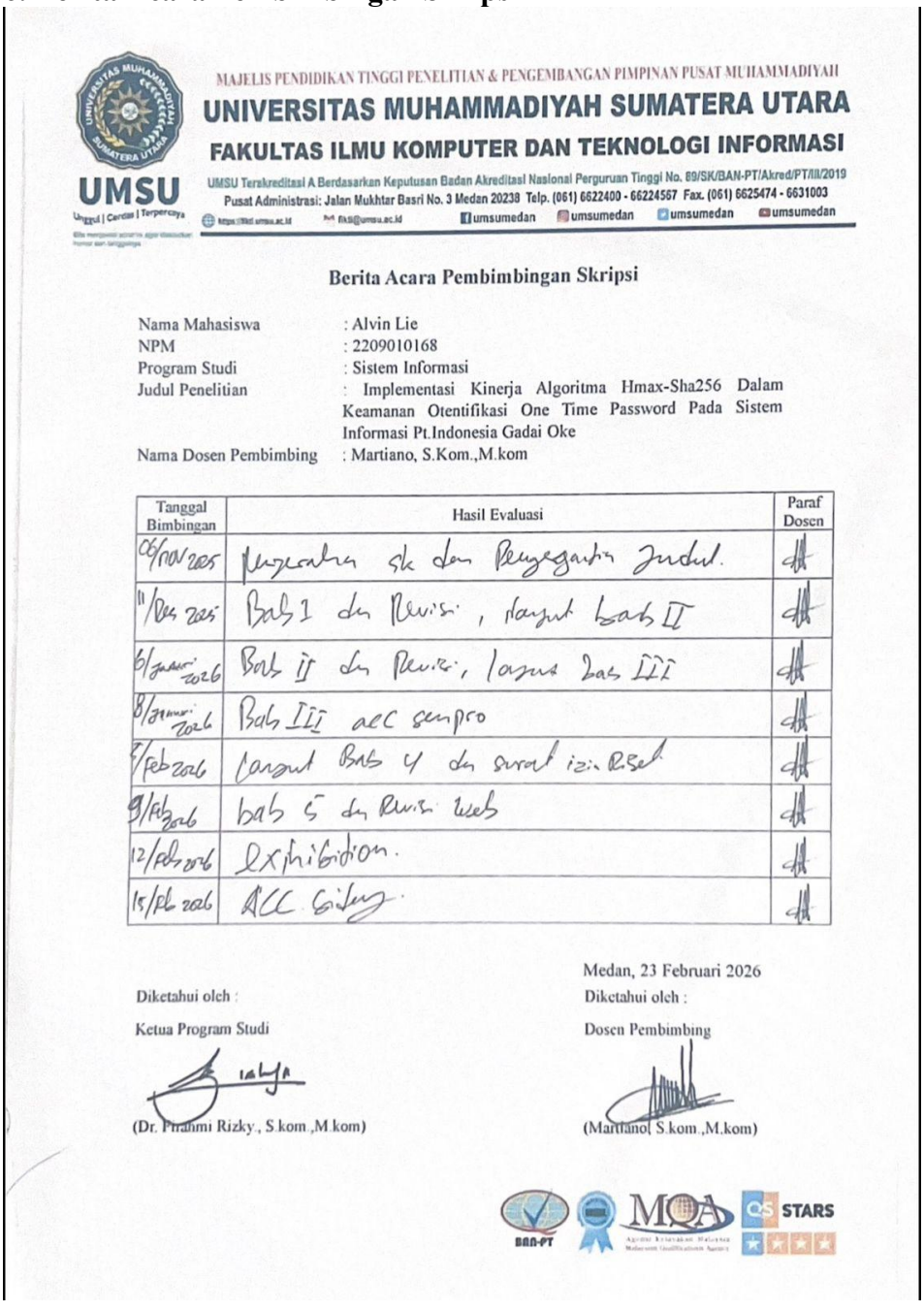
Nama Mahasiswa	: Alvin Lie
NPM	: 2209010168
Program Studi	: Sistem Informasi
Judul Proposal	: MODIFIKASI KEAMANAN OTENTIKASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA HMAC-SHA256 PADA SISTEM INFORMASI PT. INDONESIA GADAI OKE

Materi/Point yang Diperbaiki :

		Paraf
Martiano, S.Kom.,M.Kom Pembimbing	-	
Indah Purnama Sari,S.T., M.Kom Penguji I	-	
Dr. Al-Khowarizmi, M.kom Penguji II	Penjelasan service worker dibab 2	

Berita acara ini **ditandatangani** setelah skripsi diperbaiki sesuai petunjuk/arahan dari Pembimbing dan Penguji/Pembahas.

6. Berita Acara Pembimbingan Skripsi



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAN-PT/Akred/PT/II/2019
Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003
https://fikti.umsu.ac.id fikti@umsu.ac.id umsumedan umsumedan umsumedan umsumedan

Berita Acara Pembimbingan Skripsi

Nama Mahasiswa : Alvin Lie
NPM : 2209010168
Program Studi : Sistem Informasi
Judul Penelitian : Implementasi Kinerja Algoritma Hmax-Sha256 Dalam Keamanan Otentifikasi One Time Password Pada Sistem Informasi Pt.Indonesia Gadai Oke
Nama Dosen Pembimbing : Martiano, S.Kom.,M.kom

Tanggal Bimbingan	Hasil Evaluasi	Paraf Dosen
05/Nov 2025	Pengecekan sk dan Pengesahan Judul.	
11/Des 2025	Bab 1 dan Revisi, lanjut bab II	
6/januari 2026	Bab II dan Revisi, lanjut bab III	
8/januari 2026	Bab III acc sempro	
5/feb 2026	Lanjut Bab 4 dan surat izin Riset	
9/feb 2026	bab 5 dan Revisi web	
12/feb 2026	Exhibition.	
15/feb 2026	ACC Sidang.	

Medan, 23 Februari 2026

Diketahui oleh :
Ketua Program Studi

(Dr. Fatahmi Rizky., S.kom.,M.kom)

Diketahui oleh :
Dosen Pembimbing

(Martiano, S.kom.,M.kom)

7. Surat Penerimaan Naskah Publikasi Artikel Ilmiah

Jurnal Ilmu Komputer dan Sistem Informasi
JIRSI
Sinta 4 Terakreditasi
Homepage : https://www.jurnal.unity-academy.sch.id/index.php/jirsi
Email : jirsi.jurnal@gmail.com

Medan, 26 April 2026
No : 004/ LoA/JIRSI-Vol.5 No.2/04/2026
Hal : Surat Penerimaan Naskah Publikasi Artikel Ilmiah

Kepada Yth:
Bapak/ Ibu Penulis (Author)
Alvin Lie; Martiano
Di
Tempat

Assalamu'alaikum Wr. Wb.
Dengan hormat

Tim Redaksi **Jurnal Ilmu Komputer dan Informatika (JIRSI) Unity Academy Research and Publication** telah menerima paper dengan judul:

"Modifikasi Keamanan Otentikasi OTP Menggunakan Algoritma HMAC-SHA256 Pada Sistem Informasi PT Indonesia Gadai Oke"

Berdasarkan hasil review, artikel tersebut dinyatakan ***DITERIMA*** untuk dipublikasikan di Jurnal "JIRSI" Volume 5, Nomor 2, Mei 2026. Kami akan mengirimkan softcopy edisi tersebut pada akhir bulan penerbitan ke email penulis. Artikel tersedia secara online di alamat https://jurnal.unity-academy.sch.id/index.php/jirsi/index .
Demikian surat ini kami sampaikan, atas perhatian dan kerjasamanya kami ucapkan terimakasih.

Unity Academy Research and Publication
Editor in Chief

Unity Academy

Muhammad Eka, ST., M.Kom

8. Turnitin

SKRIPSI ALVIN LIE 2209010168 FIX.docx

ORIGINALITY REPORT

9%	8%	4%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	docplayer.info Internet Source	1%
2	text-id.123dok.com Internet Source	<1%
3	positivityloop.com Internet Source	<1%
4	repository.its.ac.id Internet Source	<1%
5	repository.upstegal.ac.id Internet Source	<1%
6	dspace.uii.ac.id Internet Source	<1%
7	repository.umsu.ac.id Internet Source	<1%
8	qwords.com Internet Source	<1%
9	123dok.com Internet Source	<1%
10	Arfan Sansprayada, Riva Abdillah Aziz, Kartika Mariskhana, Ita Dewi Sintawati. "Implementasi Internal System Menggunakan Framework CI Studi Kasus PT Ampera Abadi", Jurnal Minfo Polgan, 2025 Publication	<1%