

Kombinasi Algoritma Huffman dan *Rivest Shamir Adleman* Untuk Optimalisasi Kapasitas Informasi Pada *Quick Response Code*

Combination of Huffman and Rivest Shamir Adleman Algorithm to Optimize Information Capacity in Quick Response Code

Muliyadi¹, Rahmat Karim²

^{1,2}Program Studi Ilmu Komputer, Universitas Sembilanbelas November Kolaka
E-mail: ¹muliyadi@usn.ac.id, ²rahmatkarim.usn@gmail.com

Abstrak

Quick response code adalah matriks dua dimensi yang dapat menyimpan informasi dalam empat mode pengkodean berbeda. QR Code menawarkan kemudahan dan kecepatan dalam menyimpan dan mengakses informasi sehingga dalam implementasinya terbatas pada penyematan informasi singkat seperti kode produk, tautan website ataupun kode pembayaran. Pada perkembangannya, QR Code seharusnya dapat mengubah kebiasaan manusia dalam menyimpan informasi akan tetapi hal ini sulit dilakukan karena QR Code memiliki batasan karakter bergantung pada metode pengkodean. Oleh karena itu tujuan dari penelitian ini adalah untuk mengoptimalkan kapasitas informasi pada QR Code dengan mengkombinasikan algoritma Huffman dan metode Rivest Shamir Adleman. Hasil penelitian menunjukkan bahwa metode yang diusulkan yaitu *flowchart QR Code Generator* dan *QR Code Reader* dapat mengoptimalkan kapasitas informasi pada QR Code sebesar 81%. Hal ini menunjukkan bahwa QR Code versi 40 dengan level L yang memiliki batasan 4296 karakter alfanumerik, dapat dioptimalkan sampai dengan 7775 karakter alfanumerik. Oleh karena itu kesimpulan dari penelitian ini adalah metode yang diusulkan yaitu *flowchart QR Code Generator* dan *QR Code Reader* tidak hanya dapat mengoptimalkan kapasitas informasi pada QR Code tapi juga dapat mengamankan informasi yang disematkan pada QR Code dengan tidak mengorbankan kecepatan pemrosesan saat pemindaian.

Kata kunci: Optimalisasi, Kapasitas QR Code, Kompresi Huffman, Enkripsi RSA

Abstract

A quick response code is a two-dimensional matrix storing information in four distinct encoding modes. QR Codes offer convenience and speed in storing and accessing information, so their implementation is limited to embedding short information such as product codes, website links or payment codes. In its development, QR Codes should be able to change human habits in storing information. However, this is difficult because QR Codes have character limits depending on the encoding mode. Therefore, this research aims to optimize the information capacity of QR Codes by combining the Huffman algorithm and the Rivest Shamir Adleman method. The results of this research show that the proposed method, namely the QR Code Generator and QR Code Reader flowchart, can optimize the information capacity of the QR Code by 81%. It shows that QR Code version 40 with level L, which has a limit of 4296 alphanumeric characters, can be optimized up to 7775 c characters. Therefore, this research concludes that the proposed method, namely the flowchart QR Code Generator and QR Code Reader, can not only optimize the information capacity of the QR Code but can also secure the information embedded in the QR Code without sacrificing processing speed when scanning.

Keywords: Optimization, QR Code Capacity, Huffman Compression, RSA Encryption

1. PENDAHULUAN

Quick response code (QR Code) adalah sebuah matriks dua dimensi atau kode dua dimensi yang berisi modul hitam dengan latar belakang putih atau bentuk warna lainnya yang dapat menyimpan informasi yang berbentuk numerik, alfanumerik, biner ataupun kanji. QR Code pertama kali dikembangkan oleh Denso Wave pada tahun 1994 untuk melacak inventaris suku cadang mobil hasil pabrikan Toyota. Sejak diperkenalkan secara publik pada tahun 1999, QR Code telah banyak diimplementasikan dalam berbagai bidang seperti pemasaran [1], pembayaran [2], reservasi tiket [3], identifikasi produk [4] dan jejaring sosial. Pemanfaatan yang masif ini disebabkan oleh kemampuan QR Code yang dapat menyimpan informasi dan dapat dibaca kembali dengan cepat dan mudah menggunakan pemindai yang terpasang pada *smartphone*.

Secara umum, QR Code dimanfaatkan untuk menyimpan informasi dalam bentuk text yang memiliki jumlah karakter terbatas seperti url, kartu nama, kontak, email, kode produk dan lain-lain. Dalam implementasinya, informasi terbatas yang memanfaatkan QR Code dapat ditemukan pada aplikasi sistem pembayaran online [5][6][7] ataupun reservasi tiket perjalanan [3][8][9]. Implementasi QR Code secara luas juga menyisakan sejumlah masalah baru yang dapat dialami dalam penggunaannya. Masalah-masalah tersebut diantaranya yaitu keterbatasan dalam memasukan informasi [10], kesalahan dalam memindai kembali informasi [11], peretasan informasi secara ilegal [12], ketergantungan pada teknologi dan ancaman pada privasi. Oleh karena itu banyak penelitian yang telah dilakukan dengan berbagai sudut pandang untuk meningkatkan fitur – fitur dari QR Code itu sendiri.

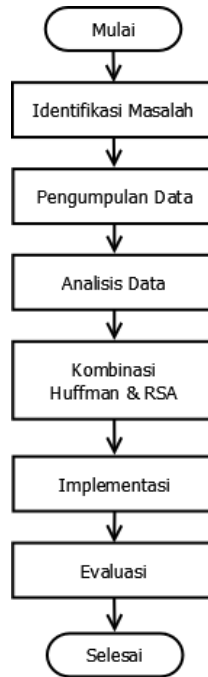
Pada perkembangannya, QR Code seharusnya dapat mengubah kebiasaan manusia dalam menyimpan informasi jika setiap informasi yang berbentuk text dapat dengan bebas disematkan dalam QR Code. Akan tetapi hal ini sulit dilakukan karena QR Code memiliki batasan karakter bergantung level dan versi dari QR Code itu sendiri. QR Code versi 40 dengan level L memiliki batasan yaitu 7089 karakter untuk numerik, 4296 karakter untuk alfanumerik, 2953 karakter untuk biner, dan 1817 untuk kanji [13]. Artinya informasi informasi yang disematkan pada QR Code tidak boleh melebihi batasan maksimum. Selain itu juga semakin banyak jumlah karakter yang disematkan maka akan semakin kompleks bentuk dari QR Code itu sendiri [13]. Semakin kompleks bentuk QR Code, maka semakin sulit juga alat pemindai dalam membaca kembali QR Code [11][14].

Beberapa hal yang dapat dilakukan untuk mengoptimalkan penyimpanan informasi dalam QR Code adalah dengan melakukan transformasi pada informasi itu sendiri. Hidayat pada tahun 2022 menggunakan metode kompresi Huffman untuk meningkatkan jumlah data atau informasi pada QR Code. Penggunaan metode Huffman yang melakukan kompresi dengan teknik *lossless* terbukti dapat meningkatkan ruang penyimpanan QR Code dimana informasi dapat dikompresi sampai 57% [15]. Sementara pada tahun 2023, Cheshtaa bhardwaj menggunakan teknik kompresi ZIP untuk meningkatkan kapasitas penyimpanan data pada QR Code. Metode kompresi ZIP memadukan antara algoritma LZ77 dan Huffman dan terbukti dapat meningkatkan ruang penyimpanan QR Code tanpa kehilangan informasi dalam bentuk data terkompresi [16]. Kedua pendekatan di atas dilaporkan dapat meningkatkan kapasitas penyimpanan informasi yang diikuti dengan peningkatan ukuran file pada QR Code. Selain itu juga kedua pendekatan di atas menggunakan dataset yang berbeda sehingga perlu ada justifikasi lebih lanjut.

Oleh karena itu pada penelitian ini, kami mengkombinasikan algoritma kompresi *Huffman* dan enkripsi *Rivest Shamir Adleman* (RSA) untuk mengoptimalkan informasi dalam bentuk text pada QR Code. Penelitian ini menghasilkan rancangan dalam bentuk *flowchart QR Code Generator* dan QR Code Reader sehingga informasi dalam bentuk text yang memiliki jumlah karakter di atas batas maksimum QR Code tetap dapat disematkan dalam QR Code dengan tidak merubah bentuk kompleksitas dari QR Code itu sendiri.

2. METODE PENELITIAN

Penelitian ini dilakukan dengan tujuan untuk mengoptimalkan kapasitas informasi berupa text pada QR Code. Adapun tahapan dari penelitian ini yaitu identifikasi permasalahan, pengumpulan data primer dan sekunder, analisis data, kombinasi algoritma Huffman dan RSA, implementasi algoritma hasil kombinasi, pengujian, dan evaluasi. Secara lebih jelas, tahapan penelitian dapat dilihat pada gambar 1 berikut.



Gambar 1. Alur Penelitian

2.1 Identifikasi masalah

Pada tahap identifikasi masalah, diperoleh informasi bahwa QR Code telah banyak digunakan diberbagai bidang termasuk pemasaran, keamanan dan pendidikan. Hal ini karena QR Code dapat menyimpan informasi dan dapat dipindai kembali dengan mudah. Akan tetapi QR Code memiliki keterbatasan pada jumlah informasi yang dapat disematkan dimana bentuk atau ukuran dari QR Code akan berbanding lurus dengan banyaknya informasi. Oleh karena itu diperlukan sebuah teknik yang dapat mengoptimalkan kapasitas informasi pada QR Code.

2.2 Pengumpulan & Analisis Data

Pengumpulan data baik data primer dan data sekunder dilakukan dengan tujuan untuk menemukan informasi yang dapat mendukung tercapainya tujuan penelitian. Pada tahap ini diperoleh informasi bahwa terdapat tiga variabel yang dapat mempengaruhi kapasitas informasi pada QR Code yaitu jenis, ukuran fisik dan tingkat pemampatan data.

Tahap analisis data dilakukan dengan tujuan untuk melihat peluang intervensi yang dapat dilakukan. Berdasarkan gambar 2 di bawah ini dapat diketahui bahwa semakin banyak informasi yang disematkan, maka akan semakin kompleks bentuk kotak-kotak dalam QR Code. Kompleksitas bentuk QR Code dapat mempengaruhi kemampuan alat pemindai dalam mengambil informasi [11][14]. Oleh karena itu variabel tingkat pemampatan data dapat dieksploitasi untuk mengoptimalkan kapasitas informasi pada QR Code tanpa merubah bentuk kompleksitas dari QR Code itu sendiri.



Gambar 2. Perubahan kompleksitas QR Code berdasarkan kapasitas informasi (jumlah karakter)

2.3 Kombinasi Algoritma *Huffman* dan *Rivest Shamir Adleman* (RSA)

Algoritma Huffman dan Rivest Shamir Adleman adalah algoritma-algoritma yang mendukung pemampatan data. Algoritma Huffman dapat melakukan kompresi pada informasi dengan cara menggantikan code yang lebih kecil pada karakter yang sering digunakan dan code yang lebih panjang pada karakter yang jarang digunakan [17]. Sedangkan algoritma *Rivest Shamir Adleman* (RSA) adalah algoritma yang dapat mengenkripsi informasi yang berukuran kecil melalui kunci publik dan kunci privat [18]. Oleh karena itu kedua algoritma tersebut akan dikombinasikan sehingga menghasilkan *flowchart QR Code Generator* dan *QR Code Reader* yang dapat mengoptimalkan kapasitas informasi dalam QR Code.

2.3.1 Algoritma Kompresi Huffman

Berikut ini adalah langkah-langkah dari algoritma Huffman:

1. Menghitung frekuensi kemunculan dari setiap karakter data yang akan dikompresi
2. Mengurutkan karakter-karakter tersebut berdasarkan frekuensi kemunculan dari yang paling sering muncul ke yang paling jarang muncul
3. Membuat pohon biner dengan karakter-karakter tersebut sebagai daunnya, dimana setiap simpul dalam pohon memiliki bobot yang sama dengan frekuensi kemunculan karakter yang sesuai.
4. Menggabungkan simpul-simpul dengan bobot terkecil menjadi simpul baru yang memiliki bobot yang sama dengan jumlah bobot simpul-simpul yang digabungkan. Proses ini diulangi sampai hanya tersisa satu simpul.
5. Memberikan kode biner 0 pada setiap anak simpul kiri dan kode biner 1 pada setiap anak simpul kanan.
6. Melakukan traverse pohon dari root hingga setiap daun untuk menentukan kode biner pada setiap karakter.

2.3.2 Algoritma Enkripsi Rivest Shamir Adleman (RSA)

Adapun langkah-langkah dari metode Rivest Shamir Adleman adalah sebagai berikut:

1. Memilih 2 bilangan prima yang cukup besar. Asumsikan a dan b .
2. Menghitung $c = a \times b$
3. Menghitung $\varphi(c) = (a - 1) \times (b - 1)$
4. Memilih sebuah bilangan misalkan d yang relatif prima dengan $\varphi(c)$ dan lebih kecil dari $\varphi(c)$. Sepasang bilangan (c, d) akan menjadi kunci publik untuk tujuan enkripsi

5. Menghitung e sebagai kebalikan dari modulo $\varphi(c)$, dimana e adalah bilangan yang memenuhi persamaan $d \times e \equiv 1(mod \varphi(c))$. Sepasang bilangan (c, e) akan menjadi kunci privat untuk tujuan dekripsi.
6. Untuk melakukan enkripsi pesan m , pesan m di konveri ke dalam bentuk bilangan bulat m yang umumnya menggunakan metode ASCII atau Unicode.
7. Menghitung $f = m^d mod c$. f adalah pesan yang telah dienkrpsi
8. Menghitung $m = f^e mod c$ jika ingin melakukan dekripsi pada pesan yang telah dilakukan enkripsi.

2.4 Implementasi dan Pengujian

Untuk melihat efektifitas metode yang dikembangkan yaitu kombinasi antara Algoritma Huffman dan *Rivest Shamir Adleman* dalam optimalisasi kapasitas informasi pada QR Code, metode yang diusulkan perlu diimplementasikan pada dataset tertentu. Penelitian ini menggunakan text dari beberapa abstrak di *repository.ugm.ac.id* dan beberapa artikel yang diasumsikan sebagai data atau informasi yang harus disematkan pada QR Code. Berikut adalah 12 dataset yang digunakan untuk mengimplementasikan metode yang diusulkan.

Tabel 1 Dataset penelitian

No	Text	Jumlah Karakter	Sumber	Keterangan Text
1	File01.txt	1880	https://etd.repository.ugm.ac.id/penelitian/detail/199445	Indonesia
2	File02.txt	1182	https://etd.repository.ugm.ac.id/penelitian/detail/223371	Indonesia
3	File03.txt	2510	https://etd.repository.ugm.ac.id/penelitian/detail/204255	Indonesia
4	File04.txt	1176	https://etd.repository.ugm.ac.id/penelitian/detail/86806	Indonesia
5	File05.txt	2375	https://etd.repository.ugm.ac.id/penelitian/detail/204255	Inggris
6	File06.txt	1094	https://etd.repository.ugm.ac.id/penelitian/detail/86806	Inggris
7	File07.txt	2119	https://etd.repository.ugm.ac.id/penelitian/detail/151492	Inggris
8	File08.txt	2775	https://marketinstitute.org/index.php/2023/08/02/sopa-restores-constitutional-limits-on-out-of-control-agencies/	Inggris
9	File09.txt	4092	https://uspolicy.org/sauer-congress-doing-what-it-should-do-defending-innovators/	Inggris
10	File10.txt	4886	https://etd.repository.ugm.ac.id/penelitian/detail/204255	Inggris & Indonesia
11	File11.txt	4396	https://etd.repository.ugm.ac.id/penelitian/detail/88187	Inggris & Indonesia
12	File12.txt	2274	https://etd.repository.ugm.ac.id/penelitian/detail/86806	Inggris & Indonesia

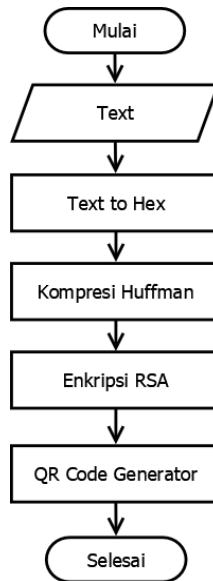
3. HASIL DAN PEMBAHASAN

QR Code memiliki batasan dalam menampung informasi. Batasan tersebut berupa jumlah karakter dari numerik, alfanumerik, biner ataupun kanji yang disesuaikan dengan versi dan tipe dari QR Code. Untuk mengatasi hal tersebut dilakukan kombinasi antara algoritma kompresi Huffman dan Enkripsi RSA. Hasil kombinasi kedua metode tersebut terdeskripsi dalam bentuk rancangan *flowchart QR Code Generator* dan *flowchart QR Code Reader*.

3.1 *Flowchart QR Code Generator* dan *QR Code Reader*

3.1.1 *Flowchart QR Code Generator*

Untuk dapat mengoptimalkan kapasitas informasi dalam QR Code, informasi diubah sedemikian sehingga mengalami penyusutan dalam jumlah karakter. Berikut adalah *flowchart QR Code Generator* yang memanfaatkan metode Kompresi Huffman dan Enkripsi RSA.

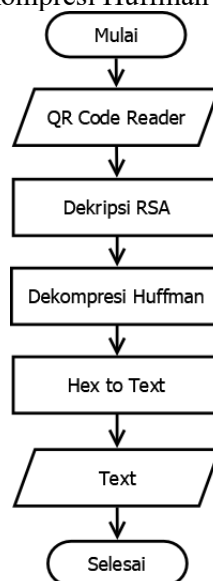


Gambar 3. Flowchart QR Code Generator

Berdasarkan *flowchart QR Code Generator* yang diusulkan pada gambar 3 di atas, dapat diketahui bahwa text akan terlebih dahulu diubah ke dalam bentuk hex, kemudian dilakukan kompresi dengan metode Huffman. Hasil Kompresi kemudian dilakukan enkripsi dengan menggunakan Metode RSA. Tahap akhir adalah QR Code Generator yang menampilkan informasi yang telah dimampatkan dalam bentuk QR Code. Dengan melalui tahapan ini maka QR Code tidak hanya dapat menampung lebih banyak data atau informasi akan tetapi juga sekaligus dapat mengamankan data atau informasi yang disematkan.

3.1.2 Flowchart QR Code Reader

Informasi yang telah termampatkan dalam bentuk QR Code, tidak langsung dapat dibaca atau diterima oleh user. Oleh karena itu diperlukan QR Code Reader yang dapat mengembalikan informasi yang telah termampatkan tersebut. Berikut adalah *flowchart QR Code Reader* yang memanfaatkan metode Kompresi Huffman dan Enkripsi RSA.



Gambar 4. Flowchart QR Code Reader

Berdasarkan *flowchart QR Code Reader* pada gambar 4 di atas, dapat diketahui bahwa QR Code Reader akan melakukan *scanning* terlebih dahulu pada QR Code, kemudian dilakukan dekripsi dengan menggunakan Metode RSA. Setelah itu dilakukan dekompresi dengan metode Huffman sehingga data atau informasi dapat kembali ke dalam bentuk hex. Langkah selanjutnya adalah melakukan perubahan dari hex ke text sehingga informasi yang termampatkan ke dalam bentuk QR Code dapat diterima atau dibaca secara lengkap oleh user.

3.2 Implementasi *Flowchart QR Code Generator* dan QR Code Reader pada text

Flowchart QR Code Generator yang memanfaatkan metode kompresi Huffman dan enkripsi RSA dapat mengoptimalkan kapasitas informasi pada QR Code yang memiliki batasan pada jumlah karakter alfanumerik. Berikut adalah contoh implementasi *flowchart QR Code Generator* dalam mengoptimalkan kapasitas informasi pada QR Code.

- Langkah 1: input text

Pemilihan bidang keahlian selama pelaksanaan Program Latihan Profesi (PLP) yang tepat untuk mahasiswa Progran Studi Pendidikan Ilmu Komputer Universitas Pendidikan Indonesia merupakan suatu hal yang sangat penting karena dapat memaksimalkan kemampuan mahasiswa. Dalam pemilihan bidang keahlian PLP untuk mahasiswa, koordinator PLP sering kali mengalami kesulitan untuk menentukan bidang keahlian yang sesuai dengan kemampuan yang dimiliki mahasiswa sehingga mahasiswa tidak bisa mengaplikasikan kemampuan yang dimilikinya secara optimal. Untuk mengatasi masalah tersebut diperlukan suatu sistem komputer berupa sistem pendukung keputusan yang dapat membantu menentukan kesesuaian mahasiswa yang dibutuhkan oleh bidang keahlian PLP. Penelitian ini bertujuan membangun sebuah sistem untuk mendukung keputusan pemilihan bidang keahlian PLP mahasiswa dengan menggunakan metode Analytical Hierarchy Process (AHP); metode ini digunakan untuk penentuan bobot kriteria dan sub kriteria dari setiap alternatif bidang keahlian PLP. Berdasarkan hasil akhir dari sistem pendukung keputusan pemilihan bidang keahlian ini setiap mahasiswa hanya mendapatkan satu rekomendasi bidang keahlian PLP dari perangkaan akhir dimana bidang keahlian yang mendapatkan hasil nilai akhir tertinggi dijadikan sebagai rekomendasi bidang keahlian untuk dipilih oleh mahasiswa dalam pelaksanaan PLP. Penggunaan metode AHP pada sistem sangat efisien dikarenakan sesuai dengan kondisi lapangan yaitu mempertimbangkan parameter Ketrampilan mahasiswa dalam pemilihan bidang keahlian PLP. Disamping itu dilakukan juga pengujian usability terhadap sistem yang dikembangkan dengan menggunakan elemen observasi efektivitas, efisiensi, dan kepuasan. Pengujian usability sistem dilakukan terhadap Tim Koordinator PLP dan diperoleh rerata tingkat usability sistem.

- Langkah 2: Transformasi text ke dalam bentuk hex (ASCII)

Source code untuk melakukan transformasi text ke dalam bentuk hex dapat dilihat pada tautan berikut: [text2hexfunction](#).

```
50 65 6D 69 6C 69 68 61 6E 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 73 65 6C 61 6D 61 20 70 65 6C 61 6B
73 61 6E 61 61 6E 20 50 72 6F 67 72 61 6D 20 4C 61 74 69 68 61 6E 20 50 72 6F 66 65 73 69 20 28 50 4C 50 29 20 79
61 6E 67 20 74 65 70 61 74 20 75 6E 74 75 6B 20 6D 61 68 61 73 69 73 77 61 20 50 72 6F 67 72 61 6E 20 53 74 75 64
69 20 50 65 6E 64 69 64 69 6B 61 6E 20 49 6C 6D 75 20 4B 6F 6D 70 75 74 65 72 20 55 6E 69 76 65 72 73 69 74 61 73
20 50 65 6E 64 69 64 69 6B 61 6E 20 49 6E 64 6F 6E 65 73 69 61 20 6D 65 72 75 70 61 6B 61 6E 20 73 75 61 74 75 20
68 61 6C 20 79 61 6E 67 20 73 61 6E 67 61 74 20 70 65 6E 74 69 6E 67 20 6B 61 72 65 6E 61 20 64 61 70 61 74 20 6D
65 6D 61 6B 73 69 6D 61 6C 6B 61 6E 20 6B 65 6D 61 6D 70 75 61 6E 20 6D 61 68 61 73 69 73 77 61 2E 20 44 61 6C
61 6D 20 70 65 6D 69 6C 69 68 61 6E 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 50 4C 50 20 75 6E 74 75 6B
20 6D 61 68 61 73 69 73 77 61 2C 20 6B 6F 6F 72 64 69 6E 61 74 6F 72 20 50 4C 50 20 73 65 72 69 6E 67 20 6B 61 6C
69 20 6D 65 6E 67 61 6C 61 6D 69 20 6B 65 73 75 6C 69 74 61 6E 20 75 6E 74 75 6B 20 6D 65 6E 65 6E 74 75 6B 61
6E 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 79 61 6E 67 20 73 65 73 75 61 69 20 64 65 6E 67 61 6E 20 6B
65 6D 61 6D 70 75 61 6E 20 79 61 6E 67 20 64 69 6D 69 6C 69 6B 69 20 6D 61 68 61 73 69 73 77 61 20 73 65 68 69 6E
67 67 61 20 6D 61 68 61 73 69 73 77 61 20 74 69 64 61 6B 20 62 69 73 61 20 6D 65 6E 67 61 70 6C 69 6B 61 73 69 6B
61 6E 20 6B 65 6D 61 6D 70 75 61 6E 20 79 61 6E 67 20 64 69 6D 69 6C 69 6B 69 6E 79 61 20 73 65 63 61 72 61 20 6F
70 74 69 6D 61 6C 2E 20 55 6E 74 75 6B 20 6D 65 6E 67 61 74 61 73 69 20 6D 61 73 61 6C 61 68 20 74 65 72 73 65 62
75 74 20 64 69 70 65 72 6C 75 6B 61 6E 20 73 75 61 74 75 20 73 69 73 74 65 6D 20 6B 6F 6D 70 75 74 65 72 20 62 65
72 75 70 61 20 73 69 73 74 65 6D 20 70 65 6E 64 75 6B 75 6E 67 20 6B 65 70 75 74 75 73 61 6E 20 79 61 6E 67 20 64
61 70 61 74 20 6D 65 6D 62 61 6E 74 75 20 6D 65 6E 65 6E 74 75 6B 61 6E 20 6B 65 73 65 73 75 61 69 61 6E 20 6D 61
68 61 73 69 73 77 61 20 79 61 6E 67 20 64 69 62 75 74 75 68 6B 61 6E 20 6F 6C 65 68 20 62 69 64 61 6E 67 20 6B 65
61 68 6C 69 61 6E 20 50 4C 50 2E 20 50 65 6E 65 6C 69 74 69 61 6E 20 69 6E 69 20 62 65 72 74 75 6A 75 61 6E 20 6D
65 6D 62 61 6E 67 75 6E 20 73 65 62 75 61 68 20 73 69 73 74 65 6D 20 75 6E 74 75 6B 20 6D 65 6E 64 75 6B 75 6E 67
20 6B 65 70 75 74 75 73 61 6E 20 70 65 6D 69 6C 69 68 61 6E 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 50
4C 50 20 6D 61 68 61 73 69 73 77 61 20 64 65 6E 67 61 6E 20 6D 65 6E 67 67 75 6E 61 6B 61 6E 20 6D 65 74 6F 64 65
20 41 6E 61 6C 79 74 69 63 61 6C 20 48 69 65 72 61 72 63 68 79 20 50 72 6F 63 65 73 73 20 28 41 48 50 29 3B 20 6D
65 74 6F 64 65 20 69 6E 69 20 64 69 67 75 6E 61 6B 61 6E 20 75 6E 74 75 6B 20 70 65 6E 65 6E 74 75 61 6E 20 62 6F
62 6F 74 20 6B 72 69 74 65 72 69 61 20 64 61 6E 20 73 75 62 20 6B 72 69 74 65 72 69 61 20 64 61 72 69 20 73 65 74 69
61 70 20 61 6C 74 65 72 6E 61 74 69 66 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 50 4C 50 2E 20 42 65 72
```

```
64 61 73 61 72 6B 61 6E 20 68 61 73 69 6C 20 61 6B 68 69 72 20 64 61 72 69 20 73 69 73 74 65 6D 20 70 65 6E 64 75
6B 75 6E 67 20 6B 65 70 75 74 75 73 61 6E 20 70 65 6D 69 6C 69 68 61 6E 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69
61 6E 20 69 6E 69 20 73 65 74 69 61 70 20 6D 61 68 61 73 69 73 77 61 20 68 61 6E 79 61 20 6D 65 6E 64 61 70 61 74
6B 61 6E 20 73 61 74 75 20 72 65 6B 6F 6D 65 6E 64 61 73 69 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 50
4C 50 20 64 61 72 69 20 70 65 72 61 6E 67 6B 69 6E 67 61 6E 20 61 6B 68 69 72 20 64 69 6D 61 6E 61 20 62 69 64 61
6E 67 20 6B 65 61 68 6C 69 61 6E 20 79 61 6E 67 20 6D 65 6E 64 61 70 61 74 6B 61 6E 20 68 61 73 69 6C 20 6E 69 6C
61 69 20 61 6B 68 69 72 20 74 65 72 74 69 6E 67 67 69 20 64 69 6A 61 64 69 6B 61 6E 20 73 65 62 61 67 61 69 20 72
65 6B 6F 6D 65 6E 64 61 73 69 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 75 6E 74 75 6B 20 64 69 70 69 6C
69 68 20 6F 6C 65 68 20 6D 61 68 61 73 69 73 77 61 20 64 61 6C 61 6D 20 70 65 6C 61 6B 73 61 6E 61 61 6E 20 50 4C
50 2E 20 50 65 6E 67 67 75 6E 61 61 6E 20 6D 65 74 6F 64 65 20 41 48 50 20 70 61 64 61 20 73 69 73 74 65 6D 20 73
61 6E 67 61 74 20 65 66 69 73 69 65 6E 20 64 69 6B 61 72 65 6E 61 6B 61 6E 20 73 65 73 75 61 69 20 64 65 6E 67 61
6E 20 6B 6F 6E 64 69 73 69 20 6C 61 70 61 6E 67 61 6E 20 79 61 69 74 75 20 6D 65 6D 70 65 72 74 69 6D 62 61 6E 67
6B 61 6E 20 70 61 72 61 6D 65 74 65 72 20 4B 65 74 72 61 6D 70 69 6C 61 6E 20 6D 61 68 61 73 69 73 77 61 20 64 61
6C 61 6D 20 70 65 6D 69 6C 69 68 61 6E 20 62 69 64 61 6E 67 20 6B 65 61 68 6C 69 61 6E 20 50 4C 50 2E 20 44 69 73
61 6D 70 69 6E 67 20 69 74 75 20 64 69 6C 61 6B 75 6B 61 6E 20 6A 75 67 61 20 70 65 6E 67 75 6A 69 61 6E 20 75 73
61 62 69 6C 69 74 79 20 74 65 72 68 61 64 61 70 20 73 69 73 74 65 6D 20 79 61 6E 67 20 64 69 6B 65 6D 62 61 6E 67
6B 61 6E 20 64 65 6E 67 61 6E 20 6D 65 6E 67 67 75 6E 61 6B 61 6E 20 65 6C 65 6D 65 6E 20 6F 62 73 65 72 76 61 73
69 20 65 66 65 6B 74 69 76 69 74 61 73 2C 20 65 66 69 73 69 65 6E 73 69 2C 20 64 61 6E 20 6B 65 70 75 61 73 61 6E
2E 20 50 65 6E 67 75 6A 69 61 6E 20 75 73 61 62 69 6C 69 74 79 20 73 69 73 74 65 6D 20 64 69 6C 61 6B 75 6B 61 6E
20 74 65 72 68 61 64 61 70 20 54 69 6D 20 4B 6F 6F 72 64 69 6E 61 74 6F 72 20 50 4C 50 20 64 61 6E 20 64 69 70 65
72 6F 6C 65 68 20 72 65 72 61 74 61 20 74 69 6E 67 6B 61 74 20 75 73 61 62 69 6C 69 74 79 20 73 69 73 74 65 6D 2E
```

- Langkah 3: lakukan kompresi dengan menggunakan metode Huffman

Source code untuk melakukan untuk melakukan kompresi dengan metode Huffman dapat dilihat pada tautan berikut: [huffman](#).

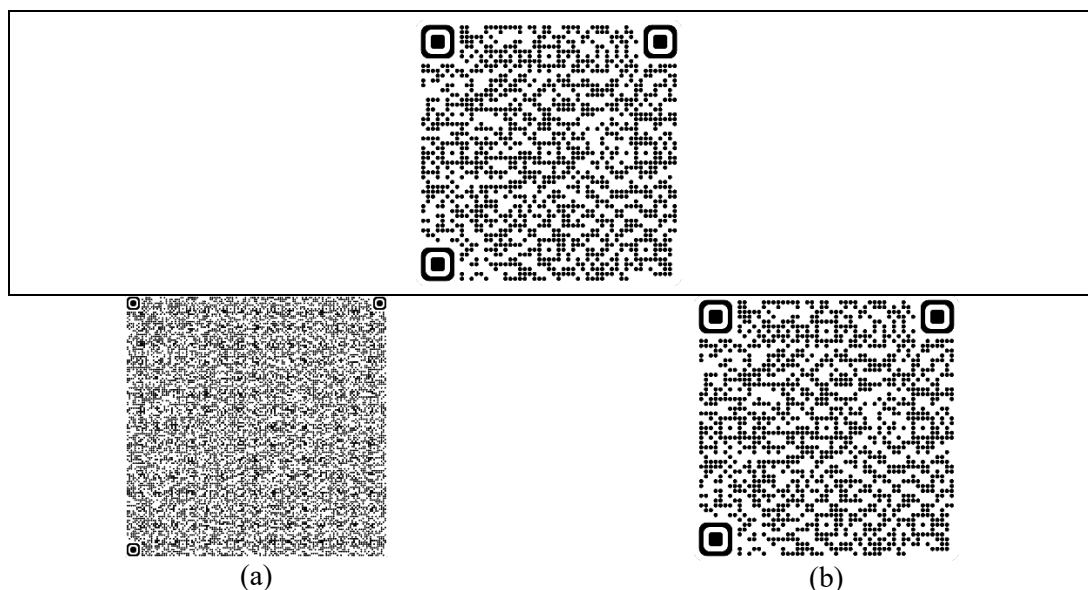
```
01 90 4E B8 DD 38 07 D5 B4 BB 94 E9 5B B9 20 90 4D 31 0C CA D2 85 63 12 39 55 88 28 93 48 75 67 8A 67 C9
86 B1 8C 50 E6 D1 CC AA 48 EF D6 B1 ED DA 34
```

- Langkah 4: lakukan enkripsi dengan menggunakan metode RSA

Source code untuk melakukan untuk melakukan enkripsi dengan metode RSA dapat dilihat pada tautan berikut: [rsa](#).

```
01 00 90 00 4E 00 2E 00 68 00 DD 00 0A 00 91 00 BE 00 13 00 A4 00 7D 00 E9 00 5B 00 8B 00 C1 00 90 00 4D 00 5F
00 0C 00 29 00 48 00 57 00 7A 00 26 01 22 00 DF 00 29 01 28 00 65 00 D2 00 75 00 67 00 A1 00 67 00 3F 00 58 00 C8
00 BA 00 F1 00 14 01 D1 00 FA 00 AA 00 D2 00 D8 00 4C 00 C8 00 BF 00 08 01 34
```

- Langkah 5: simpan hasil enkripsi ke dalam bentuk QR Code



Gambar 5. Perbedaan kompleksitas bentuk QR Code yang tidak menggunakan *flowchart QR Code Generator* (a) dan yang menggunakan *flowchart QR Code Generator* (b).

Berdasarkan gambar 5 di atas dapat diketahui bahwa terjadi perbedaan yang sangat mendasar antara text yang langsung disematkan ke dalam bentuk QR Code dengan text yang menggunakan *flowchart QR Code Generator*. Perbedaan tersebut dipengaruhi oleh banyaknya jumlah karakter yang tersematkan dalam QR Code. Secara mendasar gambar 5a dan 5b memuat informasi yang sama tetapi dengan jumlah karakter yang berbeda. Penyusutan jumlah karakter ini mengkonfirmasi bahwa kapasitas informasi pada QR Code yang memiliki batasan dalam jumlah karakter alfanumerik dapat dioptimalkan dengan menggunakan *flowchart QR Code Generator* yang diusulkan pada penelitian ini. Akan tetapi QR Code hasil *flowchart QR Code Generator* tidak dapat langsung dibaca oleh user karena informasi yang disematkan telah bertransformasi. Oleh karena itu diperlukan *flowchart QR Code Reader* (gambar 3) agar informasi pada gambar 5b dapat dibaca atau tersampaikan dengan baik kepada user.

3.3 Pengujian *Flowchart QR Code Generator* dan QR Code Reader pada text

QR Code Generator yang memanfaatkan metode kompresi Huffman dan Enkripsi RSA terkonfirmasi dapat diterapkan pada data atau informasi yang berbasis text. Text yang memiliki jumlah karakter tertentu ditransformasi ke dalam bentuk hex melalui tahapan kompresi dan enkripsi sehingga jumlah karakter dari text dapat menyusut. Selain itu juga, *QR Code Reader* yang diusulkan dapat bekerja dengan baik dalam mengembalikan informasi yang disematkan ke dalam QR Code sehingga user dapat menerima atau membaca pesan yang disampaikan. Untuk melihat efektivitas QR Code Generator dan QR Code Reader, maka *flowchart* yang diusulkan diuji pada 12 dataset yang berbentuk text baik yang Bahasa Indonesia, Bahasa Inggris maupun Indonesia-Inggris. Berikut adalah hasil pengujian *QR Code Generator* dan *QR Code Reader* yang memanfaatkan metode Kompresi Huffman dan Enkripsi RSA dalam mengoptimalkan kapasitas informasi pada QR Code.

Tabel 2 Hasil Pengujian *QR Code Generator* dalam Mengoptimalkan Kapasitas Informasi pada QR Code

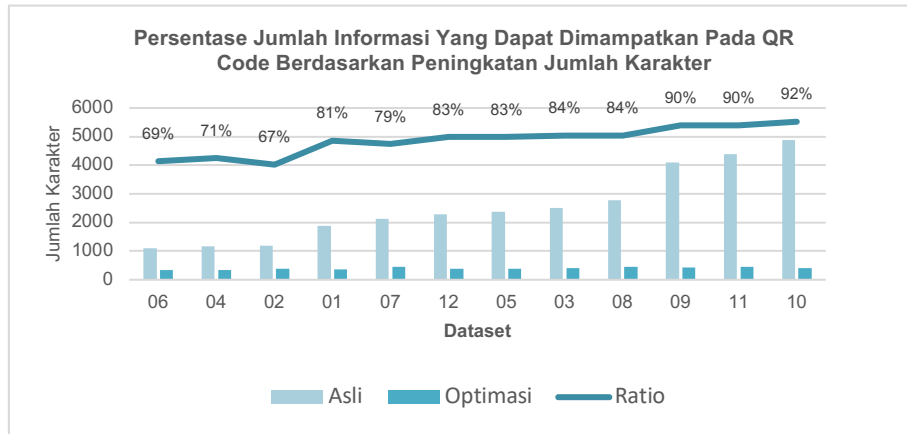
No	Text	Jumlah Karakter		Ratio
		Asli	Optimasi	
1	File01.txt	1880	362	81 %
2	File02.txt	1182	392	67 %
3	File03.txt	2510	398	84 %
4	File04.txt	1176	338	71 %
5	File05.txt	2375	392	83 %
6	File06.txt	1094	338	69 %
7	File07.txt	2119	446	79 %
8	File08.txt	2775	446	84 %
9	File09.txt	4092	419	90 %
10	File10.txt	4886	413	92 %
11	File11.txt	4396	461	90 %
12	File12.txt	2274	383	83 %

Berdasarkan tabel 2 dapat diketahui bahwa QR Code Generator dapat melakukan optimalisasi pada text yang berbahasa Indonesia, Bahasa Inggris, dan Indonesia-Inggris. Pada text yang berbahasa indonesia yaitu file01.txt-file04.txt yang memiliki jumlah rata-rata 1687 karakter dapat dikompresi menjadi 373 karakter. Pada text yang berbahasa inggris yaitu file05.txt-file09.txt yang memiliki jumlah rata-rata 2491 karakter dapat dikompresi menjadi 408 karakter. Sedangkan text yang mengandung bahasa indonesia dan Inggris dengan jumlah rata-rata 3852 karakter dapat dikompresi menjadi 419 karakter.

3.4 Evaluasi *Flowchart QR Code Generator* dan QR Code Reader dalam Optimalisasi Informasi Berupa Text pada QR Code

Flowchart QR Code Generator yang diusulkan dalam penelitian ini terbukti dapat diimplementasikan pada text baik yang berbahasa indonesia, bahasa inggris ataupun indonesia-inggris. Begitupun juga dengan *Flowchart QR Code Reader*, secara teknis dapat mengembalikan

atau menyampaikan informasi yang tersimpan dalam QR Code kepada user. Untuk melihat efektivitas *flowchart QR Code Generator* dalam optimalisasi informasi berupa text pada QR Code maka diperlukan parameter yang dapat menjadi pembanding utama dengan metode atau pendekatan lain. Pada penelitian ini yang menjadi parameter adalah seberapa banyak (%) *flowchart QR Code Generator* dapat mengoptimalkan informasi berupa text pada QR Code.



Gambar 6 Grafik peningkatan jumlah kapasitas informasi pada QR Code menggunakan *flowchart QR Code Generator*

Untuk tujuan evaluasi, maka gambar 6 adalah hasil pengujian *flowchart QR Code Generator* pada 12 dataset dimana datasetnya disusun berdasarkan jumlah karakter dari yang yang terkecil ke yang terbanyak. Hal ini dilakukan dengan tujuan untuk mengetahui apakah ada korelasi yang signifikan antara banyaknya informasi dalam bentuk text yang disematkan dengan besarnya persentase optimalisasi kapasitas informasi pada QR Code.

Berdasarkan gambar 6 maka dapat diketahui bahwa dataset 06 yang memiliki 1094 karakter dapat diminimumkan menjadi 338 karakter. Artinya terjadi optimalisasi pada kapasitas QR Code sebesar 69%. Pada dataset 03 yang memiliki 2510 karakter, dapat diminimumkan menjadi 398 karakter. Artinya terjadi optimalisasi sebesar 84% pada kapasitas QR Code. Sedangkan pada dataset 10 yang memiliki 4886 karakter dapat diminimumkan menjadi 413 karakter. Artinya terjadi optimalisasi sebesar 92% pada kapasitas QR Code. Secara keseluruhan *flowchart QR Code Generator* dapat mengoptimalkan kapasitas informasi pada QR Code sebanyak sebesar 81%. Metode yang diusulkan pada penelitian ini lebih baik dari pada metode yang diusulkan oleh Erna Zuni Astuti pada tahun 2013 dan Hidayat pada tahun 2022 yang tingkat optimalisasinya berturut-turut sebesar 50% dan 56,6% [17][15].

Selain itu juga dapat diketahui bahwa besarnya optimalisasi kapasitas pada QR Code yang menggunakan QR Code Generator berbanding lurus dengan banyaknya karakter yang disematkan pada QR Code. Artinya semakin banyak jumlah informasi dalam bentuk text yang disematkan pada QR Code semakin besar pula tingkat optimalisasi kapasitas pada QR Code. Hal ini dapat terjadi karena *Flowchart QR Code Generator* dibangun dengan memanfaatkan metode kompresi Huffman dan Enkripsi RSA, dimana karakter yang sering muncul akan digantikan dengan code yang lebih kecil dan karakter yang jarang muncul akan digantikan oleh karakter yang lebih panjang.

Dengan terjadinya pemampatan informasi pada QR Code sebesar 81% maka bentuk QR Code yang dihasilkan oleh QR Code generator tidak akan terlalu kompleks. Dengan normalnya bentuk QR Code ini maka akan meminimalisir terjadinya kesalahan saat melakukan pemindaian dengan menggunakan QR Code Reader.

4. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian maka dapat disimpulkan bahwa *flowchart QR Code Generator* yang memanfaatkan metode kompresi Huffman dan Enkripsi RSA dapat mengoptimalkan kapasitas informasi pada QR Code sebesar 81%. Bentuk QR Code yang dihasilkan oleh QR Code Generator tetap normal (tidak kompleks) sehingga memudahkan dalam proses pemindaian ulang menggunakan QR Code Reader. Penelitian ini pada dasarnya menghasilkan rancangan QR Code Generator dan QR Code Reader sehingga memungkinkan untuk diadopsi atau dikembangkan ke dalam bentuk aplikasi yang siap pakai.

DAFTAR PUSTAKA

- [1] K. Rotsios, A. Konstantoglou, D. Folinias, T. Fotiadis, L. Hatzithomas, and C. Boutsouki, "Evaluating the Use of QR Codes on Food Products," *Sustainability*, vol. 14, no. 8, 2022, doi: 10.3390/su14084437.
- [2] L.-Y. Yan, G. W.-H. Tan, X.-M. Loh, J.-J. Hew, and K.-B. Ooi, "QR code and mobile payment: The disruptive forces in retail," *J. Retail. Consum. Serv.*, vol. 58, p. 102300, 2021, doi: <https://doi.org/10.1016/j.jretconser.2020.102300>.
- [3] N. Gangurde, S. Ghosh, A. Giri, and S. Gharat, "Ticketing System Using AES Encryption Based QR Code," in *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2022, pp. 201–206, doi: 10.1109/ICSSIT53264.2022.9716234.
- [4] R. Jadhav, A. Shaikh, M. A. Jawale, A. B. Pawar, and P. William, "System for Identifying Fake Product using Blockchain Technology," in *2022 7th International Conference on Communication and Electronics Systems (ICCES)*, 2022, pp. 851–854, doi: 10.1109/ICCES54183.2022.9835866.
- [5] M. Mulyadi, "Optimization Model of Integrated Logistics Using Continuous Review and Greedy Algorithm," *Indones. J. Inf. Technol.*, vol. 01, no. 02, pp. 20–29, 2017, [Online]. Available: <http://usnsj.com/index.php/InJITech/article/view/388>.
- [6] G. L. Intal, J. D. Payas, L. M. Fernandez, and B. M. Domingo, "Restaurant Information System (RIS) with QR Code to Improve Service Operations of Casual Fine Dining Restaurant," in *2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA)*, 2020, pp. 1054–1059, doi: 10.1109/ICIEA49774.2020.9102036.
- [7] G. Wang, A. Sutikno, F. Ginting, and N. Angelica, "Applying QR Code in Mobile Banking Use," in *2021 International Conference on Information Management and Technology (ICIMTech)*, 2021, vol. 1, pp. 835–839, doi: 10.1109/ICIMTech53080.2021.9534952.
- [8] V. D. Dokania, M. M. Sevak, D. D. Patel, and P. S. Barve, "QR Code based Smart Parking System," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, 2020, pp. 167–170, doi: 10.1109/ICCSP48568.2020.9182180.
- [9] L. Yingli, "Design and Implementation of QR Code Payment in Power Self-service Payment Terminal," in *2021 6th International Symposium on Computer and Information Processing Technology (ISCIPT)*, 2021, pp. 432–435, doi: 10.1109/ISCIPT53667.2021.00093.
- [10] A. M. Ali and A. K. Farhan, "Enhancement of QR Code Capacity by Encrypted Lossless Compression Technology for Verification of Secure E-Document," *IEEE Access*, vol. 8, pp. 27448–27458, 2020, doi: 10.1109/ACCESS.2020.2971779.
- [11] O. Appiah, E. M. Martey, C. B. Ninfaakanga, N. S. Awarayi, and E. Opoku, "Performance Evaluation of Corrupted QR Code Scanners," in *2021 IEEE 8th International Conference on Adaptive Science and Technology (ICAST)*, 2021, pp. 1–6, doi: 10.1109/ICAST52759.2021.9682029.
- [12] C. Bhardwaj, H. Garg, and S. Shekhar, "An Approach for Securing QR code using Cryptography and Visual Cryptography," in *2022 International Conference on*

- Computational Intelligence and Sustainable Engineering Solutions (CISES)*, 2022, pp. 284–288, doi: 10.1109/CISES54857.2022.9844332.
- [13] S. Tiwari, “An Introduction to QR Code Technology,” in *2016 International Conference on Information Technology (ICIT)*, 2016, pp. 39–44, doi: 10.1109/ICIT.2016.021.
- [14] H. E. Gonçalves, L. X. Medeiros, and A. C. Mateus, “Algorithm for Locating the Vertices of a QR Code and Removing Perspective,” *IEEE Lat. Am. Trans.*, vol. 19, no. 11, pp. 1933–1940, 2021, doi: 10.1109/TLA.2021.9475627.
- [15] H. Hidayat and M. Y. Asshobry, “Penerapan Metode Kompresi Huffman untuk Meningkatkan Jumlah Data Informasi pada Kode Quick Response,” *JOINS (Journal Inf. Syst.*, vol. 7, no. 2, pp. 155–165, 2022, doi: 10.33633/joins.v7i2.6789.
- [16] C. Bhardwaj and H. Garg, “An Approach for Enhancing Data Storage Capacity in Quick Response Code using Zip Compression Technique,” in *2023 International Conference on Artificial Intelligence and Smart Communication (AISC)*, 2023, pp. 520–524, doi: 10.1109/AISC56616.2023.10085559.
- [17] E. Zuni Astuti and E. Yudi Hidayat, “Kode Huffman Untuk Kompresi Pesan,” *Techno.Com*, vol. 12, no. 2, pp. 117–126, 2013.
- [18] Y. Anshori, A. Y. Erwin Dodu, and D. M. P. Wedananta, “Implementasi Algoritma Kriptografi Rivest Shamir Adleman (RSA) pada Tanda Tangan Digital,” *Techno.Com*, vol. 18, no. 2, pp. 110–121, 2019, doi: 10.33633/tc.v18i2.2166.