



# **APLIKASI DETEKSI PLAGIARISME JUDUL TUGAS AKHIR BERBASIS WEB DENGAN MENGGUNAKAN ALGORITMA RABIN-KARP ROLLING HASH (STUDI KASUS: AMIK MAHAPUTRA RIAU)**

Herianto<sup>1</sup>, Yulisman<sup>2</sup>, Winda Herrianti Manullang<sup>3</sup>, Yuda Irawan<sup>4</sup>

<sup>1,2,3,4</sup>Sistem Informasi, STMIK Hangtuh Pekanbaru

**Email :**

[herianto.sy@gmail.com](mailto:herianto.sy@gmail.com)<sup>1</sup> [yulisman@htp.ac.id](mailto:yulisman@htp.ac.id)<sup>2</sup> [windaherriantimanullang@gmail.com](mailto:windaherriantimanullang@gmail.com)<sup>3</sup> [yudairawan89@gmail.com](mailto:yudairawan89@gmail.com)<sup>4</sup>

## **Abstract**

*The thesis the web-based final task plagiarism detection application using the rabin-karp algorithm (case study: Amik Mahaputra Riau). Where this system is used in the environment of mahaputra riau. With this system , it can make it easier for administrators, department heads, and students at the mahaputra riau campus to detect plagiarism in the final assignment of students and simplify the work process of the head of department in checking the same final assignments and detect plagiarism. Accessed through each student account. This information system is designed using a model Unified Modeling Language (UML). The method used is the waterfall method. The reason for using this method is because the waterfall method approaches systematically and sequentially in building a system. The process of the waterfall method, namely the spelling of a system is carried out sequentially. The system produced will be of good quality, due to its implementation in stages so that is not focused on certain stages. It is expected that with this information system, it can facilitate administrators, department heads, and student to detect plagiarism in final assignments and other documents and make it easier for students to find out the final percentage plagiarism without having to come to campus.*

**Keywords:** System, Plagiarism, Algoritma Rabin-Karp, UML, Waterfall..

## **Abstrak**

Artikel ini membahas tentang Aplikasi Deteksi Plagiarisme Tugas Akhir Berbasis Web Dengan Menggunakan Algoritma Rabin-Karp Rolling Hash (Studi Kasus: Amik Mahaputra Riau). Dimana Sistem ini digunakan di Amik Mahaputra Riau. Dengan adanya sistem ini dapat mempermudah admin, ketua jurusan dan mahasiswa di kampus Amik Mahaputra Riaudalam pendeteksian plagiarisme tugas akhir mahasiswa dan mempermudah proses kerja ketua jurusan dalam mengecek tugas akhir yang sama serta mempermudah mahasiswa mengupload berkas tugas akhir dan mendeteksi plagiarisme isi tugas akhir yang diakses melalui akun mahasiswa masing-masing. Sistem informasi ini didesain menggunakan model Unified Modeling Language (UML).Metode yang digunakan yaitu metode waterfall. Alasan menggunakan metode ini adalah karena metode waterfall melakukan pendekatan secara sistematis dan berurutan dalam membangun suatu sistem. Proses metode waterfall yaitu pada pengerjaan dari suatu sistem dilakukan secara berurutan. Sistem yang dihasilkan akan berkualitas baik, dikarenakan pelaksanaannya secara bertahap sehingga tidak terfokus pada tahapan tertentu. Diharapkan dengan adanya sistem informasi ini, bisa memudahkan admin, ketua jurusan dan mahasiswa dalam mendeteksi plagiarisme tugas akhir maupun dokumen lain dan mempermudah mahasiswa untuk mengetahui hasil persentase plagiarisme tugas akhirnya tanpa harus datang ke kampus.

**Keywords:** Sistem, Plagiarisme, Tugas Akhir, Algoritma Rabin-Karp, UML, Waterfall..

## PENDAHULUAN

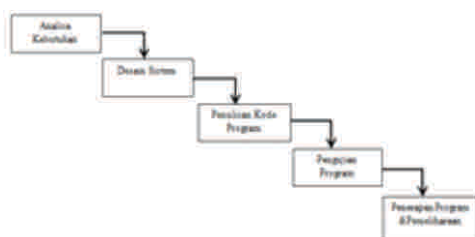
Di dunia pendidikan, pelaku plagiarisme dapat mendapat hukuman berat seperti dikeluarkan dari sekolah atau universitas. Pelaku plagiat disebut sebagai plagiator. Menurut Peraturan Menteri Pendidikan Republik Indonesia Nomor 17 Tahun 2010 dikatakan plagiat adalah perbuatan sengaja atau tidak sengaja dalam memperoleh atau mencoba memperoleh kredit atau nilai untuk suatu karya ilmiah, dengan mengutip sebagian atau seluruh karya dan atau karya pihak lain yang diakui sebagai karya ilmiahnya, tanpa menyatakan sumber secara tepat dan memadai. Untuk melakukan deteksi plagiarisme dokumen teks pada intinya adalah dengan melakukan pencocokan string. Salah satu yang dapat digunakan dalam pencocokan string adalah Algoritma Rabin-Karp RollingHash. Algoritma Rabin-Karp adalah algoritma pencarian kata yang mencari sebuah pola berupa substring dalam sebuah teks menggunakan fungsi hashing. (Noprisson, dkk. 2013).

AMIK Mahaputra Riau adalah salah satu perguruan tinggi swasta yang ada di Pekanbaru yang memiliki satu program studi manajemen informatika dengan pendidikan diploma tiga. Mahasiswa AMIK Mahaputra Riau untuk menyelesaikan pendidikannya harus melakukan karya ilmiah (Tugas Akhir).

Proses tugas akhir di AMIK Mahaputra Riau dimulai dari pengajuan judul sampai ke ujian akhir (ujian komprehensif). Namun, selama proses penyusunan dan sampai ujian akhir tugas akhir baik mahasiswa, dosen, maupun pihak akademik tidak pernah melakukan pengecekan plagiarisme terhadap tugas akhir mahasiswa. Selama ini proses yang terjadi hanya pemeriksaan terhadap seleksi judul dan pemeriksaan kelengkapan isi tugas akhir sebelum ujian akhir (komprehensif) secara manual yaitu dengan mengecek satu persatu dan mengetik hasil pengecekan tersebut di Microsoft Excel dan Microsoft Word. Sehingga jika kemiripan tugas akhir baik dari segi judul maupun isi tidak terdeteksi adanya plagiarisme oleh mahasiswa dan itu terjadi setiap tahunnya.

## METODE PENELITIAN

Metode analisis data dalam perancangan perangkat lunak Aplikasi Deteksi Plagiarisme Tugas Akhir Berbasis Web Dengan Menggunakan Algoritma Rabin-Karp Rolling Hash (Studi Kasus: Amik Mahaputra Riau) menggunakan paradigma perangkat lunak model waterfall.



Gambar. 1 Metode Waterfall Menurut Kadir 2014

Tahapan Metode Waterfall (Kadir, 2014):

### 1. Analisa Kebutuhan

Pada tahap ini dilakukan analisa kebutuhan sistem. Analisa disini mengumpulkan data yang diperlukan penulis dalam membangun aplikasi, dimulai dari penelitian yang dilakukan ke Amik Mahaputra Riau, wawancara serta studi literature sehingga nantinya akan didapatkan aplikasi yang dapat digunakan admin, ketua jurusan, mahasiswa dan dosen dalam pendeteksian plagiarism tugas akhir dengan menggunakan algoritma rabin-karp rolling hash. Disini dianalisa bagaimana nantinya proses pengembangan aplikasi dan pemanfaatan dari segi hardware maupun software serta perangkat-perangkat pendukung lainnya.

### 2. Desain Sistem

Pada tahap ini mulailah dirancang bagaimana nantinya membangun suatu aplikasi yang akan membantu admin, ketua jurusan, mahasiswa dan dosen dalam mendeteksi plagiarism isi tugas akhir maupun kemiripan judul dengan menggunakan algoritma rabin-karp. Adapun tahapan desain disini meliputi perancangan aplikasi dengan membuat skema seperti Use Case Diagram, Activity Diagram, dan Class Diagram. Berikut adalah gambaran model perancangan sistem:

#### a. Perancangan data

Pada tahap analisa data terdapat dua jenis data yang dilakukan proses nantinya oleh sistem yakni data masukan (input data) dan data keluaran (output data).

#### b. Perancangan pengguna

Pengguna adalah orang yang mengoperasikan sistem, pengguna akan memanfaatkan dan berinteraksi dengan sistem. Pengguna yang akan menggunakan Aplikasi Deteksi Plagiarisme Tugas Akhir adalah sebagai berikut:

#### 1. Administrator

Administrator dapat mengelola, merespon, menambah, mengubah, dan menghapus data dan lain-lain. Administrator pada sistem ini adalah Admin yang ditunjuk ketua jurusan AMIK Mahaputra Riau yang bertanggung jawab dengan aplikasi deteksi plagiarism.

#### 2. Dosen

Dosen dalam hal ini menjadi pengguna aktif yang melihat data tugas akhir mahasiswa yang terbukti tidak plagiarism dan cek data mahasiswa yang akan dibimbing dalam penyelesaian tugas akhir.

#### 3. Mahasiswa

Mahasiswa dalam aplikasi deteksi plagiarisme ini adalah pengguna tetap yang hanya bisa mengupload tugas akhir, menunggu proses dan melihat hasil persentase plagiarism kemudian mencetak hasil plagiarism kepada ketua jurusan dan melakukan bimbingan tugas akhir.

#### 4. Ketua Jurusan

Ketua Jurusan pada sistem ini adalah Pengguna Aktif yang ditunjuk untuk bertanggung jawab dengan aplikasi deteksi

plagiarisme.

c. Perancangan Model Sistem

Unified modeling language (UML) muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. (Rosa & Shalahuddin, 2013: 137). Berikut perancangan model sistem informasi:

1) Use Case Diagram

Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan informasi yang akan dibuat. (Rossa & Shalahuddin, 2013).

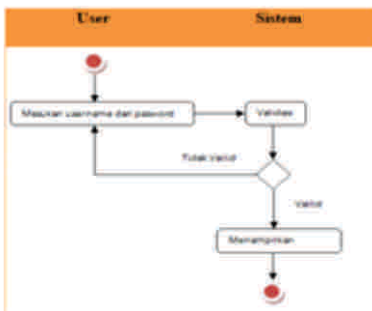


Gambar. 2 Diagram Use Case

2) Activity Diagram

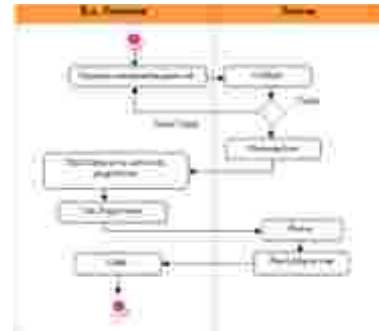
Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rossa & Shalahuddin, 2013). Berikut beberapa perancangan diagram aktivitas user pada sistem:

a. Diagram Aktivitas (Activity Diagram) Login Admin



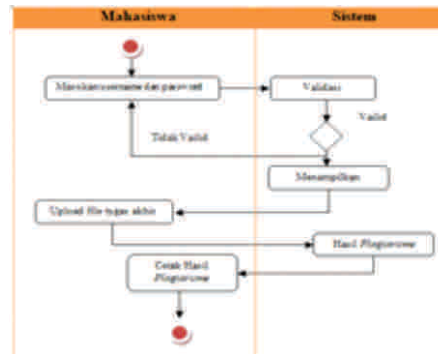
Gambar. 3 Diagram Activity Login Admin

b. Diagram Aktivitas (Activity Diagram) Menu Input Ka. Jurusan



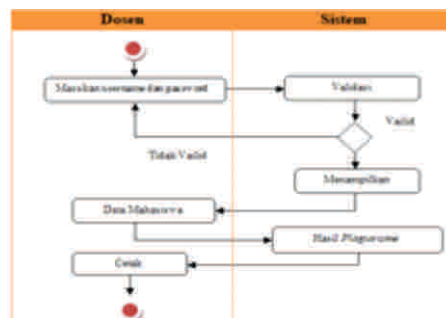
Gambar. 4 Diagram Activity Ka. Jurusan

c. Diagram Aktivitas (Activity Diagram) Menu Input Mahasiswa



Gambar 5 Diagram Activity Input Mahasiswa

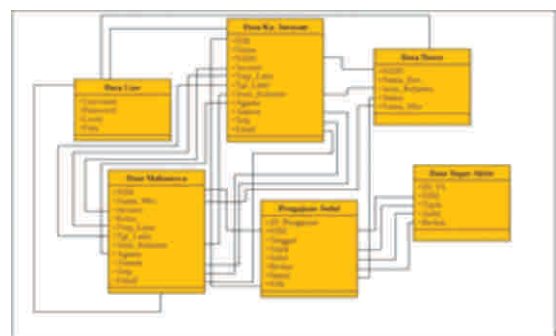
d. Diagram Aktivitas (Activity Diagram) Menu Input Dosen



Gambar. 6 Diagram Activity Dosen

3) Diagram Kelas (Class Diagram)

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa dan Shalahuddin, 2013).



Gambar. 7 Diagram Kelas (Class Diagram)

#### 4)Penulisan Kode Program

Penulisan kode program atau coding merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan menterjemahkan transaksi yang diminta oleh user. Tahapan ini merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing sistem yang telah dibuat tadi. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

#### 5)Pengujian Program

Tahapan akhir dimana sistem yang baru diuji kemampuan dan keefektifannya sehingga didapatkan kekurangan dan kelemahan sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi menjadi lebih baik dan sempurna.

#### 6)Penerapan Program dan Pemeliharaan

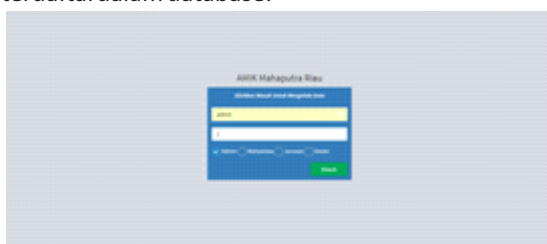
Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.

## HASIL & PEMBAHASAN

Hasil dan pembahasan merupakan tampilan dari sistem yang telah dibangun:

#### 1.Tampilan Menu Login

Menu Login terdiri dari empat, pertama login pada bagian admin, kedua login pada bagian mahasiswa, ketiga login pada bagian ketua jurusan, dan keempat login pada bagian dosen. Untuk masuk ke dalam sistem, admin harus menginputkan Username dan Password yang telah terdaftar dalam database.



**Gambar.8 Tampilan Halaman Login**

#### 2.Tampilan Tampilan Halaman Awal Pengguna.

Pada tampilan halaman awal pengguna ini akan ditampilkan Home, Info Data Mahasiswa, Data Dosen, Ketua Jurusan, Pengajuan Judul, Tugas Akhir, Laporan Pengajuan, K-Gram dan Pengguna Sistem.



**Gambar.9 Tampilan Awal Pengguna**

#### 3.Tampilan Input Mahasiswa

Berikut adalah tampilan input data mahasiswa yang dilakukan oleh admin agar mahasiswa dapat mengakses akun dengan masing-masing email dan nim dan masuk sebagai mahasiswa.



**Gambar.10 Tampilan Input Mahasiswa**

#### 4.Tampilan Daftar Data Mahasiswa

Berikut adalah tampilan daftar data mahasiswa yang dilakukan oleh admin untuk mengetahui tugas akhir yang akan diakses mahasiswa.



**Gambar.11 Tampilan Daftar Data Mahasiswa**

#### 5.Tampilan Menu Input Data Dosen

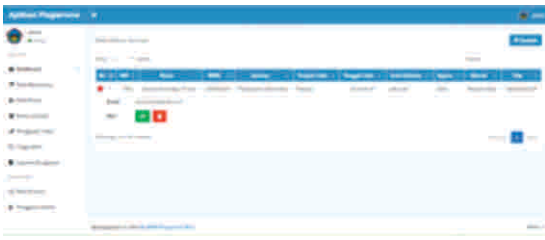
Pada tampilan menu input data dosen terdapat daftar data mahasiswa yang telah dilakukan oleh admin agar mahasiswa dapat melihat nama, jurusan, tempat lahir, tanggal lahir, jenis kelamin, agama, email dan foto.



**Gambar.12 Tampilan Menu Input Data Dosen**

#### 6.Tampilan Menu Input Ketua Jurusan

Berikut ini adalah tampilan input data ketua jurusan yang dilakukan oleh admin.



**Gambar. 13 Tampilan Menu Input Ketua Jurusan**

#### 7. Tampilan Data Pengajuan Tugas Akhir

Berikut ini adalah tampilan daftar data pengajuan judul tugas akhir yang dilakukan oleh admin.



**Gambar. 14 Tampilan Data Pengajuan Tugas Akhir**

#### 8. Tampilan Laporan Cetak Plagiarisme

Tampilan laporan cetak plagiarisme digunakan untuk mengetahui tingkat persentase plagiarismenya dan sebagai bukti mahasiswa yang akan diserahkan kepada ketua jurusan.



**Gambar. 15 Tampilan Laporan Cetak Plagiarisme**

#### 9. Tampilan Daftar Data Tugas Akhir

Tampilan daftar data tugas akhir digunakan agar mahasiswa dan ketua jurusan dapat melihat tugas akhir yang akan diupload dan ketua jurusan serta admin mudah untuk memproses data tugas akhir dari mahasiswa.



**Gambar. 16 Tampilan Daftar Data Tugas Akhir**

#### 10. Tampilan Menu Cek Plagiarisme Tugas Akhir

Pada tampilan menu ini mahasiswa dapat melakukan upload berkas tugas akhir dan mendeteksi plagiarisme tugas akhir dengan

menggunakan hak akses masing-masing dan dapat mengetahui hasil dari kemiripan judul dan berkas tugas akhir dengan perbandingan tugas akhir mahasiswa lain.



**Gambar 17 Tampilan Menu Cek Plagiarisme Tugas Akhir**

#### 11. Tampilan Bukti Terdeteksi Plagiarisme

Berikut ini adalah tampilan bukti terdeteksi plagiarisme yang dilakukan admin dalam pengecekan tugas akhir mahasiswa untuk mengetahui nilai persentase tingkat plagiarismenya dan tidak dapat dicetak. Karena persentase tingkat plagiarisme tertinggi adalah tidak lebih dari 50%, sehingga untuk tampilan cetak otomatis tidak tampak.



**Gambar. 18 Tampilan Bukti Terdeteksi Plagiarisme**

#### 12. Tampilan Menu Laporan Pembayaran

Berikut ini adalah tampilan dari bukti lulus plagiarisme yang dilakukan oleh admin untuk mengetahui nilai persentase terbukti tidak melakukan plagiat, dan secara otomatis tampilan cetak laporannya akan muncul dan dapat dicetak bukti plagiarismenya.



**Gambar. 19 Tampilan Bukti Lulus Plagiarisme**

## KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan pembahasan, yang didukung dengan penerapan metode penelitian maka diperoleh kesimpulan bahwa:

- Bahwa sistem dalam membandingkan file memberikan hasil berupa presentase plagiarisme.
- Terdapat beberapa faktor yang mempengaruhi performa algoritma rabin karp, diantaranya banyaknya konten sebuah file akan memperpanjang waktu prosesnya (running time), (stemming), dan (preprocessing) membuat waktu proses cenderung lebih lama tetapi tingkat akurasi similaritynya lebih

tinggi, nilai modulo berpengaruh pada waktu proses, dan semakin kecil k-grams akan menghasilkan akurasi nilai plagiarisme yang lebih baik.

Berdasarkan kesimpulan diatas, ada beberapa saran dalam implementasi sistem aplikasi deteksi plagiarisme yang dapat penulis kemukakan yaitu sebagai berikut :

- a. Sistem dapat membandingkan file uji dengan semua file sumber pada database sistem.
- b. Jika masih menggunakan algoritma rabin karp, lebih dibenahi pada tahap stemming agar hasil pencocokannya lebih akurat.
- c. Diharapkan sistem dapat mendeteksi sinonim (persamaan) dan berbagai bentuk plagiat, serta dapat mengkonversi dokumen lain dengan ekstensi lain seperti .doc dan .pdf untuk kemudahan pengguna.

### DAFTAR PUSTAKA

- Aan, 2015. Definisi Web Berbasis Linux Operasi Sistem.
- Deslianti, Mutaqqin. 2016. Perangkat Lunak Aplikasi Pengolah Kata.
- Hermawan, dkk. 2011. Sistem Informasi Penjadwalan Kegiatan Belajar Mengajar Berbasis Web. IJSE – Indonesian Journal on Software Engineering Volume 2 No 1 – 2016 Program Studi Sistem Informasi STMIK ProVisi Semarang ISSN : 2461-0690
- Kadir. 2014. Metode Pengembangan Sistem dan Tahapan Metode Waterfall.
- Menteri Pendidikan Republik Indonesia. 2010. Peraturan Undang-undang tentang Plagiarisme.
- ROFIQOH, Ummi; IRAWAN, Yuda; MELYANTI, Rika. Aplikasi Pendaftaran Dan Penerimaan Data Persyaratan Calon Anggota Legislatif Sementara Secara Online Pada Partai Keadilan Sejahtera Di Kabupaten Bengkalis. Riau Journal Of Computer Science, 2020, 6.2: 145-153.
- Nazaruddin. 2012. Integrated Development Environment (IDE) untuk pengembangan aplikasi Android.
- LENARDO, Gilang Citra, et al. Pemanfaatan Bot Telegram sebagai Media Informasi Akademik di STMIK Hang Tuah Pekanbaru. JTIM: Jurnal Teknologi Informasi dan Multimedia, 2020, 1.4: 351-357.
- Noprison, dkk. 2013 Implementasi Algoritma Rabin-Karp Untuk Menentukan Keterkaitan Antar Publikasi Penelitian Dosen Pada tahun 2013. ISSN : 1414-9999
- Irawan, Y., Wahyuni, R., & Herianto, H. (2019). Media Pembelajaran Bahasa Inggris Dasar Menggunakan Macromedia Flash 8 di Tk Kartika 1.50 Kecamatan Sail Kota Pekanbaru. Informatika, 11(2), 1-7.
- Ruslan. 2016. Perangkat Lunak Aplikasi spreadsheet Electronic dan Program Word Processing
- Santoso, Zefriyenni. 2015. Sistem Informasi Penjualan dan Pengendalian Persediaan Barang Menggunakan Metode Economic Order Quantity (EOQ) Menggunakan Bahasa Pemrograman Java dan Database MYSQL Pada Toko Kansa Elpiji. ISSN : 2356 – 0010
- Shalahuddin dan Rosa, 2009 : 4. Kualitas Perangkat Lunak Dalam Sebuah Sistem dari Segi Kepuasan Perangkat Lunak
- Sastroasmoro. 2014. Implementasi Plagiarisme dalam Pembuatan Karya Ilmiah.
- Soelistiyo, 2011. Definisi Plagiarisme di Kalangan Sekolah dan Universitas.
- IRAWAN, Yuda, et al. Sistem Pendukung Keputusan untuk Menentukan Kegiatan Ekstrakurikuler Berdasarkan Bakat dan Minat Menggunakan Metode SAW (Simple Additive Weighting). JTIM: Jurnal Teknologi Informasi dan Multimedia, 2019, 1.3: 198-205.
- Syahputra, 2015. Implementasi Algoritma Innowing Untuk Deteksi Kemiripan Text. ISSN : 2301-9425
- MUHAIMIN, Abdi Muhaimin Abdi, et al. Sistem Informasi Tracer Study Berbasis Web Pada Program Pascasarjana Fisip Universitas Riau. Jurnal Ilmu Komputer, 2020, 9.2: 71-77.
- Rahmalisa, U., & Linarta, A. (2020). Rancang Bangun Aplikasi Absensi dan Penggajian pada Kantor KPU Menggunakan Framework Laravel (Studi Kasus: KPU Kabupaten Bengkalis). Journal of Technopreneurship and Information System (JTIS), 3(3), 86-91.
- Rahmalisa, U., & Muhandi, M. (2019). Penerapan Metode TOPSIS untuk Seleksi Penerima Beasiswa (Studi Kasus: SMAN 2 Tebing Tinggi Timur). Jurnal Teknologi Sistem Informasi dan Aplikasi, 2(1), 31-35.
- Sutabri, 2012 : 138. Pendekatan Perancangan Model Konseptual Menggunakan Model Data Relasional.
- Irawan, Y., Wahyuni, R., & Herianto, H. (2021). Morse Code Receiver On Invisible Light Using Background Subtraction Method. Journal of Robotics and Control (JRC), 2(4), 283-286.