APLIKASI PENDETEKSI SITUS *PHISING* BERBASIS *WEBSITE* MENGGUNAKAN METODE *NAÏVE BAYES*

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan Studi Pada

Program Studi Teknik Informatika

Oleh:

Mohammad Zaidan Zufar

19090027

PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK HARAPAN BERSAMA

TEGAL

2023

APLIKASI PENDETEKSI SITUS *PHISING* BERBASIS *WEBSITE*

MENGGUNAKAN METODE *NAÏVE BAYES*

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan Studi Pada

Program Studi Teknik Informatika

Oleh:

Mohammad Zaidan Zufar

19090027

PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK HARAPAN BERSAMA

TEGAL

2023

HALAMAN PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama : Mohammad Zaidan Zufar

NIM : 19090027

Adalah mahasiswa Program Studi Sarjana Terapan Teknik Informatika Politeknik Harapan Bersama. Dengan ini saya menyatakan bahwa laporan Skripsi yang berjudul:

"APLIKASI PENDETEKSI SITUS *PHISING* BERBASIS *WEBSITE* MENGGUNAKAN METODE *NAÏVE BAYES*"

Merupakan hasil pemikiran sendiri secara orisinil yang saya susun secara mandiri dengan tidak melanggar kode etik hak karya cipta. Apabila dikemudian hari Laporan Skripsi ini terbukti melanggar kode etik karya cipta, maka saya bersedia untuk melakukan penelitian baru dan menyusun sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Tegal, Agustus 2023

Yang membuat pernyataan,

Mohammad Zaidan Zufar
NIM 19090027

HALAMAN REKOMENDASI

Pembimbing Skripsi memberikan rekomendasi kepada :

Nama : Mohammad Zaidan Zufar

NIM : 19090027

Program Studi : Sarjana Terapan Teknik Informatika

Judul Skripsi : Aplikasi Pendeteksi Situs *Phising* Berbasis *Website* Menggunakan Metode *Naïve Bayes*

Untuk mengikuti Ujian Skripsi karena telah memenuhi persyaratan yang telah ditentukan.

Tegal, Juli 2023

Pembimbing I,

Dega Surono Wibowo, S.T., M.Kom.
NIPY. 06.014.183

Pembimbing II,

M. Nishom, M.Kom.
NIPY. 09.017.337

HALAMAN PENGESAHAN

Nama : Mohammad Zaidan Zufar

NIM : 19090027

Program Studi : Sarjana Terapan Teknik Informatika

Judul Skripsi : Aplikasi Pendeteksi Situs *Phising* Berbasis *Website* Menggunakan Metode *Naïve Bayes*

Dinyatakan Lulus Ujian Skripsi pada program studi Sarjana Terapan Teknik Informatika Politeknik Harapan Bersama.

Tegal, Agustus 2023

Dewan Penguji

Nama		Tanda Tangan
1. Ketua	: Muhammad Fikri Hidayattullah, S.T., M.Kom.	1.
2. Anggota I	: Taufiq Abidin, S.Pd., M.Kom.	2.
3. Anggota II	: M. Nishom, M.Kom.	3.

Mengetahui,
Ketua Program Studi Sarjana Terapan Teknik Informatika

Slamet Wiyono, S.Pd., M.Eng.
NIPY. 08.015.222

ABSTRAK

Phising merupakan salah satu bentuk tindakan *cyber crime* yang semakin meningkat dan menjadi ancaman bagi individu maupun bisnis. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan aplikasi deteksi *phising* berbasis web yang efisien dan akurat dengan menggunakan metode *naive bayes*. Penulis mengumpulkan kumpulan data besar yang mencakup contoh situs web yang *good* dan *bad*, dan melakukan langkah *preprocessing* data untuk mempersiapkan dataset untuk klasifikasi *Naive Bayes*. Pada fase *training*, penulis melatih model klasifikasi *naive bayes* menggunakan kumpulan data yang telah diproses sebelumnya dan mengevaluasi performa menggunakan berbagai metrik dengan skor *precision 97%*, *accuration 96%*, *recall 97%*, dan *fi-score 96%*. Hasil evaluasi menunjukkan bahwa program deteksi *phising* berbasis web dengan metode *Naive Bayes* dapat memberikan hasil yang memuaskan dalam mengidentifikasi situs *phising* yaitu dengan hasil *training* sebesar 98%, *testing* 96% dan evaluasi menggunakan data aktual 72%. Dalam penelitian ini, penulis berhasil mengembangkan aplikasi deteksi *phising* berbasis web dengan menggunakan metode *naive bayes* yang mampu memberikan akurasi yang tinggi dalam mengidentifikasi situs *phising*. Selain itu, penulis juga memberikan beberapa rekomendasi untuk pengembangan dan perbaikan lebih lanjut pada penelitian berikutnya, seperti memperluas dataset dengan mengumpulkan lebih banyak data, meningkatkan performa model klasifikasi, dan mengurangi risiko *overfitting*. Diharapkan hasil penelitian ini dapat membantu pengguna internet untuk lebih waspada dan melindungi diri dari tindakan *phising*. Aplikasi deteksi *phising* berbasis web yang efisien dan akurat dapat membantu pengguna untuk mengidentifikasi situs *phising* dan meningkatkan kesadaran akan bahaya penipuan *online*. Selain itu, hasil penelitian ini juga dapat menjadi acuan bagi penelitian selanjutnya dalam mengembangkan aplikasi deteksi *phising* yang lebih baik dan efektif.

Kata Kunci : *Phising, Website, Deteksi, Naïve Bayes*

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala karunia dan rahmat-Nya yang telah melimpah dalam perjalanan penyusunan laporan Skripsi ini. Laporan ini merupakan hasil dari penelitian penulis yang bertujuan untuk menjelajahi dan menganalisis isu-isu terkait *Cyber Crime* atau Kejahatan Dunia Maya yang kini semakin relevan dan penting dalam dunia teknologi dan informasi.

Skripsi merupakan suatu kewajiban yang harus dilaksanakan untuk memenuhi salah satu syarat kelulusan dalam mencapai Sarjana Terapan Teknik Informatika Politeknik Harapan Bersama. Selama melaksanakan penelitian dan kemudian tersusun dalam laporan Skripsi, banyak pihak yang telah memberikan bantuan, dukungan, serta bimbingan.

Pada kesempatan ini tak lupa penulis mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Agung Hendarto, S.E., M.A., selaku Direktur Politeknik Harapan Bersama Tegal,
2. Slamet Wiyono, S.Pd., M.Eng., selaku Ketua Program Studi,
3. Dega Surono Wibowo, S.T., M.Kom., selaku Dosen Pembimbing I,
4. M. Nishom, M.Kom., selaku Dosen Pembimbing II,
5. Kedua orang tua penulis, selaku wali mahasiswa terutama untuk almarhum ayah penulis yang sudah berjuang membiayai anaknya untuk terus menyelesaikan pendidikan setinggi-tingginya,
6. Sahabat dan teman yang tidak mungkin penulis sebutkan satu-satu,

7. Semua pihak yang telah mendukung, membantu serta mendoakan penyelesaian laporan Skripsi ini.

Laporan Skripsi ini penulis susun dengan sungguh-sungguh dan semaksimal mungkin untuk menyajikan hasil penelitian yang akurat, relevan dan bermanfaat bagi pengembangan ilmu pengetahuan di bidang Teknologi Informasi. Namun, penulis sadar bahwa laporan ini masih memiliki keterbatasan dan kekurangan. Oleh karena itu, kritik, saran, dan masukan konstruktif dari pembaca akan sangat penulis hargai guna perbaikan di masa depan.

Akhir kata, semoga laporan Skripsi ini dapat memberikan manfaat dan sumbangsih dalam pengembangan teknologi keamanan informasi, khususnya dalam deteksi situs *phising*. Penulis berharap laporan ini dapat menginspirasi dan mendorong para pembaca, khususnya rekan mahasiswa, untuk terus berkontribusi dalam perkembangan ilmu pengetahuan.

Tegal, Agustus 2023

Mohammad Zaidan Zufar

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

BAB I

PENDAHULUAN

1.1. Latar Belakang

Saat ini internet sudah menjadi bagian penting dalam kehidupan masyarakat terutama pada aktivitas sosial dan finansial. Sebagai contohnya media sosial yang digunakan sebagai sarana berkomunikasi, mencari teman dan juga bisnis *online* yang digunakan beberapa pihak terutama perusahaan untuk menawarkan perdagangan *online* melalui *e-mail* dan memberitahu kepada calon pelanggan tentang *website* mereka, namun saat ini ada pihak yang tidak bertanggungjawab melakukan tindakan yang merugikan banyak orang yang salah satunya adalah tindakan *phising*[1].

Istilah *phishing* dalam bahasa Inggris datang dari kata memancing (*'fishing'*), dalam hal ini artinya memancing informasi keuangan dan kata sandi pengguna. Dengan banyaknya kasus penipuan yang dilaporkan, metode atau perlindungan tambahan sangat mendesak diperlukan. Upaya tersebut meliputi pembuatan undang-undang, pengguna pelatihan, dan tindakan teknis. *Phising* biasanya sulit dideteksi, apalagi bagi orang awam yang tidak bergerak di bidang teknis. Masalah ini diperparah dengan penggunaan yang semakin meningkat ponsel cerdas yang biasanya tidak menampilkan URL situs web secara keseluruhan[2].

Phising adalah aktivitas *cyber crime* yang menggunakan rekayasa sosial dan tipuan teknis untuk mencuri data identitas dan kredensial akun keuangan. Skema rekayasa sosial dilakukan dengan menggunakan *email*

palsu yang mengaku berasal dari institusi bisnis yang sah dan dirancang untuk mengarahkan korban ke situs web palsu yang mengelabui, sehingga korban membocorkan data keuangan seperti nama dan kata sandi. Skema subterfomen teknis menanam *crimeware* ke PC untuk mencuri kerahasiaan secara langsung, sering menggunakan sistem untuk mengelabui nama pengguna dan kata sandi akun *online* dan merusak infrastruktur navigasi lokal untuk menyesatkan konsumen ke situs palsu (atau situs asli melalui *proxy* yang dikendalikan *phisher* yang digunakan untuk memantau dan *intercept* pada konsumen)[3].

Phising merupakan suatu bentuk kegiatan yang bersifat mengancam atau menjebak seseorang dengan konsep memancing orang tersebut. Yaitu dengan menipu seseorang sehingga orang tersebut secara tidak langsung memberikan semua informasi yang dibutuhkan oleh sang penjebak. *Phising* termasuk dalam kejahatan *cyber crime*, dimana sekarang ini marak terjadi tindak kriminal melalui jaringan komputer. Seiring perkembangan zaman, tindak kriminal juga semakin merebak di seluruh dunia. Sehingga ancaman yang banyak terjadi saat ini juga melalui komputer. Bagi *hacker* cara ini merupakan cari paling mudah untuk dijadikan serangan. Meskipun dianggap mudah dan sepele tapi tetap saja ada pengguna yang masuk ke perangkap sang *hacker*[4].

Phising yaitu aktivitas seseorang untuk mendapatkan informasi rahasia pengguna dengan cara menggunakan email dan situs web palsu yang tampilannya menyerupai tampilan asli atau resmi web sebenarnya.

Informasi yang didapat atau dicari oleh *phisher* adalah berupa *password* akun atau nomor kartu kredit korban. Penjebak (*phisher*) menggunakan *email*, *banner* atau *pop-up window* untuk menjebak *user* agar mengarahkan ke situs web palsu (*fake webpage*), di mana pengguna diminta untuk memberikan informasi pribadinya. Di sinilah *phisher* memanfaatkan kecerobohan dan ketidak telitian pengguna dalam web palsu tersebut untuk mendapatkan informasi[5].

Indonesia merupakan negara hukum hal ini tercantum dalam pasal 1 ayat 3 UUD 1945. Tidak lepas dari Indonesia yang merupakan negara hukum penegak hukum yang dalam menangkap dan menegakkan hukum harus mempunyai dasar hukum yang kuat, hal ini didasari dengan Indonesia yang menganut asas legalitas yang dituangkan dalam pasal 1 ayat 1 KUHP, yaitu "suatu perbuatan tidak dapat dipidanakan kecuali berdasarkan kekuatan ketentuan perundang-undangan pidana yang telah ada". Peraturan mengenai pencurian yang dilakukan melalui elektron telah diatur dalam undang-undang nomor 11 tahun 2008 tentang Informasi dan Transaksi Elektronik yang mengatur juga tentang *cybercrime*[6].

Aksi *phising* ini semakin marak terjadi. Tercatat secara global, jumlah penipuan bermodus *phising* 42% dari modus selain *phising* yang dinyatakan dalam website *Anti-Phishing Working Group* (APWG) dalam laporan bulannya, mencatat ada 12.845 *e-mail* baru dan unik serta 2.560 situs palsu yang digunakan sebagai sarana *phising*[7]. Maka dari itu sebuah sistem pendeteksian sebuah url atau *website phising* perlu adanya, sehingga

pencurian informasi data dapat diminimalisir dengan sebuah aplikasi pendeteksi url atau *website phising*.

Dalam sebuah sistem deteksi diperlukan sebuah model untuk menjalankan sebuah sistem dimana model tersebut dibuat dari sebuah metode, dalam kasus kali ini penulis memilih *Naive Bayes*, *Naive Bayes Classifier (NBC)* merupakan salah satu metoda pembelajaran mesin yang memanfaatkan perhitungan probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya. Teori *Naive Bayes Classifier* bekerja sangat baik dibanding dengan model *classifier* lainnya.

Hal ini dibuktikan dalam jurnal "*Naïve Bayes vs Decision Trees vs Neural Networks in the Classification of Training Web Pages*" mengatakan bahwa *Naïve Bayes Classifier* memiliki tingkat akurasi yang lebih baik dibanding model classifier lainnya[8]. Metode *naïve bayes classifier* akan mencari beberapa kemungkinan aman atau tidaknya situs web yang kita akses tersebut. Dengan begitu kita bisa menjaga keamanan data kita dengan tidak mengakses web yang sudah terdeteksi sebagai web *phising*. Metode *naïve bayes* ini dipilih karena dirasa paling tepat untuk menyelesaikan masalah deteksi *website phising*. Metode ini mampu menyeleksi data dengan cara mengklasifikasikan sekumpulan data dengan memanfaatkan probabilitas dan statistik. Dimana probabilitas yang digunakan yaitu dengan menggunakan prediksi probabilitas masa depan dengan dasar-dasar masa

sebelumnya. Klasifikasi *Naive Bayes* diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri dari kelas lainnya.

Pada penelitian [9] bertujuan untuk melakukan prediksi klasifikasi penerima bantuan sembako dengan menggunakan algoritma *Naïve Bayes*. Algoritma *Naïve Bayes* memiliki fungsi untuk menemukan pengetahuan atau pola-pola kesamaan karakteristik dalam suatu kelompok atau kelas tertentu. Prediksi penerima bantuan sembako yang digunakan terdapat dua kelas, yaitu layak dan tidak layak. Data yang digunakan untuk prediksi yaitu data sampel dari desa XYZ. Pada penelitian ini algoritma *Naïve Bayes* diimplementasikan dan dianalisa menggunakan aplikasi yang dikembangkan berbasis web menggunakan bahasa pemrograman PHP.

Dengan adanya implementasi pada penelitian sebelumnya penulis bermaksud untuk melakukan penelitian yang melibatkan metode *naïve bayes* dalam melakukan sebuah deteksi pada *Uniform Resource Locator* (URL) yang terindikasi *phising* dengan memanfaatkan fungsi dari metode *naïve bayes* yang dapat menemukan pola atau karakteristik dalam suatu kelompok hanya saja dalam penelitian ini pengimplementasian model ini menggunakan *framework flask*, dengan itu maka akan terjadinya suatu relevansi jika metode ini dijadikan sebuah alat untuk mendeteksi.

1.2. Tujuan dan Manfaat

1.2.1. Tujuan Penelitian

Tujuan aplikasi ini adalah untuk memberikan informasi kepada *user* mengenai url yang mencurigakan yang akan dideteksi menggunakan *website* pendeteksi dengan metode *naïve bayes* dan *User* akan mendapat informasi yang jelas mengenai url yang mencurigakan apakah url tersebut terindikasi *phising* atau tidak itu akan diproses melalui sistem *machine learning*.

1.2.2. Manfaat Penelitian

Berdasarkan tujuan dari penelitian ini maka manfaat yang akan didapat adalah sebagai berikut :

1. Pendeteksi sebuah url yang dicurigai sebagai *website phising* melalui *platform website*.
2. Bagi masyarakat penelitian ini dapat digunakan untuk mendeteksi sebuah url yang dicurigai sebagai *website phising* atau *non phising*.
3. Bagi kominfo penelitian ini diharapkan mampu mencegah terjadinya *cyber crime.*

1.3. Tinjauan Pustaka

Penerapan Algoritma *Naïve Bayes Classifier* Untuk Meningkatkan Keamanan Data Dari *Website Phising*, penelitian tersebut berisi tentang penerapan Algoritma *Naïve Bayes* dalam mengamankan data dari bahaya Situs *Phising*, hasil dari penelitian ini adalah Algoritma *Naïve Bayes*

mengeluarkan rata-rata akurasi sebesar 92.98% , Dengan demikian hasil penerapan Algoritma *Naïve Bayes* tersebut untuk melindungi data dari *website phising* dikatakan sangat baik, dan penggunaan algoritma tersebut sudah tepat jika digunakan untuk pencegahan pencurian data dari sebuah ancaman *phising* Landasan Teori[8].

Penelitian lain yang meneliti tentang Deteksi Komentar *Cyberbullying* Pada Media Sosial Berbahasa Inggris Menggunakan Metode *Naïve Bayes* yang timbul dari masalah banyaknya pengguna Media Sosial di Indonesia mencapai 150 juta jiwa atau 56% dari total populasi penduduk Indonesia dan jumlah tersebut naik sebesar 20% dari survey sebelumnya menjadikan Sosial mempunyai sisi positif dan negatif dalam sisi negatifnya adanya *Cyberbullying* yang membuat masalah ini akan terus membesar seiring bertambahnya pengguna media sosial di Indonesia dengan hasil penelitian pengujian menggunakan metode *Naïve Bayes Classification* didapatkan nilai akurasi sebesar 80% dengan nilai rata-rata menghasilkan *precision* 81%, *recall* 80% dan *f1-score* 80%[10].

Pada penelitian ini melakukan prediksi klasifikasi penentuan penerima bantuan sembako menggunakan algoritma *Naïve Bayes*. Algoritma *Naïve Bayes* merupakan salah satu metode yang dapat digunakan untuk mengklasifikasikan data. *Bayesian classification* merupakan pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu *class*. Algoritma *Naïve Bayes* memiliki fungsi untuk menemukan pengetahuan atau pola-pola kesamaan

karakteristik dalam suatu kelompok atau kelas tertentu. Prediksi tingkat penerimaan bantuan sembako yang digunakan terdapat dua kelas, yaitu layak dan tidak layak. Data yang digunakan untuk prediksi yaitu data yang diambil dari sampel data warga di desa XYZ. Dari hasil evaluasi menggunakan *confusion matrix* didapatkan akurasi yang dihasilkan untuk 135 data *training* dengan 40 data *testing* dan tujuh atribut yang digunakan menghasilkan akurasi sebesar 86%, *recall* 85%, dan *precision* 88%. Akurasi dapat dipengaruhi oleh beberapa faktor, diantaranya: jumlah data *training*, data *testing* dan atribut yang digunakan. Untuk penelitian selanjutnya dapat menggunakan variasi data *training*, data *testing* dan atribut sehingga didapatkan model dengan akurasi yang terbaik[9].

Pada penelitian ini melakukan analisis sentimen masyarakat terhadap hasil *quick count* pemilihan presiden indonesia 2019 pada media sosial *twitter* menggunakan metode *NBC* dengan menghasilkan tingkat akurasi sebesar 82,29% dengan nilai $\alpha = 0{,}05$. Dengan klasifikasi yang diperoleh masing-masing sebesar 34,5 (471) *tweet* positif dan 65,5% (895) *tweet* negatif terhadap hasil *quick count* yang menandakan sentimen masyarakat condong pada *tweet* negatif terhadap *quick count* pemilihan presiden 2019[11].

Pada penelitian ini [12] didapatkan kesimpulan sistem deteksi ujaran kebencian dalam Bahasa Indonesia pada *tweet* dan *mention* di *Twitter* dengan menggunakan metode *Naïve Bayes* berbasis *website* berhasil mengklasifikasikan *tweet* dan *mention* di *twitter* berupa kalimat ujaran

ancaman dan bukan kalimat ujaran ancaman dengan menggunakan 90% data latih sedangkan 10 % data uji dengan akurasi final 66% dan nilai parameter *precision, recall*, dan *f-1 score* sebesar 63% dan akurasi sebesar 66 %.

Pada penelitian Implementasi Metode *Klasifikasi Naïve Bayes* Dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga Metode *Naive Bayes* memanfaatkan data training untuk menghasilkan probabilitas setiap kriteria untuk *class* yang berbeda, sehingga nilai-nilai probabilitas dari kriteria tersebut dapat dioptimalkan untuk memprediksi penggunaan listrik berdasarkan proses klasifikasi yang dilakukan oleh metode *Naive Bayes* itu sendiri. Berdasarkan data rumah tangga yang dijadikan data training, metode *Naive Bayes* berhasil mengklasifikasikan 47 data dari 60 data yang diuji. Sehingga metode *Naive Bayes* berhasil memprediksi besarnya penggunaan listrik rumah tangga dengan persentase keakuratan sebesar 78,3333%[13].

Pada penelitian Klasifikasi Berita Hoax Dengan Menggunakan Metode *Naive Bayes*, metode *naive bayes* dapat digunakan pada sistem klasifikasi berita dengan masukan berupa teks dengan diawali tahap *preprocessing* yang berupa *parsing*, *tokenization*, *stopword*, dan pembobotan kata (*term weighting*). Kemudian dilakukan klasifikasi dengan metode *naive bayes*. Tahap terakhir yaitu dilakukan pengukuran dengan menggunakan pengujian *10-fold cross validation*. Dari hasil penelitian diketahui nilai *fold 6* memberikan nilai akurasi dengan hasil terbaik dengan

hasil dengan nilai keakuratan sebesar 85.28 % yang mana terklasifikasi dokumen yang relevan sebanyak 307 dan yang tidak relevan sebanyak 53 atau error rate sebesar 14.72%. Sedangkan nilai rata-rata berdasarkan berita hoax dan berita benar nilai *precision* 0,896 dan *recall* 0.853[14].

Pada penelitian [15] *Naïve Bayes* metode yang lebih unggul dari *K-Nearest Neighbor* dalam pengklasifikasian artikel dalam bahasa indonesia dengan hasil Naïve Bayes dari 40 jurnal yang diujikan metode ini dapat mengklasifikasikan artikel jurnal berbahasa indonesia sebanyak 28 dokumen dengan *Accuracy* 70%, *Recall* 70%, *Precision* 70.9% dan *Error* 30%. Sedangkan *K-Nearest Neighbor* hanya dapat mengklasifikasi dokumen sebanyak 16 saja dan mendapatkan *Accuracy* 40%, *Recall* 40%, *Precision* 64.1% dan *Error* 60%.

Penelitian terdahulu membuktikan bahwa *Naïve Bayes* merupakan salah satu metode yang dapat digunakan dalam proses mendeteksi sebuah teks yang memiliki karakteristik tertentu dan dalam beberapa penelitian sebelumnya metode ini mempunyai hasil akurasi yang baik pula bahkan sampai 96% saat *testing* oleh penulis sehingga memutuskan menggunakan metode ini sebagai pendeteksi url yang mencurigakan dan yang nantinya akan diimplementasikan ke dalam *website* menggunakan *framework flask* sehingga *user* dapat menggunakan *tools* pendeteksi *website phising* ini dengan mudah karena kemudahan *user interface* nantinya.

1.4. Data Penelitian

Bahan yang digunakan pada penelitian ini adalah dataset yang didapatkan pada *website kaggle.com, research* pada beberapa media sosial dan data dari Kominfo Kota Tegal. Penelitian ini menggunakan data sebanyak 551.585 dengan label *website* yang berlabel *good* berjumlah 393.712 dan *bad* berjumlah 157.873 untuk mendapatkan hasil akurasi yang tinggi.

1.5. Alat Penelitian

Peralatan yang digunakan dalam penelitian ini dikelompokan menjadi dua, Perangkat Keras (*Hardware*) dan Perangkat Lunak (*Software*), yaitu sebagai berikut :

1. Perangkat keras atau *Hardware* yang terdiri dari :
   a. Laptop Lenovo L-340 Intel Core i7-9750H RAM 8 GB DDR4 SSD 512 GB Graphic Card Nvidia GeForce GTX 1650 4 GB
   b. Mouse Wireless
2. Perangkat lunak atau Software yang terdiri dari :

Tabel 1. 1 Perangkat Lunak

No	Nama Perangkat Lunak	Fungsi
1.	Windows 11	Sistem Operasi
2.	Visual Studio Code	Digunakan untuk menulis code dalam pembuatan website
3.	Kaggle Notebook dan Kaggle.com	Notebook pembuatan model dan Website penyedia dataset

4.	Chrome	Browser untuk research dan membuka kaggle notebook
5.	Microsoft Excel atau Spreadsheet	Untuk melihat dataset dan mengelolanya
6.	Figma	Perancangan UI Design dan Flow website
7.	Microsoft Word	Menulis laporan sebagai dokumentasi penelitian
8.	MySQL	Database untuk menyimpan hasil deteksi situs phising
9.	Xampp	Webserver lokal untuk menjalankan MySQL
10.	Python	Bahasa Pemrograman untuk pembuatan model
11.	sklearn.metrics	Modul untuk mengevaluasi kinerja model pada penelitian ini yang digunakan adalah accuracy_score
12.	nltk.stem.snowball	Modul untuk menghilangkan kata imbuhan pada suatu kata atau lebih dikenal dengan istilah stemming
13.	nltk.tokenize	Modul yang digunakan untuk

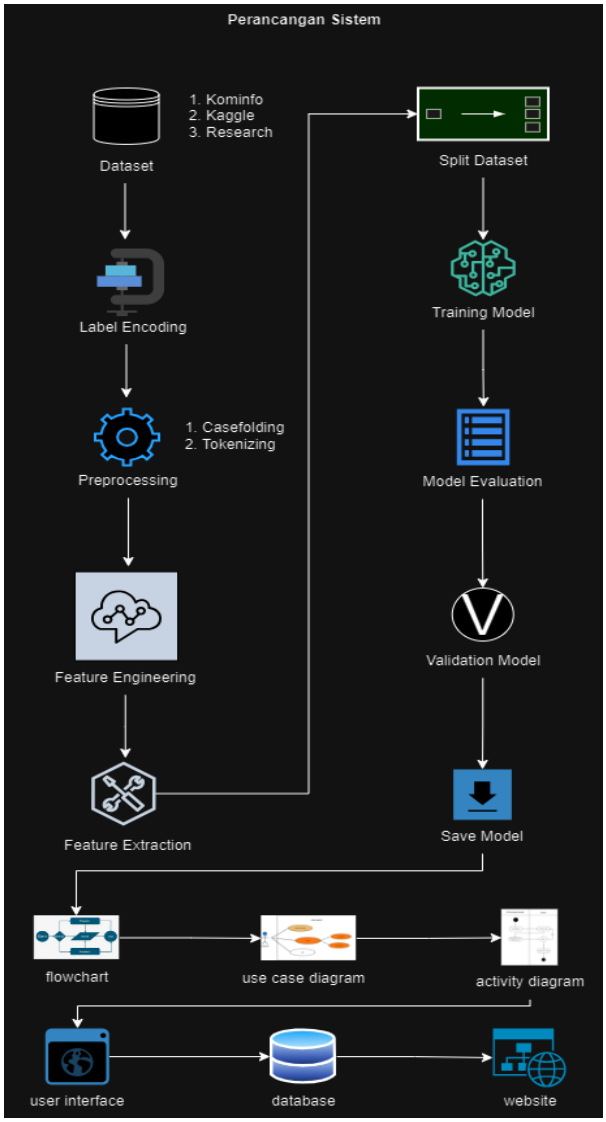
		melakukan tokenisasi teks agar suatu teks diubah menjadi unit yang lebih kecil yang disebut token
14.	sklearn.preprocessing	Modul yang digunakan untuk melakukan preprocessing data sebelum data dimasukan ke dalam model
15.	sklearn.feature_extraction	Modul yang digunakan untuk melakukan ekstraksi fitur atau representasi numerik dalam penelitian ini yang digunakan adalah TfidfVectorizer
16.	sklearn.model_selection	Modul yang digunakan untuk melakukan pemisahan data uji dan data latih
17.	sklearn.naive_bayes	Modul yang digunakan untuk memanggil model

BAB II

PRODUK

2.1. Perancangan

Pada tahapan perancangan dibutuhkan sebuah arsitektur komputer seperti pada gambar 2.1 yang bertujuan untuk mengetahui bagaimana sistem ini dibuat yang menjelaskan mengenai mekanisme dan komponen-komponen yang ada dan dibutuhkan selama pembuatan aplikasi.



Gambar 2. 1. Arsitektur Komputer

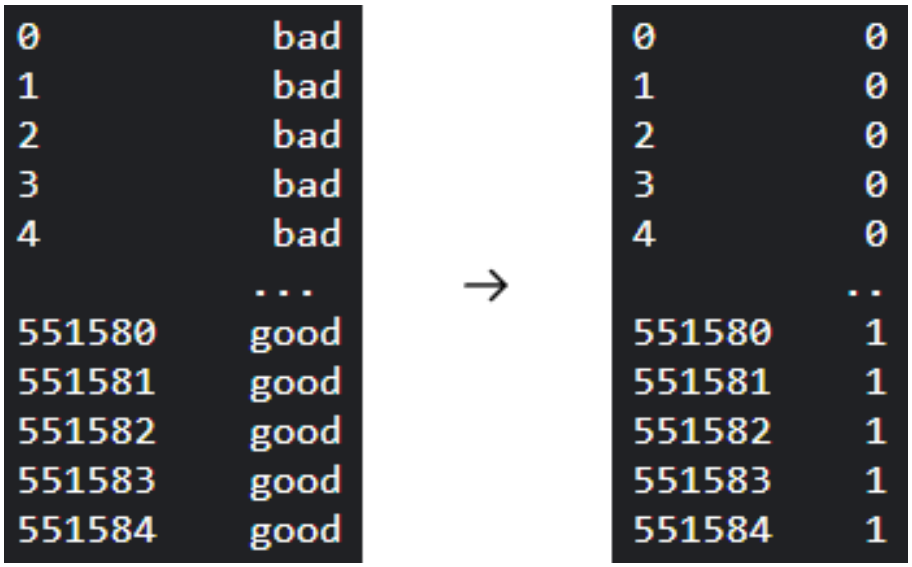
Berikut adalah penjelasan dari komponen dan langkah-langkah yang ada pada gambar arsitektur sistem diantaranya adalah sebagai berikut :

2.1.1. *Dataset*

Dataset didapatkan dari berbagai sumber beberapa diantaranya adalah dari kaggle.com yang merupakan sebuah *website* penyedia *dataset* lalu dari badan Kementerian Komunikasi dan Informasi atau yang biasa dikenal Kominfo dalam hal ini Kominfo Kota Tegal yang selanjutnya adalah hasil dari *research* dari berbagai sosial media yang total dari *dataset* tersebut adalah 551.585.

2.1.2. *Label Encoding*

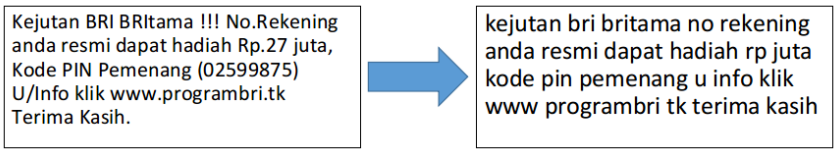
Merupakan suatu tahapan untuk mengubah label *dataset* dari yang berbentuk kata menjadi numerik label *good* akan diganti menjadi 1 dan *bad* diganti menjadi 0 contohnya seperti pada gambar 2.2.



Gambar 2. 2. Label Encoder

2.1.3. *Preprocessing*

Tahapan ini memuat langkah-langkah sebelum pembuatan model, beberapa langkah tersebut seperti *casefolding* adalah sebuah metode merubah huruf kapital ke dalam huruf kecil dengan tujuan menyamakan semua huruf dalam suatu dokumen contohnya seperti pada gambar 2.3.



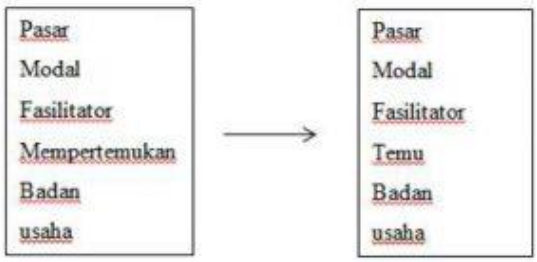
Gambar 2. 3. Casefolding

Selanjutnya ada metode *tokenizer* adalah sebuah metode yang bertujuan untuk berfungsi untuk memotong teks menjadi satu *set token* (kata) dan/atau kalimat. Proses pemotongan kata tersebut kemudian disebut *tokenization*. Istilah *tokenizer* digunakan setelah *update* status diperbaiki dengan *word normalizer*[16] contoh penggunaan metode *tokenizer* adalah seperti pada gambar 2.4.

	URL	Label	URL_TOKEN
0	nobell.it/70ffb52d079109dca5664cce6f317373782/...	0	[nobell, it, ffb, d, dca, cce, f, login, skype...
1	www.dghjdgf.com/paypal.co.uk/cycgi-bin/webscrc...	0	[www, dghjdgf, com, paypal, co, uk, cycgi, bin...
2	serviciosbys.com/paypal.cgi.bin.get-into.herf....	0	[serviciosbys, com, paypal, cgi, bin, get, int...
3	mail.printakid.com/www.online.americanexpress....	0	[mail, printakid, com, www, online, americanex...
4	thewhiskeydregs.com/wp-content/themes/widescre...	0	[thewhiskeydregs, com, wp, content, themes, wi...
...	...	...	...
551580	https://www.tiket.com/	1	[https, www, tiket, com]
551581	https://www.paypal.com/id/webapps/mpp/home	1	[https, www, paypal, com, id, webapps, mpp, home]
551582	https://www.paypal.com/id/webapps/mpp/send-pay...	1	[https, www, paypal, com, id, webapps, mpp, se...
551583	https://www.paypal.com/id/webapps/mpp/fraud-pr...	1	[https, www, paypal, com, id, webapps, mpp, fr...
551584	https://www.paypal.com/id/webapps/mpp/partner-...	1	[https, www, paypal, com, id, webapps, mpp, pa...

Gambar 2. 4. Tokenizer

Selanjutnya adalah proses *stemming* adalah pemrosesan pengembalian suatu kata menjadi kata aslinya atau menghilangkan kata tambahan dari struktur kata yang lengkap contohnya seperti pada gambar 2.5.



Gambar 2. 5. Stemming

Pada penelitian ini dilakukan uji coba menggunakan *stemming* yang merupakan metode yang biasa digunakan pada pendekatan analisis sentimen dan didapatkan bahwa hasil model yang dipakaikan *stemming* dan tidak, tidak memiliki perbedaan signifikan percobaan validasi dari 50 url yang menjadi bahan uji coba baik model yang menggunakan *stemming* ataupun tidak sama-sama hanya menghasilkan 36 data yang benar jadi diputuskan pada penelitian kali ini untuk tidak menggunakan metode *stemming* pada tahapan *preprocessing*.

2.1.4. *Feature Engineering*

Adalah proses dalam analisis data dan pembelajaran mesin di mana membuat atau mengubah fitur-fitur dari data mentah menjadi representasi yang lebih bermakna dan informatif untuk meningkatkan kinerja model pembelajaran mesin. Tujuan dari

feature engineering adalah untuk mengidentifikasi, menggabungkan, dan mengubah fitur-fitur yang relevan agar model dapat memahami pola-pola yang ada dalam data dengan lebih baik.

2.1.5. *Feature Extraction*

Pada tahapan ini menggunakan *Tf-Idf* (Term Frequency-Inverse Document Frequency) dimana hal ini merupakan teknik yang umum digunakan dalam pemrosesan bahasa alami (NLP) untuk mengubah teks menjadi representasi numerik yang dapat digunakan oleh model pembelajaran mesin. Tujuan utama dari teknik ini adalah untuk mengukur seberapa penting suatu kata dalam suatu dokumen atau korpus dengan memperhitungkan seberapa sering kata tersebut muncul dalam dokumen dan seberapa jarang muncul dalam seluruh korpus. Berikut adalah langkah-langkah utama dalam penerapan *Tf-Idf* sebagai teknik *feature extraction* :

1. *Term Frequency* (TF): Ini mengukur seberapa sering kata tertentu muncul dalam dokumen tertentu. Formula yang umum digunakan adalah :

   $$\text{TF}(\text{term},\text{document}) = \frac{\text{total jumlah term dalam document}}{\text{jumlah kemunculan term dalam document}}$$

2. *Inverse Document Frequency* (IDF): Ini mengukur seberapa jarang kata tertentu muncul dalam seluruh korpus. Formula yang umum digunakan adalah :

   $$\text{IDF}(\text{term}) = \log\left(\frac{\text{total jumlah dokumen}}{\text{jumlah dokumen yang mengundang term}}\right)$$

3. *TF-IDF Score*: Menggabungkan informasi dari TF dan IDF untuk menghasilkan skor akhir untuk suatu kata dalam suatu dokumen. Formula yang umum digunakan adalah :

$$\text{TF IDF}(\text{term},\text{document}) = \text{TF}(\text{term},\text{document}) \times \text{IDF}(\text{term})$$

2.1.6. *Split Dataset*

Membagi dataset ke dalam dua subset *training set* dan subset *test set*, serta menetapkan *random state* agar dilakukan pengacakan data sehingga hasil didapatkan dengan konsisten seperti pada gambar 2.6.

```
X_train, X_test, y_train, y_test = train_test_split(X_tf_idf, y,
                                                    test_size=0.2, random_state=42)
```

Gambar 2. 6. Split Dataset

2.1.7. *Training Model*

Seperti pada gambar 2.7 pelatihan dataset pada model *naïve bayes* dengan memanggil model tersebut dan latih dengan dataset *training subset* dan *subset test set.*

```
algorithm = MultinomialNB()
model = algorithm.fit(X_train, y_train)
```

Gambar 2. 7. Training Model

Dengan terbuatnya model maka pengimplementasian ke dalam *website* dapat dilakukan tetapi sebelum dilakukan implementasi dilakukan testing model terlebih dahulu dilakukan

testing model dengan *compare* dengan model lain hal ini dilakukan untuk melihat apakah *naïve bayes* benar-benar terbukti bagus dalam pengklasifikasian url *phising.*

2.1.8. *Model Evaluation*

Evaluasi model dengan memprediksi dengan menggunakan model yang sudah dibuat dengan *subset y_test* dan dihasilkan akurasi seperti pada gambar 2.8.

```
Jumlah prediksi benar   : 106500
Jumlah prediksi salah   : 3817
Akurasi pengujian       : 96.53997117398045 %
```

Gambar 2. 8. Model Evaluation

2.1.9. *Model Validation*

Tahapan ini penulis mencoba melakukan uji validasi model dan komparasi model berikut beberapa model yang penulis coba komparasi *Naïve Bayes, Logistic Regression, Random Forest* dan *Support Vector Machine* (SVM), *Naïve Bayes* merupakan salah satu model yang cocok pada penelitian kali ini karena cara kerja dari *Naïve Bayes* adalah pengklasifikasian berdasarkan teks oleh karena itu penulis memutuskan memakai metode tersebut. Namun pada penelitian ini penulis juga mencoba metode lain contohnya adalah *Random Forest Classifier* namun metode ini mempunyai kelemahan dengan kapasitas data yang dapat dijadikan dataset itu terbatas dari total 551.585 data *Random Forest* hanya dapat memuat 11.736 data saja dengan akurasi seperti pada gambar 2.9.

```
Confusion matrix RandomForestClassifier :
 [[ 650  192]
 [  80 1426]]

.:. RandomForestClassifier Accuracy on Training: 100.00%  .:.
.:. RandomForestClassifier Accuracy on Testing: 88.42%  .:.
```

Gambar 2. 9. Akurasi Random Forest

Pada test data aktual dari 50 data *Random Forest* mampu menghasilkan 40 data benar dengan persentase 80%, Sedangkan pada kasus yang sama metode SVM juga tidak mampu menggunakan keseluruhan dataset yang ada dan hanya menggunakan data sejumlah 11.736 dan mendapatkan data benar 36 dengan persentase 72% seperti pada gambar 2.10.

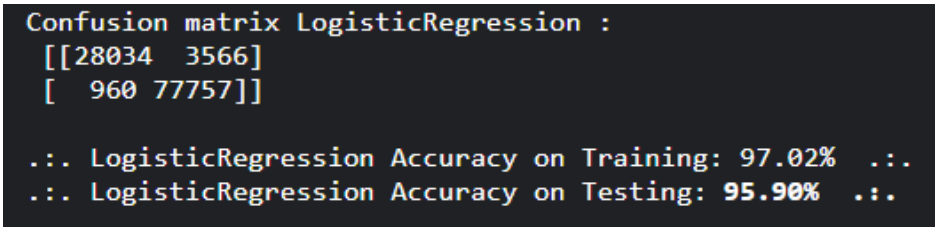
```
Confusion matrix SVM :
 [[ 575  229]
 [  45 1499]]

.:. SVM Accuracy on Training: 99.67%  .:.
.:. SVM Accuracy on Testing: 88.33%  .:.
```

Gambar 2. 10. Akurasi SVM

Pada kasus lain *Logistik Regression* dan *Naïve Bayes* mampu menggunakan seluruh data yang ada sejumlah 551.585 dengan akurasi seperti pada gambar 2.11 dan 2.12.

```
Confusion matrix MultinomialNB :
 [[28174  3426]
 [  594 78123]]

.:. MultinomialNB Accuracy on Training: 97.80%  .:.
.:. MultinomialNB Accuracy on Testing: 96.36%  .:.
```

Gambar 2. 11. Akurasi Naïve Bayes



Gambar 2. 12. Akurasi Logistik Regression

Pada *test* data aktual kedua metode tadi menghasilkan akurasi sebagai *berikut* dengan *Logistik Regression* sebesar 32 data dengan persentase 64% dan *Naïve Bayes* sebesar 36 data dengan persentase 72%. Dapat dilihat pada tabel 2.1 dibawah ini.

Tabel 2. 1. Perbandingan Antar Metode

Kriteria	Naïve Bayes	Logistic Regression	Random Forest	SVM
Data Aktual Benar	36	32	40	36
Presentase	72%	64%	80%	72%
Dataset	551.585	551.585	11.736	11.736

Dapat disimpulkan metode yang terbaik adalah *Naïve Bayes* karena pada penelitian ini metode *Naïve Bayes* menjadi metode yang dapat menggunakan seluruh dataset yang ada dengan akurasi *training* 97%, *testing* 95% serta *test* data aktual *sebesar* 72% dan menjadi metode yang memiliki presentase data *test* tertinggi dibanding *Logistic Regression.*

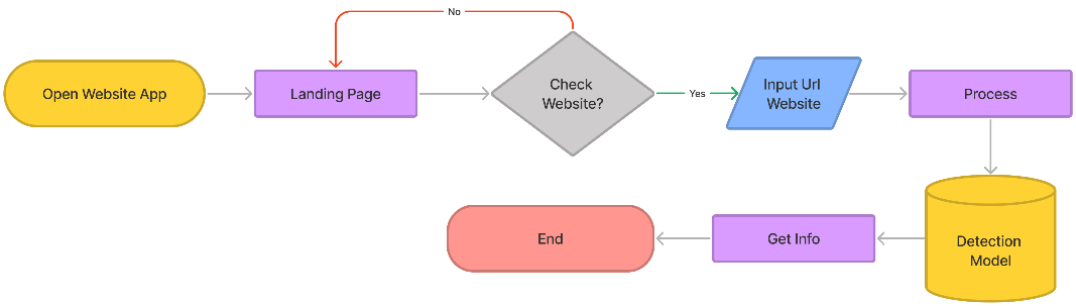
2.1.10. *Save Model*

Menyimpan model adalah suatu langkah akhir dari proses awal pembuatan model dari tahap pengumpulan dataset sampai model validasi, pada tahapan ini dibutuhkan library joblib untuk menyimpan model dan model akan tersimpan dengan ekstensi modelNaiveBayes.model. Pada gambar 2.13. berikut adalah cara untuk menyimpan model.

```
import joblib
joblib.dump((model), "modelNaiveBayes.model")
```

Gambar 2. 13. Save Model

2.1.11. *Flowchart*



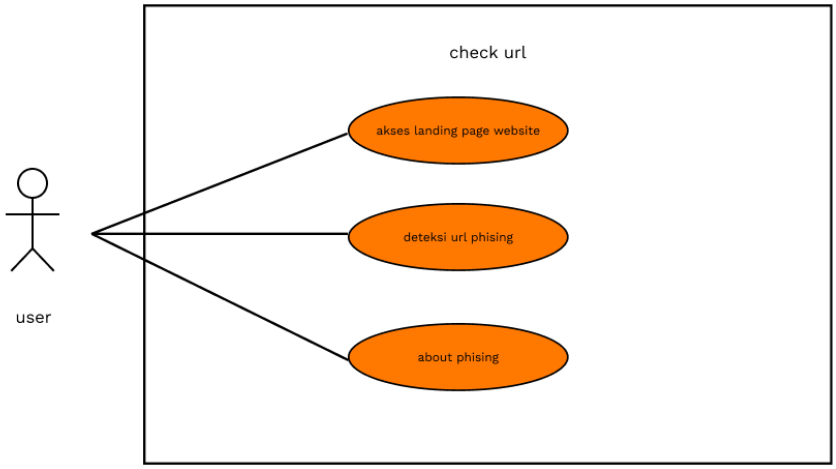
Gambar 2. 14. Flowchart

Pada gambar 2.14 merupakan *flowchart* dari langkah *user* dalam mengambil keputusan saat menjalankan aplikasi ini dengan langkah pertama yaitu membuka website itu sendiri pada browser lalu user akan mengakses halaman *landing page* dimana pada halaman tersebut *user* jika akan mengecek url harus memasukan url

pada form yang tersedia jika tidak maka *user* hanya akan berhenti di halaman *landing page* tetapi jika *user* melakukan pendeteksian maka langkah selanjutnya adalah url tersebut akan dikirim ke bagian *backend* dan selanjutnya akan dideteksi oleh model dan kemudian *user* akan mendapatkan informasi berupa *output* pada *website* dan sistem berakhir.

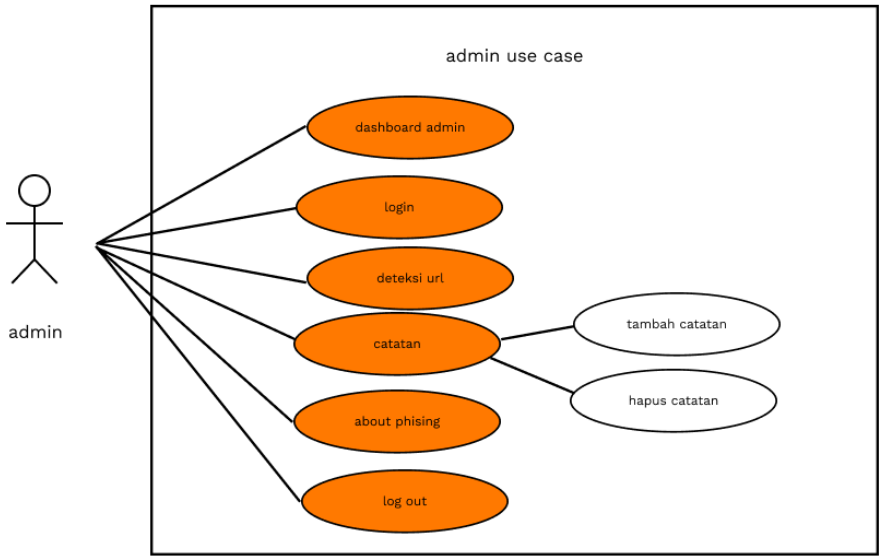
2.1.12. *Use Case Diagram*

Aplikasi ini mempunyai dua *use case* diagram diantaranya yaitu untuk *user* dan admin. Untuk *user* hanya perlu mengakses menu *landing page* dan langsung dapat menggunakan fitur utama yaitu deteksi url yang akan melalui pada gambar 2.15.



Gambar 2. 15. User Use Case Diagram

Untuk admin perlu *login* untuk mengakses *dashboard* admin dan pada halaman admin, admin dapat menambah dan menghapus catatan seperti pada gambar 2.16.



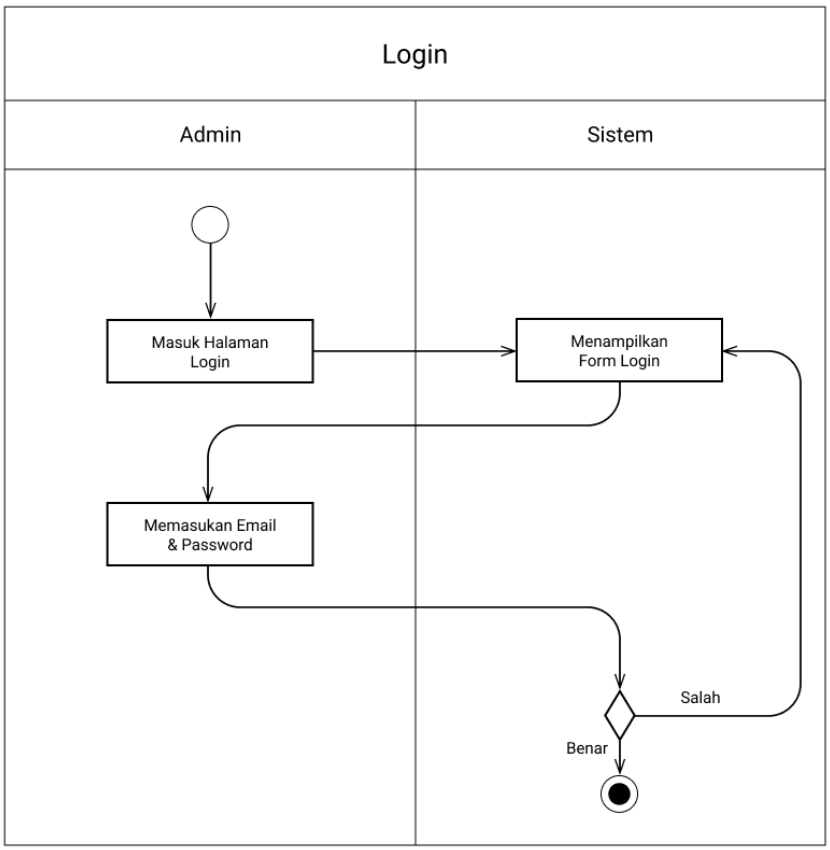
Gambar 2. 16. Admin Use Case

2.1.13. *Diagram Activity*

Pada aplikasi **"Phisers"** ini terdapat beberapa *diagram activity* diantaranya adalah *Activity Login Admin, Activity Landing Page, Activity Admin Dashboard, Diagram Activity Admin Logout* dan *Diagram Activity* Tambah Catatan. Penjelasannya adalah seperti pada berikut ini :

a. *Diagram Activity Login Admin*

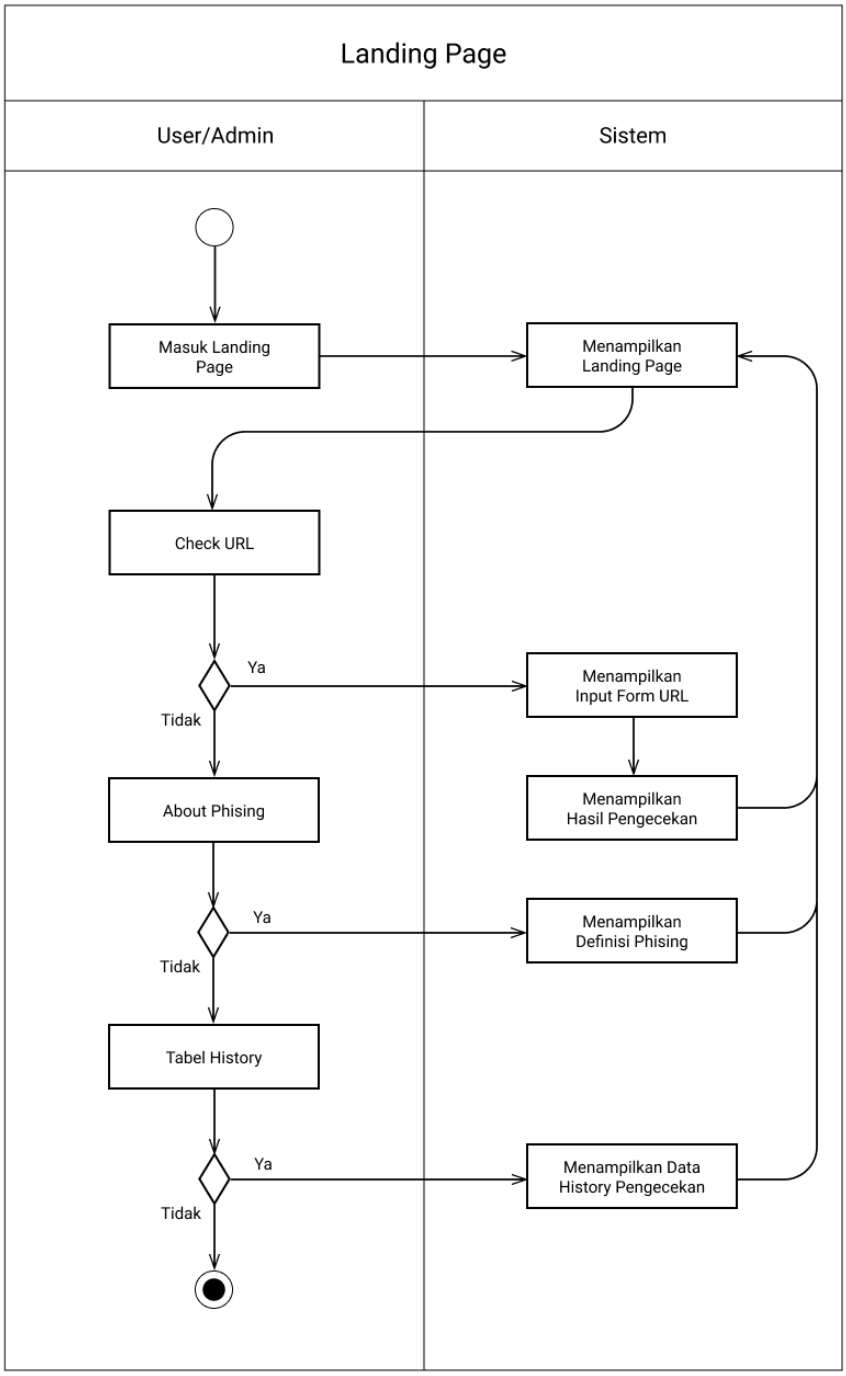
Pada gambar 2.17 dijelaskan secara visual tahapan bagaimana admin dan sistem melakukan aktifitas secara bersamaan saat melakukan *login*.



Gambar 2. 17. Diagram Activity Login

b. *Diagram Activity Landing Page*

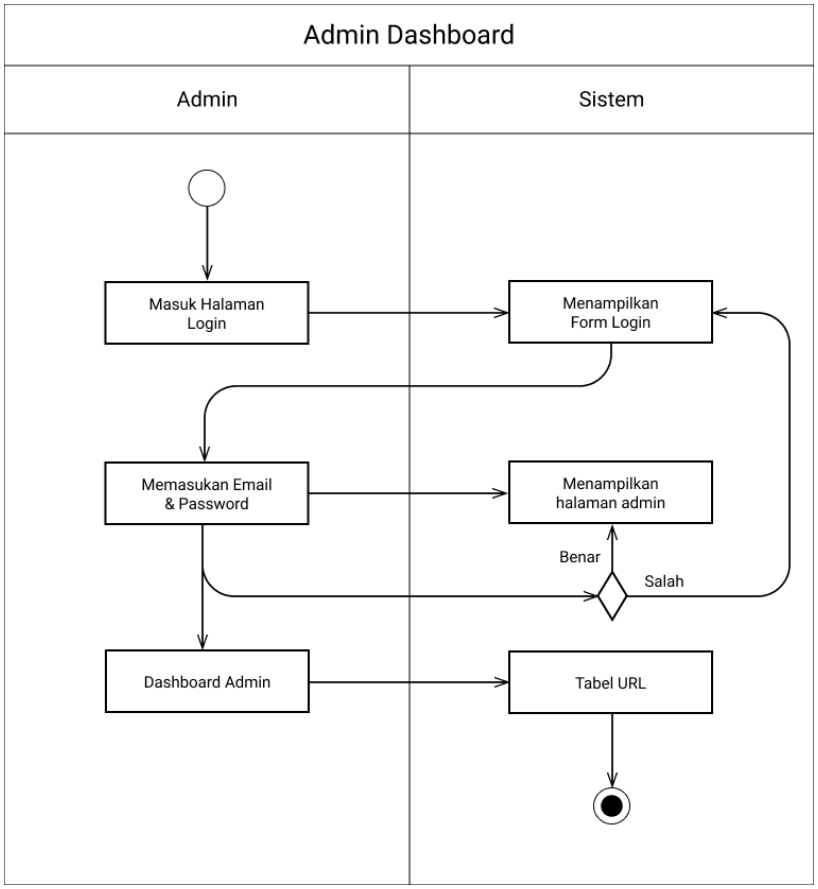
Pada gambar 2.18 dijelaskan secara visual pada halaman *landing page* bagaimana *user* atau admin dan sistem beraktifitas pada aplikasi.



Gambar 2. 18. Diagram Activity Landing Page

c. *Diagram Activity Admin Dashboard*

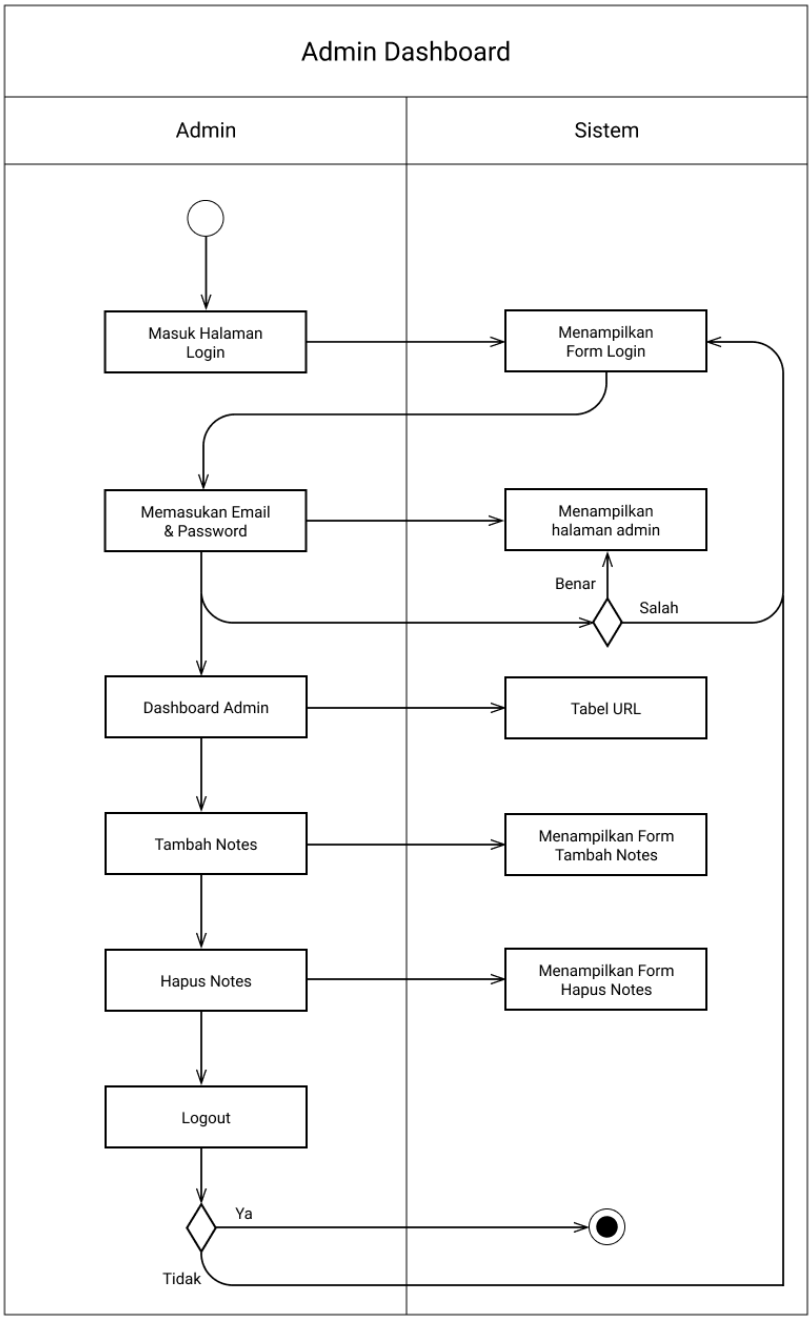
Pada gambar 2.19 dijelaskan secara visual pada halaman yang perlu diakses oleh admin untuk masuk ke dalam halaman admin *dashboard.*



Gambar 2. 19. Diagram Activity Admin Dashboard

d. *Diagram Activity Admin Logout*

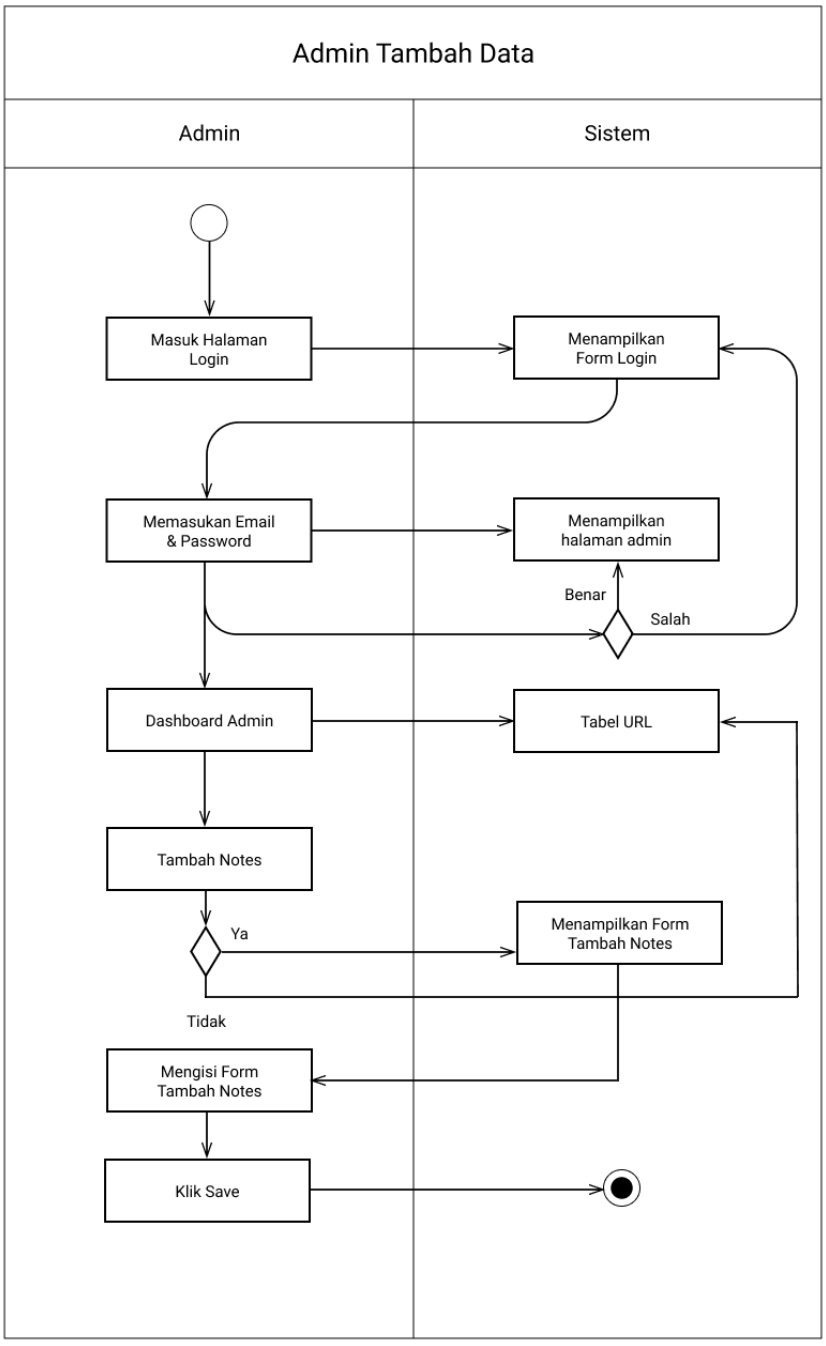
Pada gambar 2.20 dijelaskan secara visual pada halaman yang perlu diakses oleh admin untuk masuk ke dalam halaman admin *dashboard* dan admin melakukan *logout.*



Gambar 2. 20. Diagram Activity Admin Logout

e. *Diagram Activity* Tambah Catatan

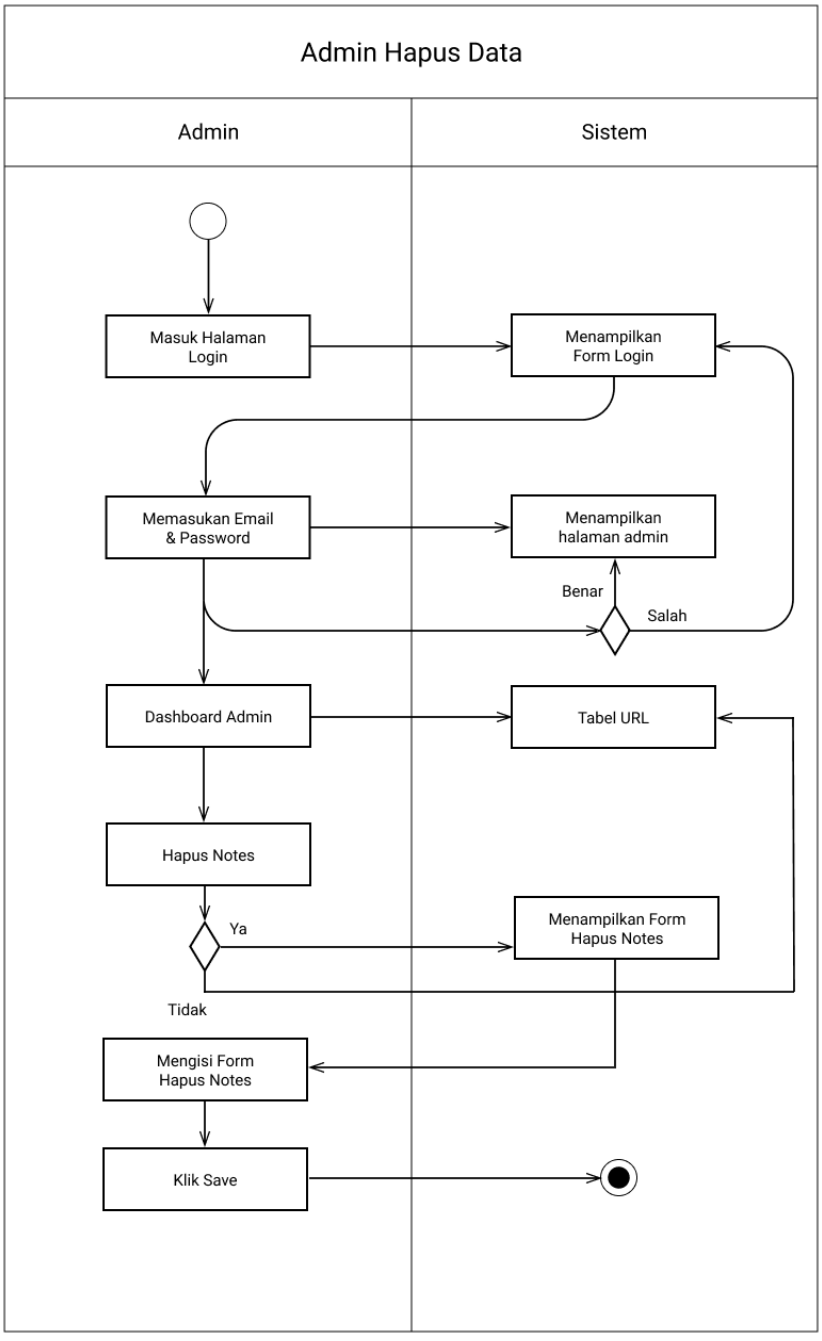
Pada gambar 2.21 dijelaskan secara visual pada halaman yang perlu diakses oleh admin untuk masuk ke dalam halaman admin *dashboard* lalu admin dapat melakukan tambah catatan.



Gambar 2. 21. Diagram Activity Tambah Catatan

f. *Diagram Activity* Hapus Catatan

Pada gambar 2.22 dijelaskan secara visual pada halaman yang perlu diakses oleh admin untuk masuk ke dalam halaman admin *dashboard* lalu admin dapat melakukan hapus catatan.



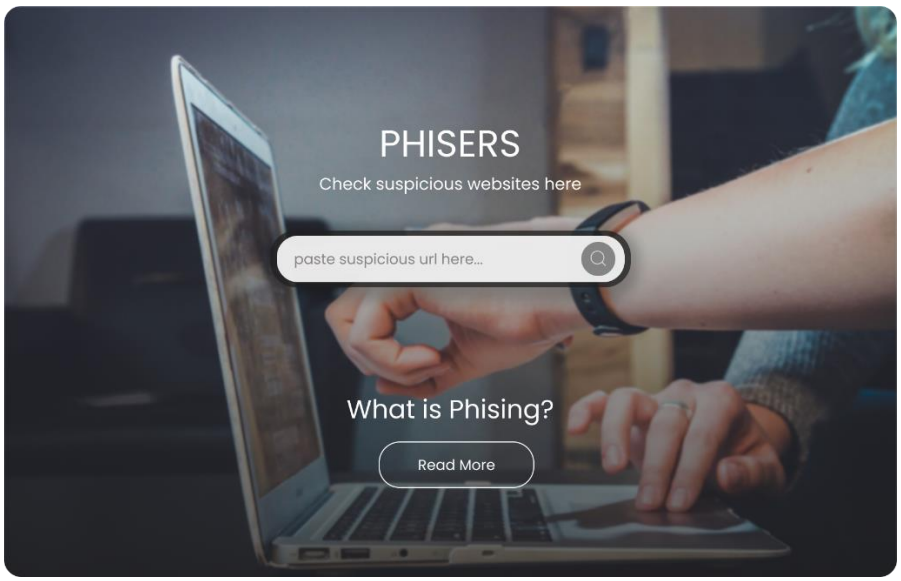
Gambar 2. 22. Diagram Activity Hapus Catatan

2.1.14. Desain Antarmuka (*Interface*)

Aplikasi Deteksi *Website Phising* "Phisers" memiliki beberapa desain sistem atau *User Interface* (UI) adalah seperti berikut :

a. *Landing Page*

Seperti pada gambar 2.23 yang memperlihatkan halaman *landing page* terdapat *icon* di atas kanan untuk *login* admin dan terdapat nama produk aplikasi ini yaitu "Phisers" dan ada *form input* url untuk mengecek url tersebut dan ada tombol *read more* untuk melihat definisi *phising*.

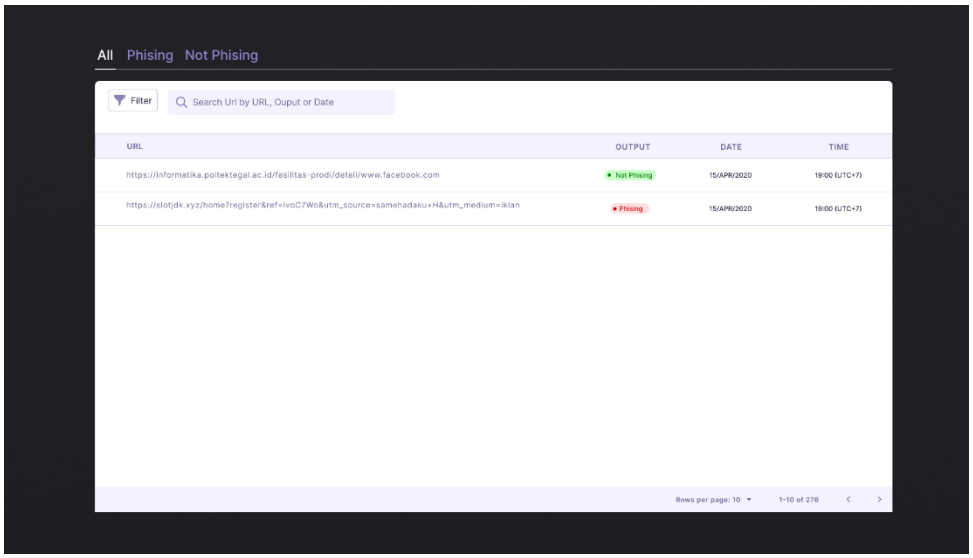


Gambar 2. 23. UI Landing Page

b. Halaman *History*

Pada halaman 2.24 memuat tabel *history* yang dimana *user* dapat melihat url yang baru saja dan yang sudah pernah dicek

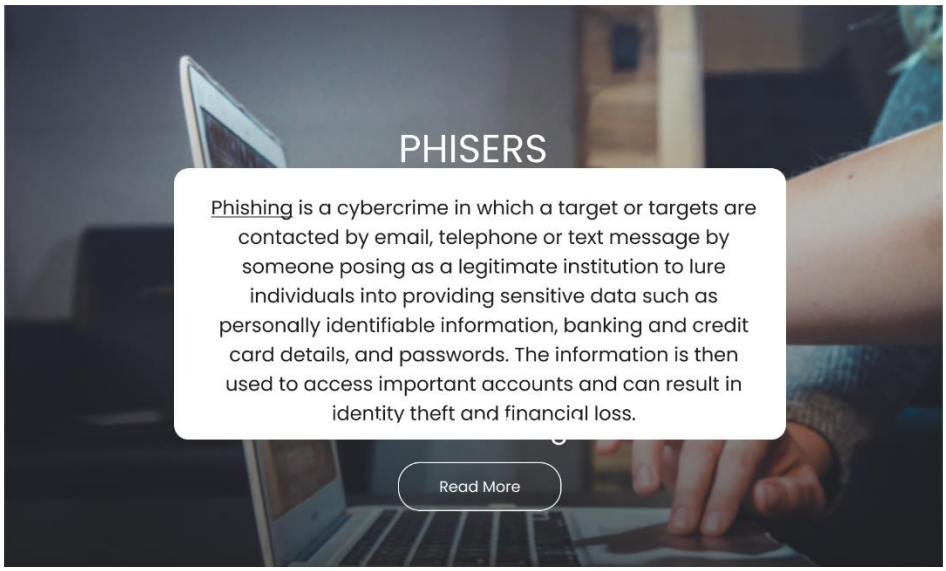
oleh *user* lain dan disertai informasi seperti *output* yang berupa pengklasifikasian url dan waktu saat *user* mengecek url.



Gambar 2. 24. UI Tabel History

c. About Phishing

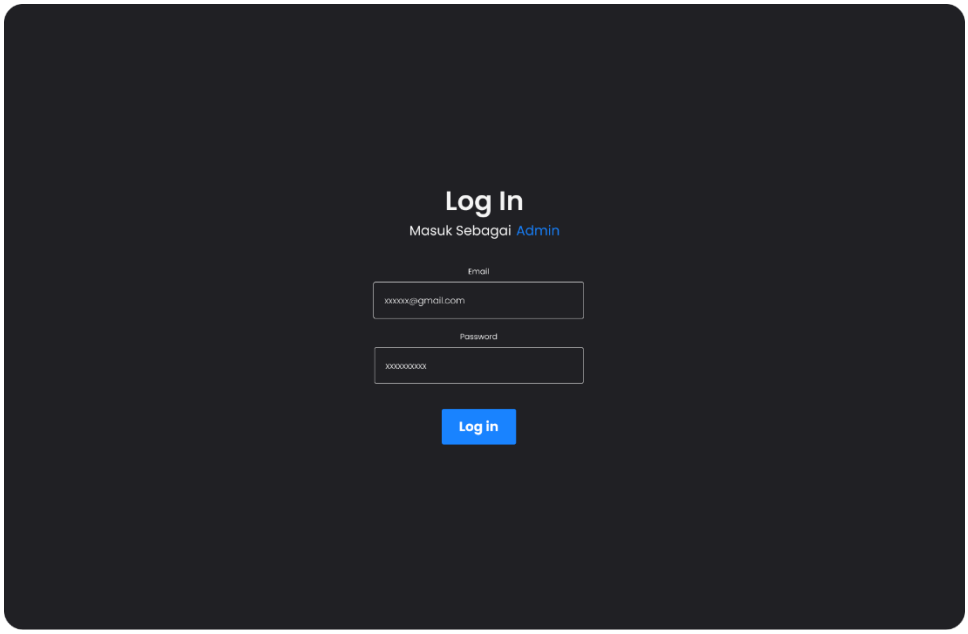
Pada gambar 2.25 halaman ini menjelaskan definisi *phising* yang tampilannya berbentuk *pop up.*



Gambar 2. 25. UI About Phising

d. Login Page

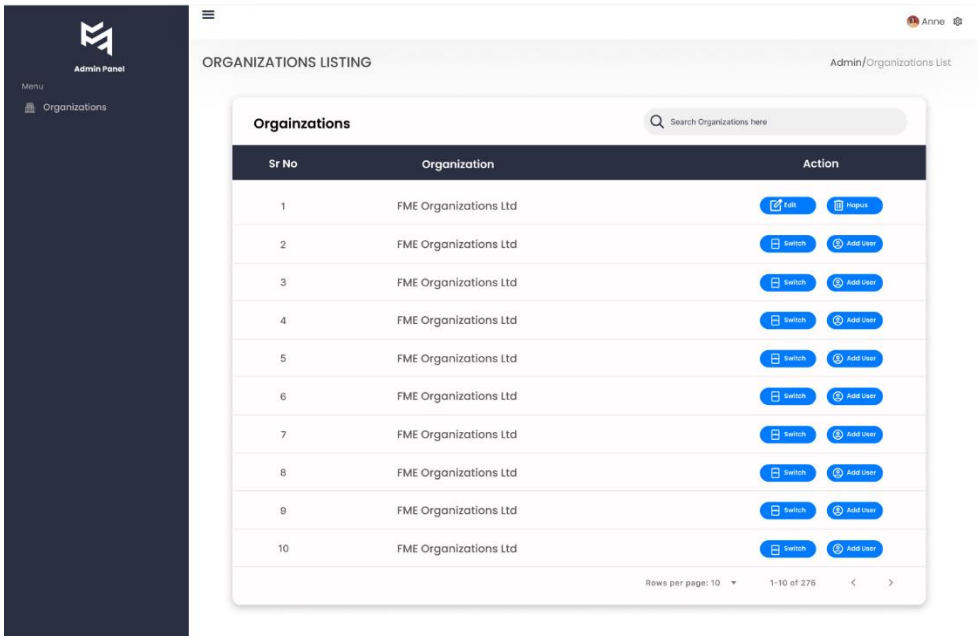
Pada gambar 2.26 adalah halaman *login* untuk mengakses halaman *dashboard* admin terdapat *form input email* dan *password*.



Gambar 2.26. UI Login Page

e. Dashboard Admin

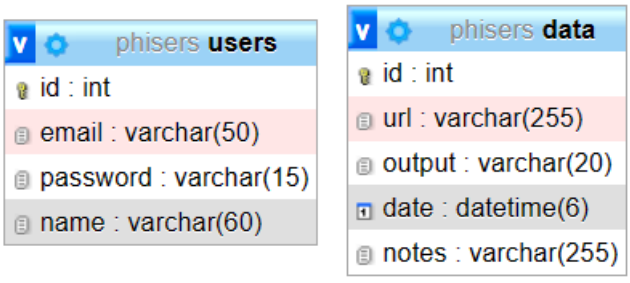
Pada gambar 2.27 adalah halaman *dashboard* admin, dimana admin dapat menambah catatan dan menghapus catatan serta ada tombol *logout* untuk kembali ke halaman *login*.



Gambar 2. 27. UI Dashboard Admin

2.1.15. *Database MySQL*

Aplikasi ini menggunakan sistem *database* MySQL dengan rincian tabel seperti ‘data’ yang berisi kumpulan url yang dicek oleh *user* dan tabel ‘users’ yang berisi *email* dan *password* untuk admin *login* terlihat seperti gambar 2.28 dibawah ini.

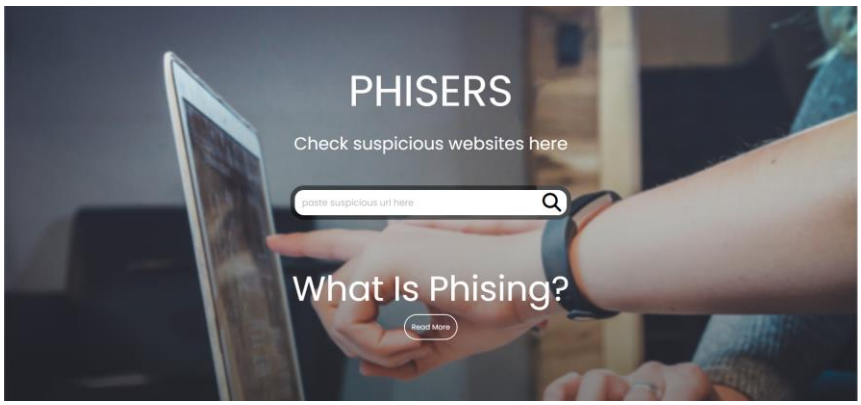


Gambar 2. 28. Desain Database

2.1.16. Implementasi Model Ke dalam *Website*

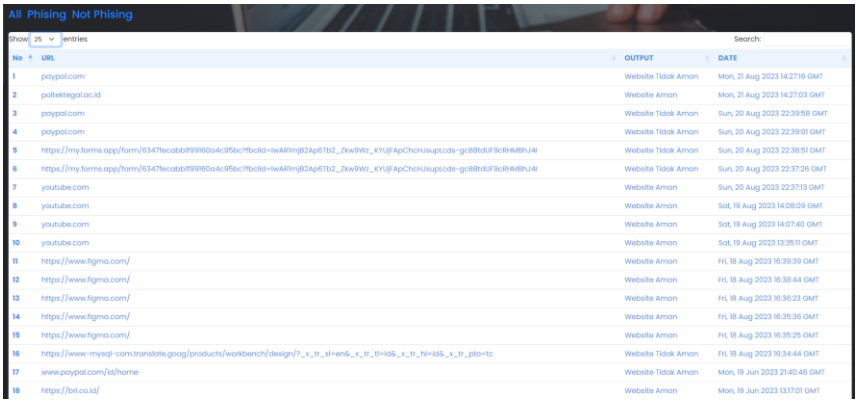
Tahapan implementasi model ke dalam *website* ini dilakukan setelah model sudah selesai pada tahapan *testing* model, di tahap ini pengimplementasian model dilakukan pada *framework flask*, beberapa hal yang perlu disiapkan adalah tampilan dari *website* dengan membuat 3 tampilan utama berikut halamannya :

1. Halaman *landing page* sebagai halaman utama dan pada halaman itu *user* dapat melakukan pendeteksian url seperti pada gambar 2.29.



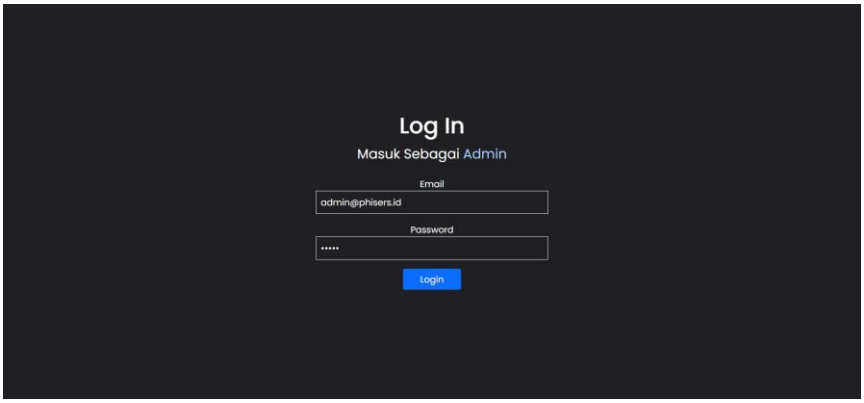
Gambar 2. 29. Tampilan Landing Page

Pada halaman *landing page* juga terdapat tabel yang berfungsi untuk melihat *history* pengecekan yang dilakukan oleh *user* tampilannya dapat dilihat pada gambar 2.30



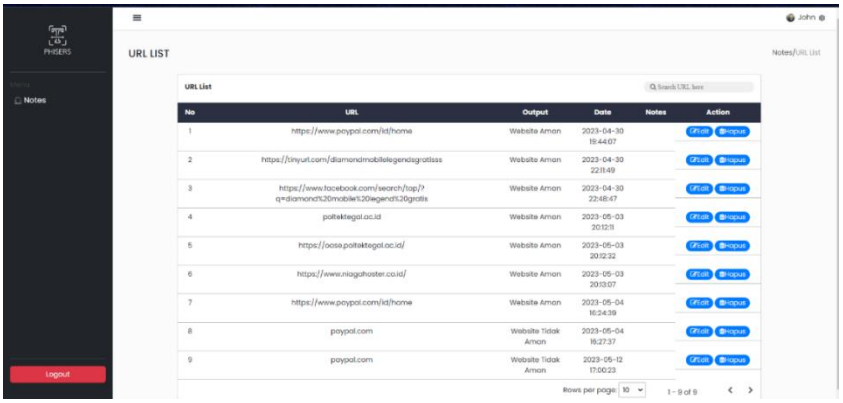
Gambar 2. 30. Tampilan Tabel History

2. Halaman *login* sebagai halaman yang harus diakses oleh admin jika ingin masuk ke dalam halaman *dashboard* admin tampilannya terlihat seperti pada gambar 2.31.



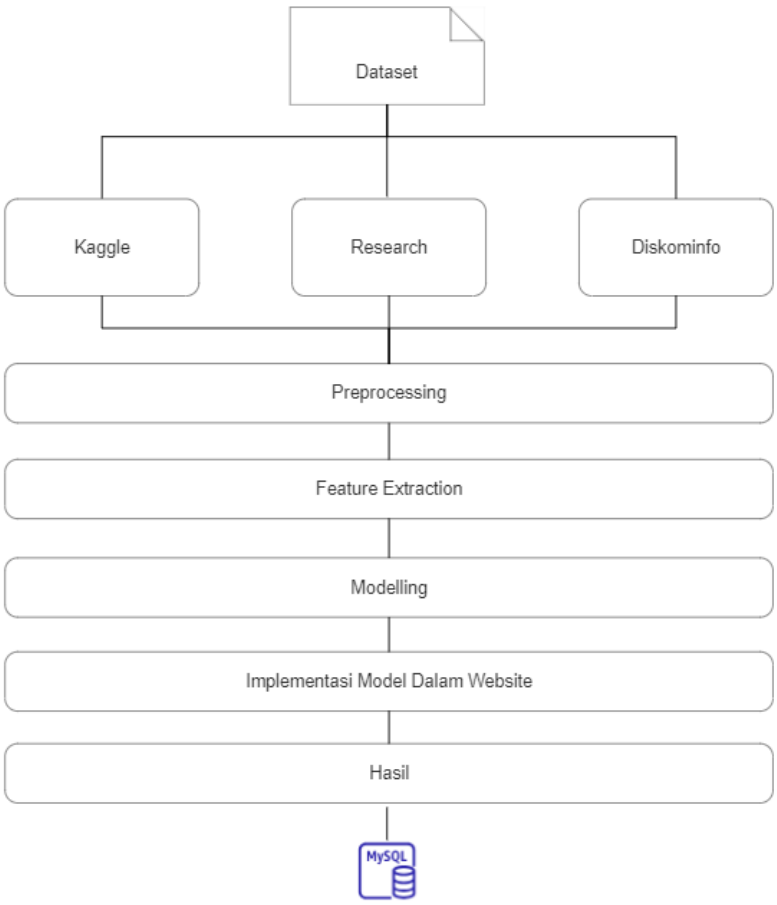
Gambar 2. 31. Tampilan Login

3. *Dashboard* admin sebagai halaman yang hanya bisa diakses oleh admin, berfungsi untuk menambahkan dan menghapus *notes* pada tiap url tampilannya dapat dilihat seperti pada gambar 2.32.



Gambar 2. 32. Tampilan Dashboard Admin

Setelah model jadi dan siap untuk digunakan untuk menjadi otak klasifikasi dari sebuah sistem maka alur dari implementasi tersebut dapat dilihat seperti pada gambar 2.33.



Gambar 2. 33. Alur Implementasi Model

2.2. Kesimpulan dan Saran

2.2.1. Kesimpulan

Tujuan utama penulis dalam penelitian ini adalah mengembangkan aplikasi deteksi penipuan berbasis web yang efisien dan akurat dengan menggunakan metode *naive bayes*. Ancaman serangan *phising* yang terus meningkat telah menjadi masalah baik bagi individu maupun bisnis. Oleh karena itu, sangat penting bagi penulis untuk memiliki alat yang andal untuk melindungi pengguna dari penipuan semacam itu.

Selama penelitian, penulis mengumpulkan kumpulan data besar yang mencakup contoh situs web yang *good* dan *bad*. Langkah *preprocessing* data melibatkan ekstraksi dan pemilihan fitur untuk mempersiapkan dataset untuk klasifikasi *Naive Bayes*. penulis memilih metode *naive bayes* karena telah terbukti mampu menyelesaikan masalah klasifikasi dan implementasi dengan relatif mudah. Pada fase percobaan, penulis melatih model klasifikasi *naive bayes* menggunakan kumpulan data yang telah diproses sebelumnya. Performa dievaluasi menggunakan berbagai metrik termasuk *score precision*, *accuracy*, *recall*, dan *F1*. Hasil evaluasi menunjukkan bahwa program deteksi *phising* berbasis web dengan metode *Naive Bayes* dapat memberikan hasil yang memuaskan dalam mengidentifikasi situs *phising*.

Singkatnya, dalam penelitian ini penulis berhasil mengembangkan aplikasi deteksi *phising* berbasis web dengan menggunakan metode *naive bayes*. Hasil percobaan menunjukkan bahwa program ini dapat memberikan akurasi yang tinggi dalam mengidentifikasi situs *phising*. Diharapkan aplikasi ini akan membantu pengguna lebih berhati-hati terhadap kemungkinan serangan *phising* dan meningkatkan kesadaran akan bahaya penipuan *online*.

2.2.2. Saran

Berdasarkan penelitian yang dilakukan penulis mengenai aplikasi pendeteksi situs *phising* berbasis *web* menggunakan metode *naive bayes*, terdapat beberapa rekomendasi yang dapat diberikan untuk pengembangan dan perbaikan lebih lanjut. Berikut adalah hal yang dapat dikembangkan pada penelitian berikutnya :

1. Memperluas dataset dengan mengumpulkan lebih banyak, dengan meningkatnya jumlah data dapat membantu meningkatkan performa model klasifikasi dan mengurangi risiko *overfitting*.
2. Menambah variasi fitur yang relevan untuk meningkatkan deteksi. Misalnya, melibatkan fitur seperti analisis tautan, analisis teks, dan analisis meta data untuk memperkaya informasi yang digunakan dalam proses klasifikasi.

3. Penggunaan metode *preprocessing* yang lebih kompleks dengan melakukan eksplorasi metode *preprocessing* yang lebih canggih untuk meningkatkan kualitas data.
4. Pengujian yang lebih kompleks, termasuk menggunakan kumpulan data aktual dan mensimulasikan pengecekan url *phising* yang lebih kompleks untuk menguji keandalan aplikasi.
5. Peningkatan *user interface* agar lebih mudah dipahami dan digunakan oleh pengguna tanpa pengetahuan teknis serta tampilan yang lebih mengikuti zaman.

Dengan menerapkan saran di atas, diharapkan penelitian ini dapat memberikan sumbangan yang lebih berarti bagi pengembangan aplikasi pendeteksi *phising* berbasis web menggunakan metode *Naive Bayes* dan membantu meningkatkan keamanan pengguna di dunia digital.

BAB III

HAK KEKAYAAN INTELEKTUAL (HKI)

3.1. Proses

Pendaftaran Hak Kekayaan Intelektual (HKI) dapat diproses setelah mengumpulkan dokumen persyaratan seperti manual book atau dokumen teknikal, KTP pemohon, surat pengalihan dan surat pernyataan yang sebelumnya telah disetujui dan ditanda tangani oleh Dosen Pembimbing I dan II serta ketua P3M.

3.2. Identitas HKI

Identitas HKI "Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naïve Bayes" dapat dilihat dibawah ini :

Nomor	: EC00202350672
Tanggal Dikeluarkan	: 30 Juni 2023
Nama Pencipta	: 1. Mohammad Zaidan Zufar 2. Dega Surono Wibowo, S.T., M.Kom. 3. M. Nishom, M.Kom.
Nama Pemegang Hak Cipta	: Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama
Jenis Ciptaan	: Program Komputer
Judul Ciptaan	: Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naïve Bayes
URL Bukti	: https://e-hakcipta.dgip.go.id/index.php/c?code=ZDc3NmMyNjVkZGI3ZDgwMTk4OTY0ZWJkNzc4MWY2ZGYK

DAFTAR PUSTAKA

[1] T. Salim and Y. C. Giap, “Data Mining Identifikasi Website Phising Menggunakan Algoritma C4.5,” vol. 8, p. 5, 2017.

[2] D. Wahyudi, M. Niswar, and A. A. P. Alimuddin, “Website Phising Detection Application Using Support Vector Machine (SVM),” vol. 5, no. 2.

[3] Z. Efendy, I. E. Putra, and R. Saputra, “Asset Rental Information System And Web-Based Facilities At Andalas University,” *J. Terap. Teknol. Inf.*, vol. 2, no. 2, pp. 135–146, Feb. 2019, doi: 10.21460/jutei.2018.22.103.

[4] M. H. Wibowo and N. Fatimah, “Ancaman Phishing Terhadap Pengguna Sosial Media Dalam Dunia Cyber Crime,” vol. 1, p. 5.

[5] D. Rachmawati, “Phising Sebagai Salah Satu Bentuk Ancaman Dalam Dunia Cyber,” vol. 13, p. 8, 2014.

[6] W. R. Yulifa, “Karya Tulis Ilmiah Penegakan Hukum Pidana Serangan Phising Pada Layanan Online Banking”.

[7] A. S. Gulo, S. Lasmadi, and K. Nawawi, “Cyber Crime dalam Bentuk Phising Berdasarkan Undang-Undang Informasi dan Transaksi Elektronik,” *PAMPAS J. Crim. Law*, vol. 1, no. 2, pp. 68–81, Apr. 2021, doi: 10.22437/pampas.v1i2.9574.

[8] A. Fatkhurohman and E. Pujastuti, “Penerapan Algoritma Naïve Bayes Classifier Untuk Meningkatkan Keamanan Data Dari Website Phising,” p. 10, 2019.

[9] A. Damuri, U. Riyanto, H. Rusdianto, and M. Aminudin, “Implementasi Data Mining dengan Algoritma Naïve Bayes Untuk Klasifikasi Kelayakan Penerima Bantuan Sembako,” vol. 8, no. 6, p. 7, 2021.

[10] J. Pardede, “Deteksi Komentar Cyberbullying Pada Media Sosial Berbahasa Inggris Menggunakan Naïve Bayes Classification,” *J. Inform.*, vol. 7, no. 1, pp. 46–54, Apr. 2020, doi: 10.31311/ji.v7i1.6920.

[11] L. A. Andika, P. A. N. Azizah, and R. Respatiwulan, “Analisis Sentimen Masyarakat terhadap Hasil Quick Count Pemilihan Presiden Indonesia 2019 pada Media Sosial Twitter Menggunakan Metode Naive Bayes Classifier, *Indones. J. Appl. Stat.*, vol. 2, no. 1, p. 34, Jul. 2019, doi: 10.13057/ijas.v2i1.29998.

[12] A. Rafi, M. Nasrun, and R. Astuti, “Deteksi Ujaran Ancaman Berbasis Website Pada Postingan Media Sosial Twitter Menggunakan Metode Naive Bayes,” vol. 8, no. 1, p. 6, 2021.

[13] A. Saleh, “Implementasi Metode Klasifikasi Naïve Bayes Dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga,” vol. 2, no. 3, 2015.

[14] H. Mustofa and A. A. Mahfudh, “Klasifikasi Berita Hoax Dengan Menggunakan Metode Naive Bayes,” *Walisongo J. Inf. Technol.*, vol. 1, no. 1, p. 1, Nov. 2019, doi: 10.21580/wjit.2019.1.1.3915.

[15] R. N. Devita, H. W. Herwanto, and A. P. Wibawa, “Perbandingan Kinerja Metode Naive Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel

Berbahasa indonesia," *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 5, no. 4, p. 427, Oct. 2018, doi: 10.25126/jtiik.201854773.

[16] A. N. Rohman, E. Utami, and S. Raharjo, "Deteksi Kondisi Emosi pada Media Sosial Menggunakan Pendekatan Leksikon dan Natural Language Processing," *Eksplora Inform.*, vol. 9, no. 1, pp. 70–76, Sep. 2019, doi: 10.30864/eksplora.v9i1.277.

LAMPIRAN

Lampiran 1. Surat Kesepakatan Bimbingan Skripsi

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :
Pihak Pertama

Nama	: Mohammad Zaidan Zufar
NIM	: 19090027
Program Studi	: Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama	: Dega Surono Wibowo, S.T., M.Kom.
Status	: Dosen
NIDN	: 0607108202
Jabatan Fungsional	: Lektor
Pangkat/Golongan	: III/C

Pada hari ini Senin tanggal 6 Maret 2023 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan syarat melakukan bimbingan 1 kali dalam 1 minggu atau setidak-tidaknya 3 kali bimbingan dalam 1 bulan (dengan progres) apabila saya tidak memenuhi syarat tersebut maka saya tidak berhak meminta surat rekomendasi mengikuti sidang skripsi dan saya berjanji memenuhi persyaratan tersebut dan menyelesaikan tepat waktu.
Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi.

Tegal, 6 Maret 2023

Pihak Pertama	Pihak Kedua
Mohammad Zaidan Zufar	Dega Surono Wibowo, S.T., M.Kom.
NIM. 19090027	NIPY. 06.014.183

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika

Slamet Riyono, S.Pd., M.Eng.
NIPY. 08.015.222

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama	: Mohammad Zaidan Zufar
NIM	: 19090027
Program Studi	: Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama	: M. Nishom, M.Kom.
Status	: Dosen
NIDN	: 0619048701
Jabatan Fungsional	: Lektor
Pangkat/Golongan	: Penata III/C

Pada hari ini Senin tanggal 6 Maret 2023 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat :

1. Pihak Pertama bersedia melaksanakan bimbingan (dengan progress) minimal 1 kali bimbingan dalam seminggu.
2. Apabila Pihak Pertama tidak dapat melaksanakan persyaratan pada poin 1 maka Pihak Pertama tidak berhak mendapatkan rekomendasi ujian sidang skripsi. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi.

Tegal, 6 Maret 2023

Pihak Pertama	Pihak Kedua
Mohammad Zaidan Zufar	M. Nishom, M.Kom.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika

Slamet Widodo, S.Pd., M.Eng.
NIPY 08.015.222

Lampiran 2. Surat Keterangan Penelitian

POLITEKNIK HARAPAN BERSAMA

Sarjana Terapan Teknik Informatika

Nomor : 82.03/TI.PHB/III/2023
Lampiran : -
Hal : Permohonan Ijin Penelitian
Kepada
Yth. : **Kepala Diskominfo Kota Tegal**
di Kota Tegal

Dengan hormat, Mahasiswa dengan identitas berikut ini:

nama : Mohammad Zaidan Zufar
NIM : 19090027
prodi : Sarjana Terapan Teknik Informatika

Bermaksud melakukan penelitian untuk keperluan Tugas Akhir dengan judul **"Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Algoritma Naïve Bayes"**. Kami memohon Bapak/Ibu memberikan Izin kepada mahasiswa yang bersangkutan agar memperoleh data, keterangan, dan bahan yang diperlukan.

Demikian permohonan ini disampaikan, Atas perhatian kami ucapkan terima kasih.

Tegal, 06 Maret 2023
Ka. Prodi S.Tr. Teknik Informatika,

Slamet Winyono, S.Pd., M.Eng
NIPY 108.015.222

Jl. Mataram No. 9 Kota Tegal 52143, Jawa Tengah, Indonesia.
(0283)452000
informatika@poltektegal.ac.id
poltektegal.ac.id

Lampiran 3. Surat Pernyataan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. N a m a : Mohammad Zaidan Zufar
   Kewarganegaraan : Indonesia
   Alamat : Desa Harjosari Lor RT/RW 023/006, Kecamatan Adiwerna, Kabupaten Tegal, Provinsi Jawa Tengah

2. Nama : Dega Surono Wibowo, S.T., M.Kom.
   Kewarganegaraan : Indonesia
   Alamat : Perumahan Sapphire Regency Blok H No.1 RT.004/RW.001, Kelurahan Pulosari, Kecamatan Brebes, 52213.

3. Nama : M. Nishom, M.Kom.
   Kewarganegaraan : Indonesia
   Alamat : Jalan Jepara Perum Griya Putri Land Blok A6, Kecamatan Margadana, Kota Tegal, Provinsi Jawa Tengah

Dengan ini menyatakan bahwa:

1. Karya Cipta yang kami mohonkan:
   Berupa : Program Komputer
   Berjudul : Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naïve Bayes
   - Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
   - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
   - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
   - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
   - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
   - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.

3. Karya Cipta yang kami mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.

4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas kami langgar, maka kami bersedia secara sukarela bahwa:
   a. permohonan karya cipta yang kami ajukan dianggap ditarik kembali; atau
   b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
   c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagimana mestinya.

Tegal, Juni 2023

Mohammad Zaidan Zufar
Pemegang Hak Cipta*

Dega Surono Wibowo, S.T., M.Kom.
Pemegang Hak Cipta*

M. Nishom, M.Kom.
Pemegang Hak Cipta*

Pemegang Hak Cipta*
* Semua pemegang hak cipta agar menandatangani di atas materai.

Lampiran 4. Surat Pengalihan HKI

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini :

1. N a m a : Mohammad Zaidan Zufar
   Kewarganegaraan : Indonesia
   Alamat : Desa Harjosari Lor RT/RW 023/006, Kecamatan Adiwerna, Kabupaten Tegal, Provinsi Jawa Tengah

2. N a m a : Dega Surono Wibowo, S.T., M.Kom.
   Kewarganegaraan : Indonesia
   Alamat : Perumahan Sapphire Regency Blok H No.1 RT.004/RW.001, Kelurahan Pulosari, Kecamatan Brebes, 52213.

3. N a m a : M. Nishom, M.Kom.
   Kewarganegaraan : Indonesia
   Alamat : Jalan Jepara Perum Griya Putri Land Blok A6, Kecamatan Margadana, Kota Tegal, Provinsi Jawa Tengah

Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada :

N a m a : Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer **"Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naïve Bayes"** untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikianlah surat pengalihan hak ini kami buat, agar dapat dipergunakan sebagaimana mestinya.

Tegal, 16 Juni 2023

Pemegang Hak Cipta
Ketua P3M

(Dr. Aldi Budi Riyanta, S.Si., M.T.)

Pencipta

(Mohammad Zaidan Zufar)

(Dega Surono Wibowo, S.T., M.Kom.)

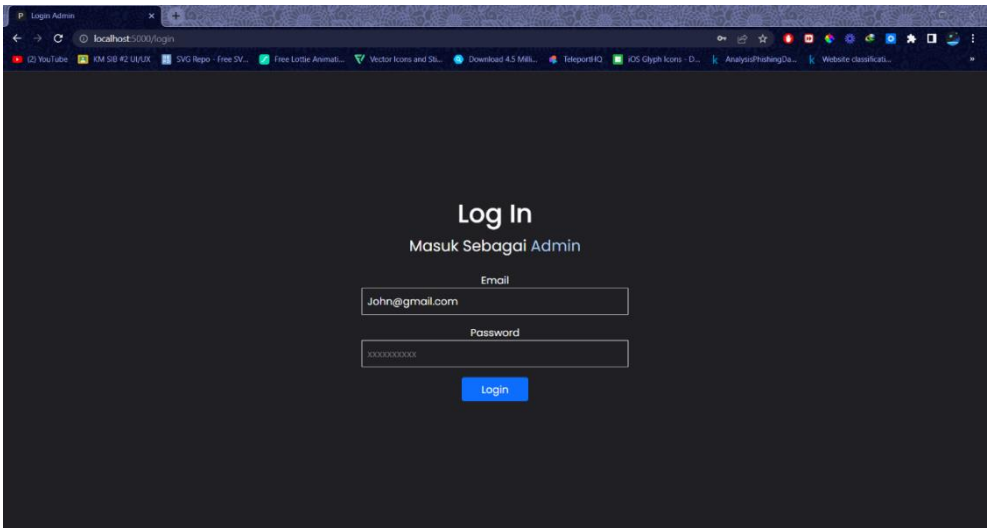
(M. Nishom, M.Kom.)

Lampiran 5. Syarat Pengajuan HKI

MANUAL BOOK

Aplikasi Pendeteksi Situs *Phising* Berbasis Website Menggunakan Metode *Naïve Bayes*

1. *Website* (Admin)
   A. *Login* Akun Admin
      1. Buka *Browser (Google Chrome, Mozila Firefox dan Microsoft Edge)*
      2. Masukan Alamat *Website* pada kolom url *browser* yaitu dengan mengetikan "www.phisers.id/login"
      3. Tekan *Enter* pada *keyboard* atau ketik tombol Go pada *browser*.
      4. Lalu akan muncul tampilan seperti pada gambar.

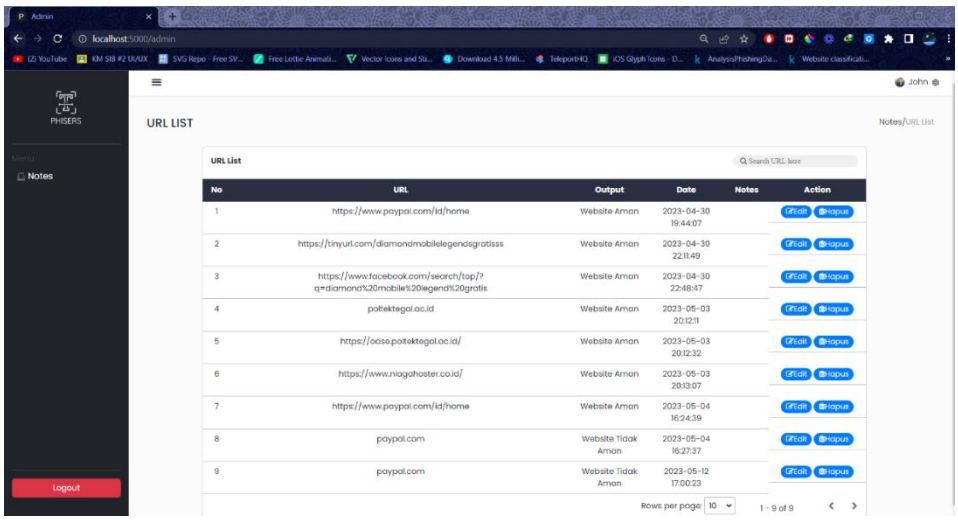


Gambar 1. 1 Halaman Login

      5. Masukan *E-mail* dan *Password* pada kolom masing-masing lalu *Enter* atau klik tombol *Login*.

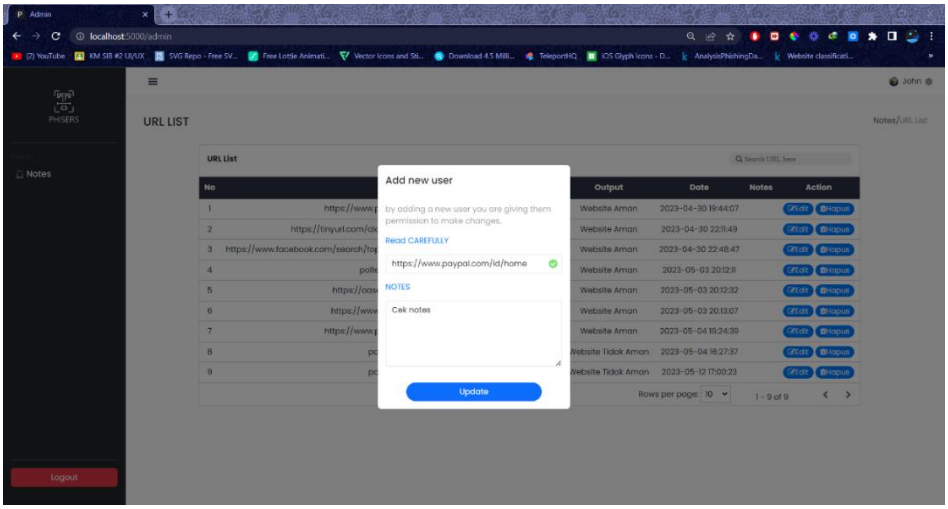
B. Tampilan *Dashboard Admin*

Tambah *Notes* dan Hapus *Notes*



Gambar 1. 2 Halaman Dashboard Admin

1. Untuk menambah *Notes* admin perlu menekan tombol edit dan mengetikan catatan apa yang perlu ditambahkan pada suatu url dan menekan tombol *update* seperti pada gambar.



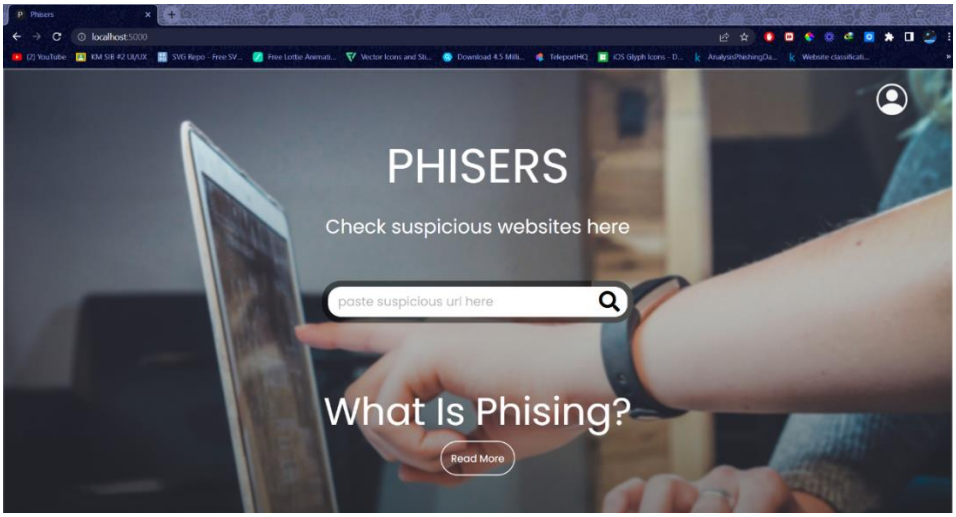
Gambar 1. 3 Input Notes dan Hapus Notes

2. Untuk menghapus catatan admin hanya perlu menekan tombol hapus yang kemudian akan secara otomatis catatan akan terhapus.

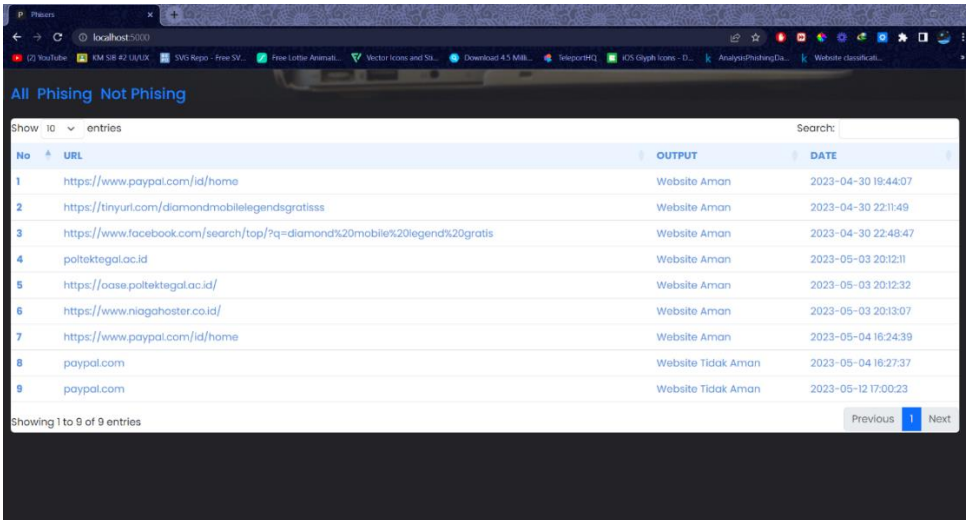
2. *Website (User)*

A. *Landing Page*

1. Buka *Browser (Google Chrome, Mozila Firefox dan Microsoft Edge)*
2. Masukan Alamat *Website* pada kolom url *browser* yaitu dengan mengetikan "www.phisers.id"
3. Tekan *Enter* pada *keyboard* atau ketik tombol Go pada *browser*.
4. Lalu akan muncul tampilan seperti pada gambar.



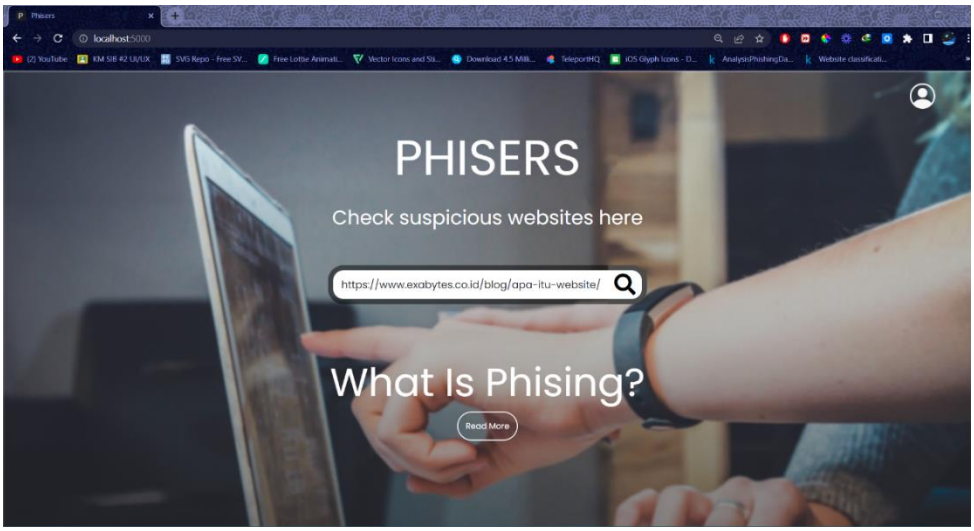
Gambar 2. 34 Halaman Landing Page 1



Gambar 2. 35 Halaman Landing Page 2

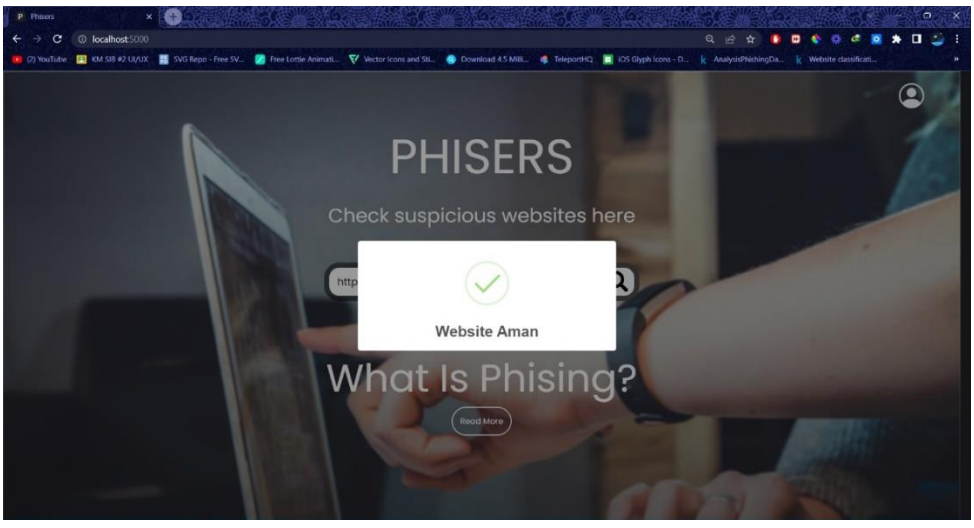
B. Mendeteksi url

1. Siapkan url yang akan dicek.
2. *Copy* url tersebut dengan kombinasi tombol Ctrl + C
3. Pastekan dengan Ctrl + V pada kolom yang ada pada *landing page website*



Gambar 2. 36 Halaman Deteksi URL

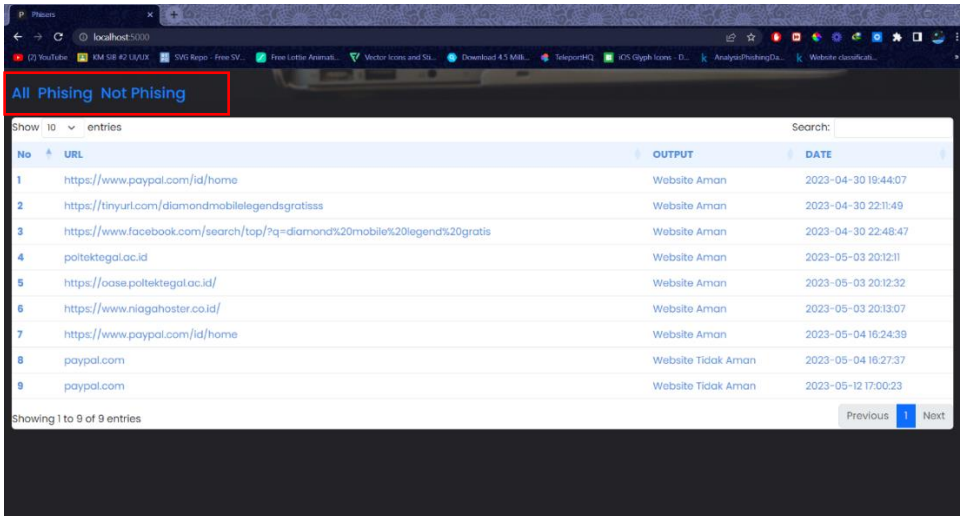
4. *Enter* atau tekan tombol logo *search* pada *website*
5. Hasil akan keluar secara otomatis seperti pada gambar



Gambar 2. 37 Output Pop Up

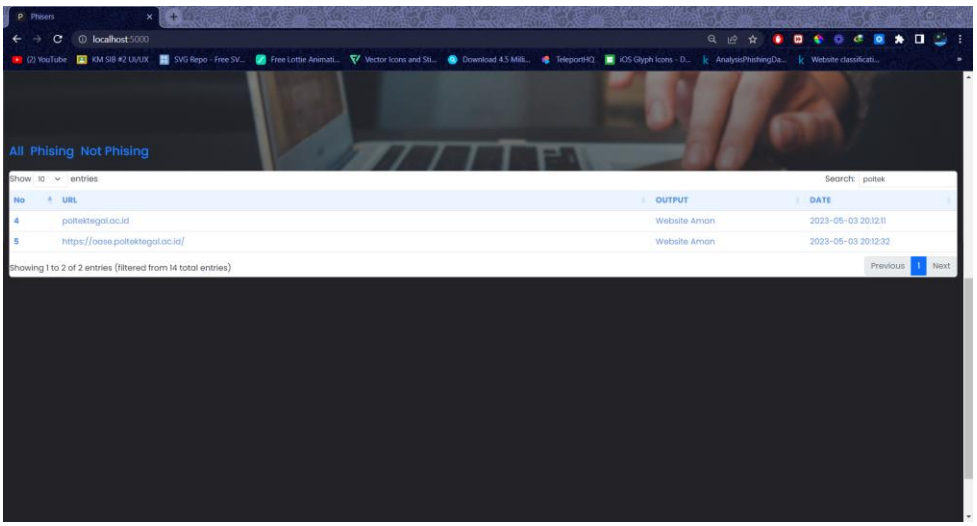
C. Melihat *history*

1. Scroll kebawah pada halaman *Landing Page*
2. Tabel dibagi menjadi 3 bagian yaitu *All*, *Phising* dan *Non-Phising*, *User* dapat memfilter ketiga jenis tersebut dengan menekan kategori yang diinginkan



Gambar 2. 38 Pilih Kategori Tabel

3. *User* juga dapat mencari secara spesifik *by name* url yang ingin dicari, *user* dapat mengetik pada kolom *search* yang tersedia maka hasil akan muncul dengan sendirinya.



Gambar 2. 39 Cari URL

DOKUMEN TEKNIKAL

1. Profil

Aplikasi Pendeteksi Situs *Phising* ini adalah sebuah aplikasi berbasis *website*, aplikasi ini memiliki fungsi untuk menangani masalah *cyber security* yaitu *phising*, cara kerja aplikasi ini adalah dengan mengecek status link/url/alamat situs yang nantinya akan terkategorikan sebagai *phising* atau bukan *phising* dengan keluarnya sebuah pop up. Platform ini menggunakan *framework flask* untuk mengintegerasikan model agar bisa berjalan di *website* serta dengan adanya histori untuk melihat *website* yang sudah pernah dicek agar *user* tahu kapan terakhir suatu *website* terakhir dicek dan apa statusnya.

2. Latar Belakang

Saat ini internet sudah menjadi bagian penting dalam kehidupan masyarakat terutama pada aktifitas sosial dan finansial. Sebagai contohnya media sosial yang digunakan sebagai sarana berkomunikasi, mencari teman dan juga bisnis *online* yang digunakan beberapa pihak terutama perusahaan untuk menawarkan perdagangan *online* melalui *e-mail* dan memberitahu kepada calon pelanggan tentang website mereka, namun saat ini ada pihak yang tidak bertanggungjawab melakukan tindakan yang merugikan banyak orang yang salah satunya adalah tindakan *phising*

Istilah *phishing* dalam bahasa Inggris datang dari kata memancing (*'fishing'*), dalam hal ini artinya memancing informasi keuangan dan kata sandi pengguna. Dengan banyaknya kasus penipuan yang dilaporkan, metode atau perlindungan tambahan sangat mendesak diperlukan. Upaya tersebut meliputi pembuatan undang-undang, pengguna pelatihan, dan tindakan teknis. *Phishing* biasanya sulit dideteksi, apalagi bagi orang awam yang tidak bergerak di bidang teknis. Masalah ini diperparah dengan penggunaan yang semakin meningkat ponsel cerdas yang biasanya tidak menampilkan URL situs web secara keseluruhan

Phising adalah aktivitas *cyber crime* yang menggunakan rekayasa sosial dan tipuan teknis untuk mencuri data identitas dan kredensial akun keuangan. Skema rekayasa sosial dilakukan dengan menggunakan *email* palsu yang mengaku berasal dari institusi bisnis yang sah dan dirancang untuk mengarahkan korban ke situs web palsu yang mengelabui, sehingga korban membocorkan data keuangan seperti : nama dan kata sandi. Skema *subterfomen* teknis menanam *crimeware* ke PC untuk mencuri kerahasiaan secara langsung, sering menggunakan sistem untuk mengelabui nama pengguna dan kata sandi akun *online* dan merusak infrastruktur navigasi lokal untuk menyesatkan konsumen ke situs web palsu (atau situs web asli melalui *proxy* yang dikendalikan *phiser* yang digunakan untuk memantau dan *intercept* pada konsumen)

Aksi *phising* ini semakin marak terjadi. Tercatat secara global, jumlah penipuan bermodus *phising* 42% dari modus selain *phising* yang dinyatakan dalam *website Anti-Phising Working Group* (APWG) dalam laporan bulannya, mencatat ada 12.845 *e-mail* baru dan unik serta 2.560 situs palsu yang digunakan sebagai sarana phising. Maka dari itu sebuah sistem pendeteksian sebuah url atau *website phising* perlu adanya, sehingga pencurian informasi data dapat diminimalisir dengan sebuah aplikasi pendeteksi url atau *website phising*.

Dalam sebuah sistem deteksi diperlukan sebuah model untuk menjalankan sebuah sistem dimana model tersebut dibuat dari sebuah metode, dalam kasus ini menggunakan *Ranfom Forest Classifier* yang merupakan salah satu metode pembelajaran mesin yang menggunakan konsep *ensemble learning* atau pembelajaran gabungan dengan membangun beberapa pohon keputusan atau *decission tree* secara acak dan menggabungkan hasil prediksi dari setiap pohon untuk menghasilkan prediksi akhir.

3. Manfaat

User akan mendapat informasi yang jelas mengenai url yang mencurigakan apakah url tersebut terindikasi *phising* atau tidak itu akan diproses melalui sistem *machine learning*.

4. Spesifikasi Teknis

Spesifikasi Teknis Meliputi :

- Soruce Code

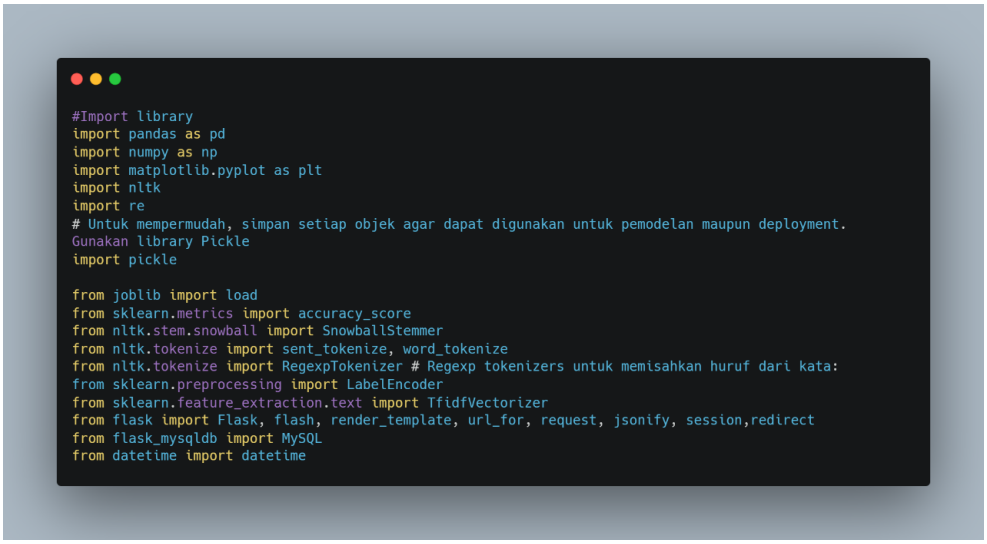
Berikut merupakan uraian spesifikasi yang digunakan untuk membangun aplikasi :

- Python
- Kaggle Notebook
- Visual Studio Code
- Xampp
- Web Browser

Source Code

- Import Library

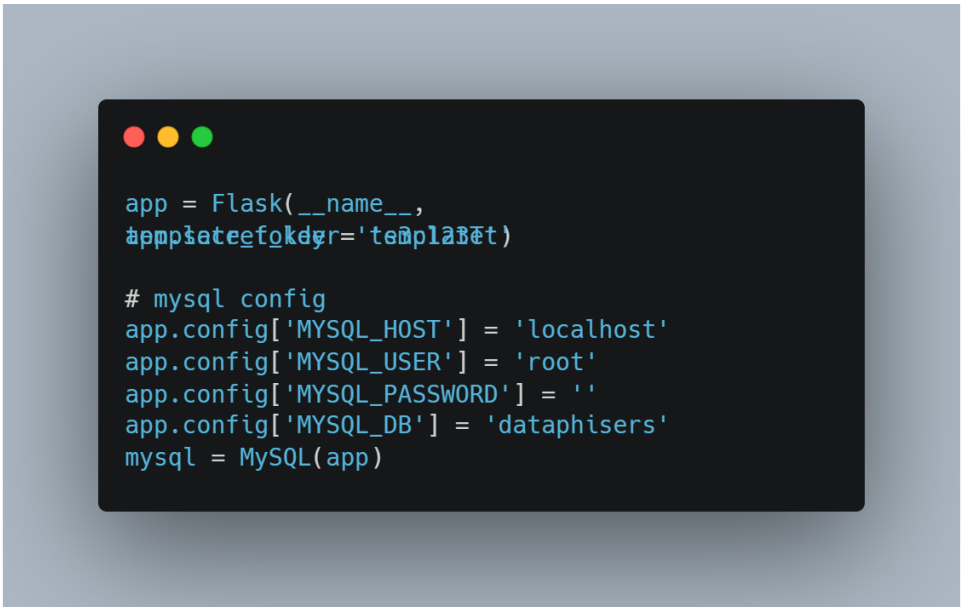
  Import library diperlukan untuk memproses program yang dijalankan di python



```
#Import library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import nltk
import re
# Untuk mempermudah, simpan setiap objek agar dapat digunakan untuk pemodelan maupun deployment.
Gunakan library Pickle
import pickle

from joblib import load
from sklearn.metrics import accuracy_score
from nltk.stem.snowball import SnowballStemmer
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.tokenize import RegexpTokenizer # Regexp tokenizers untuk memisahkan huruf dari kata:
from sklearn.preprocessing import LabelEncoder
from sklearn.feature_extraction.text import TfidfVectorizer
from flask import Flask, flash, render_template, url_for, request, jsonify, session,redirect
from flask_mysqldb import MySQL
from datetime import datetime
```

- Konfigurasi MySQL pada Flask

  Pendefinisian folder template yang digunakan untuk menyimpan file HTML dan Pengkonfigurasian MySQL pada Framework Flask



```
app = Flask(__name__,
            template_folder='template')

# mysql config
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = ''
app.config['MYSQL_DB'] = 'dataphisers'
mysql = MySQL(app)
```

- Preprocessing

  Pembuatan fungsi token untuk proses preprocessing dan import vocab.pkl, Preprocessing pada text klasifikasi digunakan untuk membersihkan dan mengubah teks mentah menjadi bentuk yang lebih sesuai untuk analisis dan pemrosesan selanjutnya sedangkan File "vocab.pkl" yang dihasilkan setelah preprocessing teks menyimpan daftar kata-kata yang muncul dalam dataset teks.

```
def token(text):
  text = text.lower()
  # Mengubah teks menjadi lower case tujuannya karena teks itu case sensitive jadi perbedaan huruf akan
    berbeda makna
  tokenizer = RegexpTokenizer(r'[A-Za-z]+')
  #untuk membersihkan teks berdasarkan karakter untuk menjadi token
  clean_text = tokenizer.tokenize(text)
  #untuk membuat token dari data yang ada
  return clean_text

sbs = SnowballStemmer("english")

vocab = pickle.load(open("./model/vocab.pkl","rb"))
```

- Fungsi landing page, admin dan logout

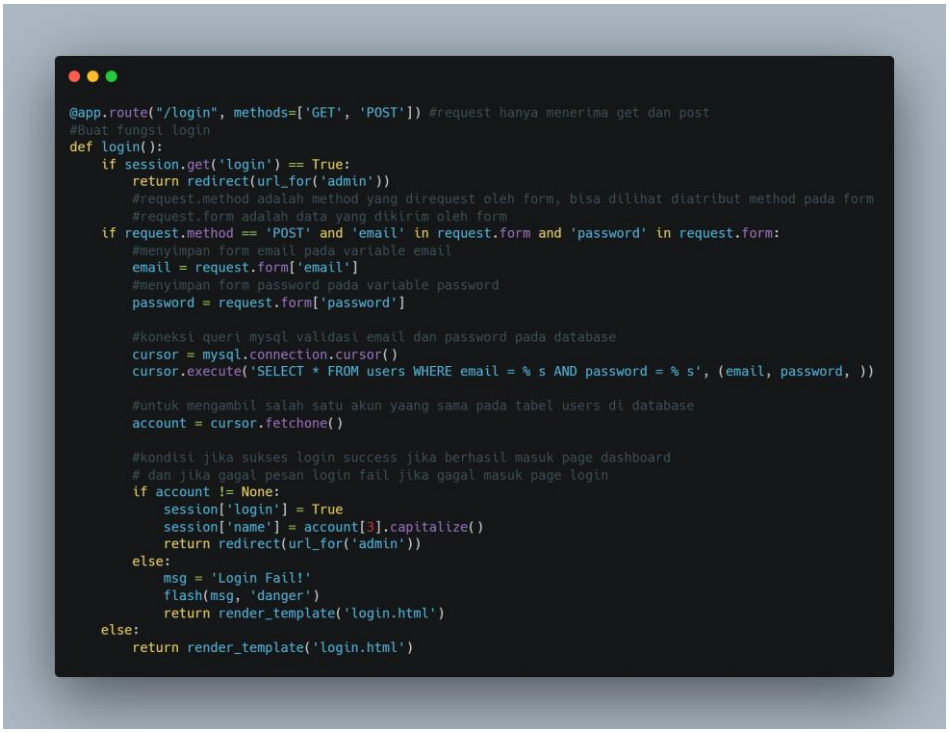
  Ada tiga bagian code dibawah :

a. def beranda adalah fungsi untuk merender template index.html dengan tampilan landing page
b. def admin adalah fungsi untuk merender template admin.html dengan tujuan tampilan ke dashboard admin tapi akan melewati tampilan login dahulu
c. def logout adalah fungsi untuk melakukan fungsi logout dan mengeluarkan halaman dashboard admin ke halaman login

```python
@app.route("/", methods=['GET'])
def beranda():
    return render_template('index.html')

@app.route("/admin", methods=['GET'])
def admin():
    if session.get('login') == True:
        return render_template('admin.html')
    else:
        return redirect(url_for('login'))

@app.route('/logout')
#membuat fungsi logout
def logout():
    #session.pop mengeluarkan atau menghapus session login atau mengubah menjadi False
    session.pop('login', False)
    #melempar ke halaman login
    msg = 'Logout Success'
    flash(msg)
    return redirect(url_for('login'))
```

- Fungsi login

  Login menggunakan email dan password dan matching terhadap data yang ada pada database maka akan masuk kehalaman dashboard admin, akan tetapi ketika user tidak memasukan data yang sesuai dengan database maka output yang dihasilkan akan tetap berada dihalaman login



```python
@app.route("/login", methods=['GET', 'POST']) #request hanya menerima get dan post
#Buat fungsi login
def login():
    if session.get('login') == True:
        return redirect(url_for('admin'))
        #request.method adalah method yang direquest oleh form, bisa dilihat diatribut method pada form
        #request.form adalah data yang dikirim oleh form
    if request.method == 'POST' and 'email' in request.form and 'password' in request.form:
        #menyimpan form email pada variable email
        email = request.form['email']
        #menyimpan form password pada variable password
        password = request.form['password']

        #koneksi queri mysql validasi email dan password pada database
        cursor = mysql.connection.cursor()
        cursor.execute('SELECT * FROM users WHERE email = % s AND password = % s', (email, password, ))

        #untuk mengambil salah satu akun yaang sama pada tabel users di database
        account = cursor.fetchone()

        #kondisi jika sukses login success jika berhasil masuk page dashboard
        # dan jika gagal pesan login fail jika gagal masuk page login
        if account != None:
            session['login'] = True
            session['name'] = account[3].capitalize()
            return redirect(url_for('admin'))
        else:
            msg = 'Login Fail!'
            flash(msg, 'danger')
            return render_template('login.html')
    else:
        return render_template('login.html')
```

- Fungsi klasifikasi

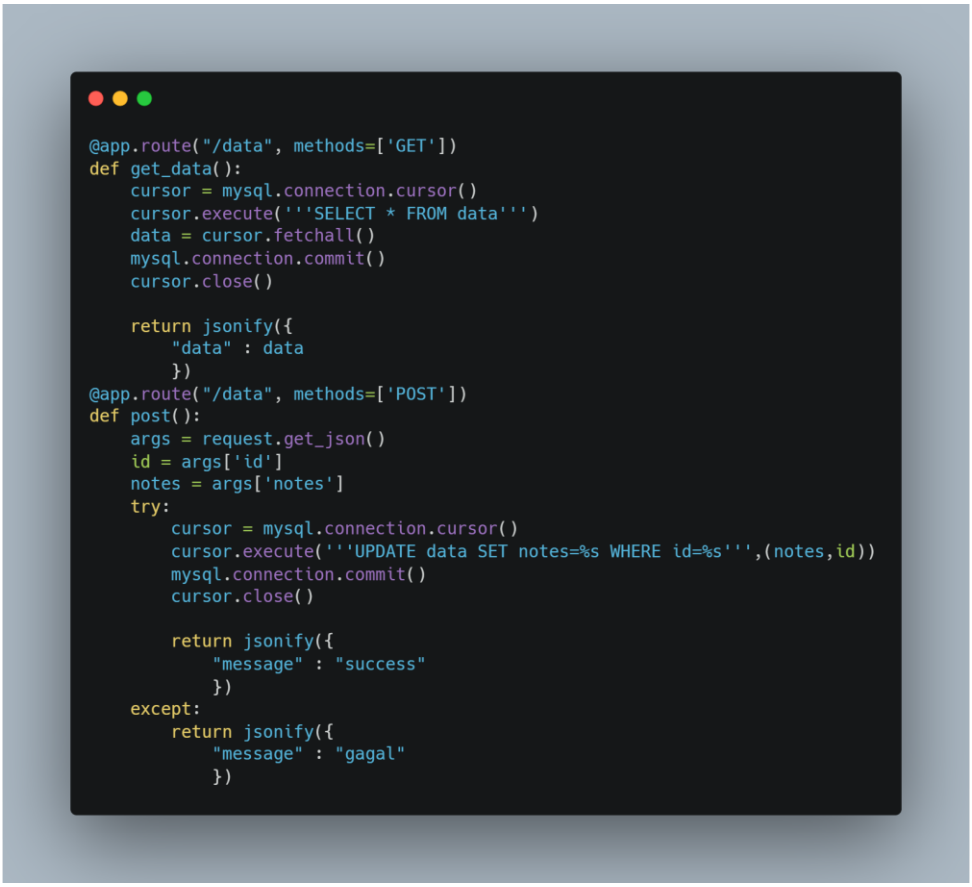
  Di fungsi klasifikasi ini hasil inputan dari form landing page itu akan di olah melalui proses tf idf dan proses pengklasifikasian dari model yang digunakan dan hasil akan menghasilkan jawaban jika hasil tersebut sama dengan 0 (bad) maka akan mengeluarkan output "Website Tidak Aman" jika hasil sama dengan 1 (good) maka akan mengeluarkan output "Website Aman" dan jika tidak keduanya maka akan mengeluarkan output "Tidak Diketahui"

```
@app.route("/klasifikasi", methods=['GET', 'POST'])
#membuat fungsi klasifikasi
def klasifikasi():
    input_text = request.get_json().get("hsl")

    pre_input_text = token(input_text)   # lakukan text pre processing pada text input
    pre_input_text = [sbs.stem(word) for word in pre_input_text]
    pre_input_text = ' '.join(pre_input_text)

    tf_idf_vec = TfidfVectorizer(vocabulary=set(vocab))

    result = model.predict(tf_idf_vec.fit_transform([pre_input_text]))  # Lakukan prediksi

    print('\nHasil Text Preprocessing :', pre_input_text)

    if (result==0):
        hsl = 'Website Tidak Aman'
    elif (result==1):
        hsl = 'Website Aman'
    else:
        hsl = 'Tidak Diketahui'

    print('\nHasil prediksi: ', input_text, ' adalah ', hsl)

    date = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    cursor = mysql.connection.cursor()
    cursor.execute('''INSERT INTO data(url,output,date) VALUES(%s,%s,%s)''',
(input_text,hsl,date))mmit()
    cursor.close()

    return jsonify({
        "Result" : hsl
        })
```

- Get dan Post data pada halaman admin

  Pada halaman dashboard admin menampilkan seluruh data url yang telah dilakukan pengecekan oleh user, pada tabel tersebut terdapat button aksi untuk mengupdate note.

  - Fungsi get data menampilkan semua data yang ada dalam tabel ‘data’
  - Fungsi post digunakan untuk mengupdate notes atau catatan pada tabel ‘data’



```python
@app.route("/data", methods=['GET'])
def get_data():
    cursor = mysql.connection.cursor()
    cursor.execute('''SELECT * FROM data''')
    data = cursor.fetchall()
    mysql.connection.commit()
    cursor.close()

    return jsonify({
        "data" : data
        })
@app.route("/data", methods=['POST'])
def post():
    args = request.get_json()
    id = args['id']
    notes = args['notes']
    try:
        cursor = mysql.connection.cursor()
        cursor.execute('''UPDATE data SET notes=%s WHERE id=%s''',(notes,id))
        mysql.connection.commit()
        cursor.close()

        return jsonify({
            "message" : "success"
            })
    except:
        return jsonify({
            "message" : "gagal"
            })
```

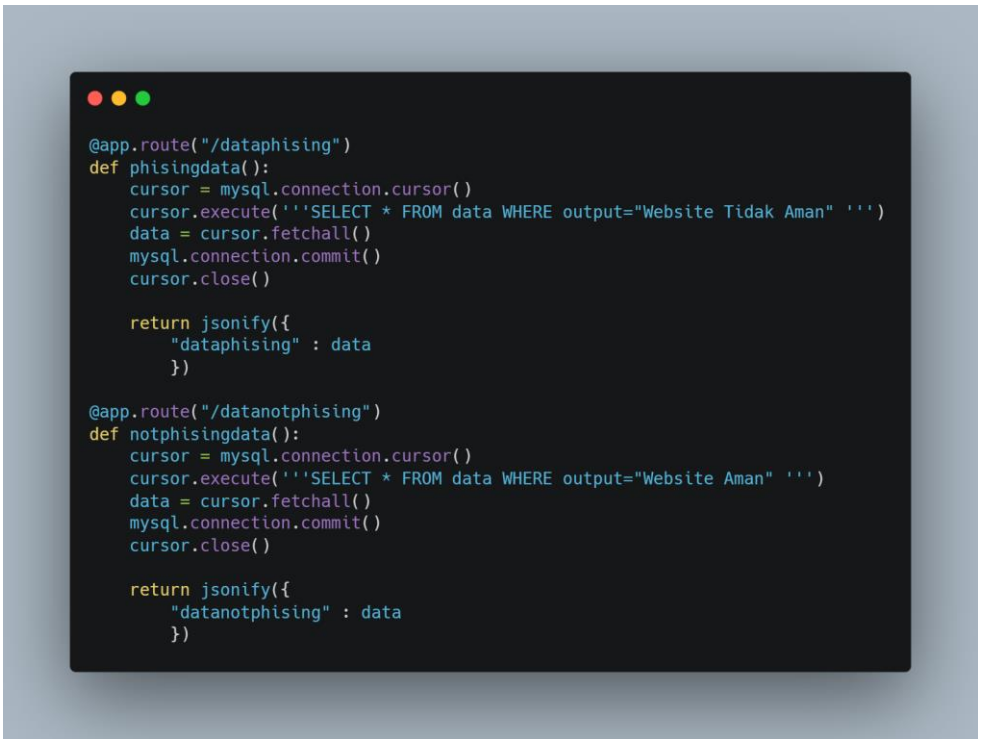
- Fungsi delete dan search

  Untuk menghapus data diperlukan sebuah fungsi yang bernama ‘hapus‘ yang berfungsi untuk menghapus data notes berdasarkan id. Sedangkan fungsi search digunakan untuk melakukan pencarian pada tabel data.

```
@app.route("/data/<int:id>")
def get(id):
    print(id)
    cursor = mysql.connection.cursor()
    cursor.execute('''SELECT * FROM data WHERE id = %s''', (id,))
    data = cursor.fetchone()
    mysql.connection.commit()
    cursor.close()

    return jsonify({
        "data" : data
        })

@app.route("/search", methods=['POST'])
def search():
    args = request.get_json()
    search = args['search'].upper()
    cursor = mysql.connection.cursor()
    cursor.execute(f'SELECT * FROM data where UPPER(url) LIKE "%
{search}%"')
    data = cursor.fetchall()
    mysql.connection.commit()
    cursor.close()

    return jsonify({
        "data" : data
        })

@app.route("/data/<int:id>", methods=['DELETE'])
def hapus(id):
    try:
        cursor = mysql.connection.cursor()
        cursor.execute('''DELETE FROM data WHERE id=%s''',(id,))
        mysql.connection.commit()
        cursor.close()

        return jsonify({
            "message" : "success"
            })
    except:
        return jsonify({
            "message" : "gagal"
            })
```

- Fungsi phisingdata dan notphisingdata

  Fungsi phising data digunakan untuk menampilkan url data yang terdeteksi webstie tidak aman sedangkan fungsi notphisingdata digunakan untuk menampilkan url data yang terdeteksi website aman



```python
@app.route("/dataphising")
def phisingdata():
    cursor = mysql.connection.cursor()
    cursor.execute('''SELECT * FROM data WHERE output="Website Tidak Aman" ''')
    data = cursor.fetchall()
    mysql.connection.commit()
    cursor.close()

    return jsonify({
        "dataphising" : data
        })

@app.route("/datanotphising")
def notphisingdata():
    cursor = mysql.connection.cursor()
    cursor.execute('''SELECT * FROM data WHERE output="Website Aman" ''')
    data = cursor.fetchall()
    mysql.connection.commit()
    cursor.close()

    return jsonify({
        "datanotphising" : data
        })
```

- Import model

  Berfungsi untuk memanggil model kedalam sistem yang akan dibangun

```python
if __name__ == "__main__":

    model = load('.\model\modelRandomForest.model')
    app.run(host="0.0.0.0", debug=True)
```

- Tampilan html login

  Tampilan halaman login akan menampilkan form input username dan password yang digunakan untuk login

```
<form action="{{ url_for('login') }}" method="POST">
    <div class="row m-0">
        <div class="col-md-12 p-0">
        Email
        </div>
    </div>
    <div class="row m-0">
        <div class="col-md-12 p-0">
            <input type="text" name="email" class="w-100 mb-3 p-2" placeholder="John@gmail.com">
        </div>
    </div>
    <div class="row m-0">
        <div class="col-md-12 p-0">
        Password
        </div>
    </div>
    <div class="row m-0">
        <div class="col-md-12 p-0">
            <input type="password" class="w-100 mb-3 p-2" name="password" placeholder="xxxxxxxxxx">
        </div>
    </div>
<button type="submit" id="login" class="btn btn-primary w-25 rounded-1">Login</button>
```

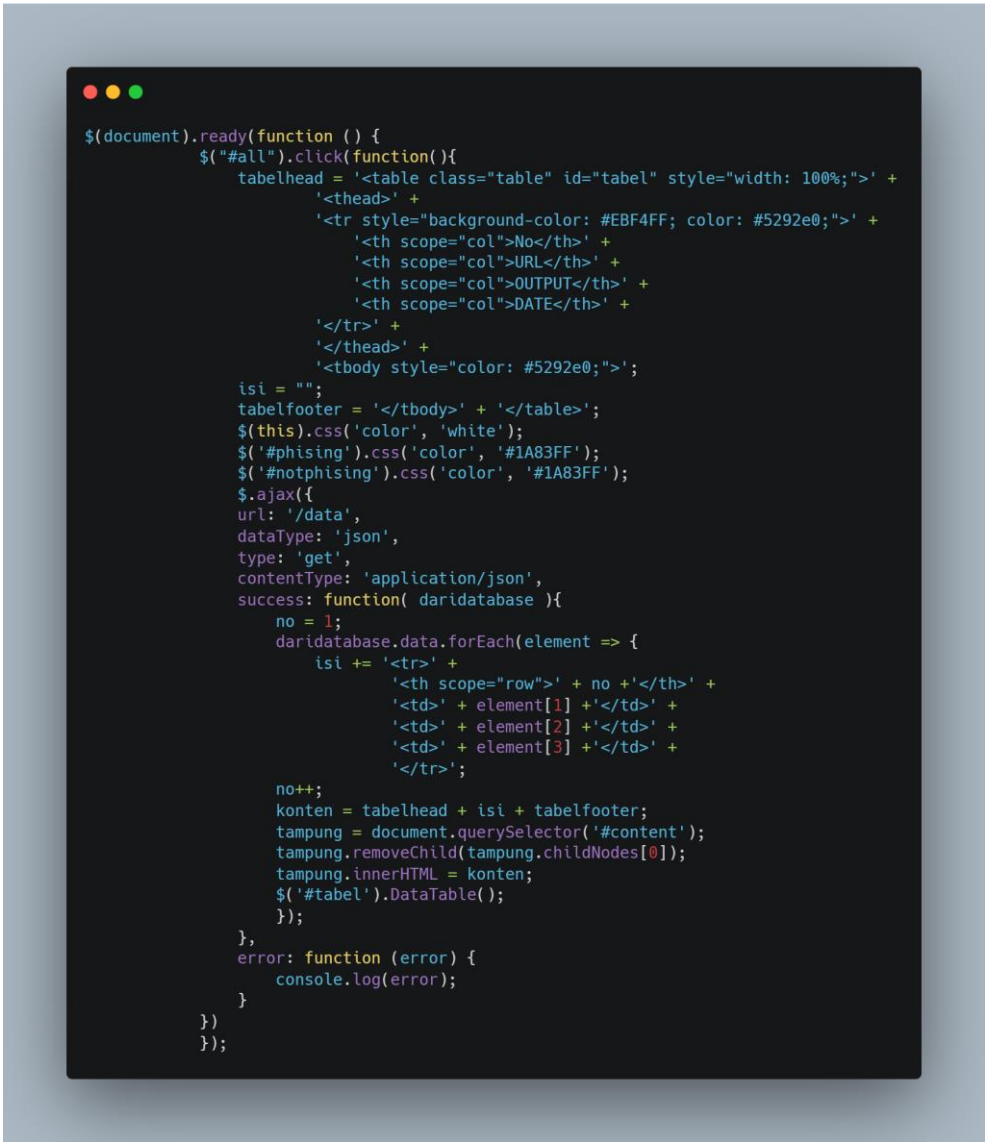
- Tampilan index.html

  Menampilkan form input untuk pengecekan url phising

```
<form action="{{ url_for('klasifikasi') }}" id="clf" method="POST" >
    <p style="color: white;font-family: Poppins; font-weight: 400; font-size: 5vw;">PHISERS</p>
    <p style="font-family: Poppins; color: white;  font-size: 2vw;">Check suspicious websites here</p>
    <div class="col-md-4  pt-5">
       <div class="input-group">
            <input style="border-top-left-radius: 30px; border-bottom-left-radius: 30px; font-family:
Poppins; border-right: 0px solid black; border-left: 10px solid rgba(0, 0, 0, 0.5); border-top: 10px
solid rgba(0, 0, 0, 0.5); border-bottom: 10px solid rgba(0, 0, 0, 0.5)" type="text" id="webUrl"
class="form-control form-control-lg" placeholder="paste suspicious url here"  aria-label="paste
suspicious url here" aria-describedby="button-addon2">
            <button style="border-top-right-radius: 30px; border-bottom-right-radius: 30px;border-
right: 10px solid rgba(0, 0, 0, 0.7); border-left: 0px solid black; border-top: 10px solid rgba(0, 0,
0, 0.7); border-bottom: 10px solid rgba(0, 0, 0, 0.7)" class="btn bg-white" id="button-addon2"
type="submit">
                <i class="fa fa-search" style=" font-size: 2.2vw; color: black;"></i>
            </button>
       </div>
   </div>
</form>
```

- Data tabel javascript

  Menampilkan tabel yang berisikan semua data yang telah dilakukan pengecekan url oleh user lain



```
$(document).ready(function () {
            $("#all").click(function(){
                tabelhead = '<table class="table" id="tabel" style="width: 100%;">' +
                        '<thead>' +
                        '<tr style="background-color: #EBF4FF; color: #5292e0;">' +
                            '<th scope="col">No</th>' +
                            '<th scope="col">URL</th>' +
                            '<th scope="col">OUTPUT</th>' +
                            '<th scope="col">DATE</th>' +
                        '</tr>' +
                        '</thead>' +
                        '<tbody style="color: #5292e0;">';
                isi = "";
                tabelfooter = '</tbody>' + '</table>';
                $(this).css('color', 'white');
                $('#phising').css('color', '#1A83FF');
                $('#notphising').css('color', '#1A83FF');
                $.ajax({
                url: '/data',
                dataType: 'json',
                type: 'get',
                contentType: 'application/json',
                success: function( daridatabase ){
                    no = 1;
                    daridatabase.data.forEach(element => {
                        isi += '<tr>' +
                                '<th scope="row">' + no +'</th>' +
                                '<td>' + element[1] +'</td>' +
                                '<td>' + element[2] +'</td>' +
                                '<td>' + element[3] +'</td>' +
                                '</tr>';
                    no++;
                    konten = tabelhead + isi + tabelfooter;
                    tampung = document.querySelector('#content');
                    tampung.removeChild(tampung.childNodes[0]);
                    tampung.innerHTML = konten;
                    $('#tabel').DataTable();
                    });
                },
                error: function (error) {
                    console.log(error);
                }
            })
            });
```

- Data tabel javascript kategori data phising

  Menampilkan tabel berisikan semua data berdasarkan kategori url yang terdeteksi phising setelah dilakukan pengecekan url oleh user lain

```
$("#phising").click(function(){
                $(this).css('color', 'white');
                $('#all').css('color', '#1A83FF');
                $('#notphising').css('color', '#1A83FF');
                tabelheadphising = '<table class="table" id="tabelphising" style="width: 100%;">' +
                        '<thead>' +
                        '<tr style="background-color: #EBF4FF; color: #5292e0;">' +
                            '<th scope="col">No</th>' +
                            '<th scope="col">URL</th>' +
                            '<th scope="col">OUTPUT</th>' +
                            '<th scope="col">DATE</th>' +
                        '</tr>' +
                        '</thead>' +
                        '<tbody style="color: #5292e0;">';
                isi = "";
                tabelfooter = '</tbody>' + '</table>';
                $.ajax({
                url: '/dataphising',
                dataType: 'json',
                type: 'get',
                contentType: 'application/json',
                success: function( daridatabase ){
                    console.log(daridatabase);
                    no = 1;
                    daridatabase.dataphising.forEach(element => {
                        isi += '<tr>' +
                                '<th scope="row">' + no +'</th>' +
                                '<td>' + element[1] +'</td>' +
                                '<td>' + element[2] +'</td>' +
                                '<td>' + element[3] +'</td>' +
                                '</tr>';
                    no++;
                    konten = tabelheadphising + isi + tabelfooter;
                    tampung = document.querySelector('#content');
                    table = document.querySelector('#tabel_wrapper');
                    tampung.removeChild(tampung.childNodes[0]);
                    tampung.innerHTML = konten;
                    $('#tabelphising').DataTable();
                    });
                },
                error: function (error) {
                    console.log(error);
                }
            })
            });
```

- Data tabel javascript kategori data notphising

  Menampilkan tabel berisikan semua data berdasarkan kategori url yang terdeteksi bukan phising setelah dilakukan pengecekan url oleh user lain

```
$("#notphising").click(function(){
                $(this).css('color', 'white');
                $('#all').css('color', '#1A83FF');
                $('#phising').css('color', '#1A83FF');
                tabelheadphising = '<table class="table" id="tabelnotphising" style="width: 100%;">' +
                        '<thead>' +
                        '<tr style="background-color: #EBF4FF; color: #5292e0;">' +
                            '<th scope="col">No</th>' +
                            '<th scope="col">URL</th>' +
                            '<th scope="col">OUTPUT</th>' +
                            '<th scope="col">DATE</th>' +
                        '</tr>' +
                        '</thead>' +
                        '<tbody style="color: #5292e0;">';
                isi = "";
                tabelfooter = '</tbody>' + '</table>';
                $.ajax({
                url: '/datanotphising',
                dataType: 'json',
                type: 'get',
                contentType: 'application/json',
                success: function( daridatabase ){
                    console.log(daridatabase);
                    no = 1;
                    daridatabase.datanotphising.forEach(element => {
                        isi += '<tr>' +
                                '<th scope="row">' + no +'</th>' +
                                '<td>' + element[1] +'</td>' +
                                '<td>' + element[2] +'</td>' +
                                '<td>' + element[3] +'</td>' +
                                '</tr>';
                    no++;
                    konten = tabelheadphising + isi + tabelfooter;
                    tampung = document.querySelector('#content');
                    table = document.querySelector('#tabel_wrapper');
                    tampung.removeChild(tampung.childNodes[0]);
                    tampung.innerHTML = konten;
                    $('#tabelnotphising').DataTable();
                    });
                },
                error: function (error) {
                    console.log(error);
                }
            })
            });
```

- Sweetalert javascript

  Digunakan untuk menampilkan pop up saat pengecekan url yang terdeteksi phising atau not phising

```
(function($){
            function kirimLink( e ){
                $.ajax({
                    url: '/klasifikasi',
                    dataType: 'json',
                    type: 'post',
                    contentType: 'application/json',
                    data: JSON.stringify( { "hsl": $('#webUrl').val() } ),
                    success: function( result ){
                        // console.log(result['Result']);
                        if (result['Result'] == 'Website Tidak Aman') {
                            Swal.fire({
                            icon: 'error',
                            title: result['Result'],
                            showConfirmButton: false,
                            })
                        }else if (result['Result'] == 'Website Aman') {
                            Swal.fire({
                            title: result['Result'],
                            icon: 'success',
                            showConfirmButton: false,
                            })
                        }else{
                            Swal.fire({
                            title: result['Result'],
                            icon: 'warning',
                            showConfirmButton: false,
                            })
                        }
                        setTimeout(function(){
                        window.location.reload(1);
                        }, 1000);
                    },
                    error: function( error ){
                        console.log( error );
                    }
                });
```

- Tombol edit notes javascript

  Berfungsi untuk mengedit data notes

```
$(document).on("click", "#editData", function () {
          const exampleModal = new bootstrap.Modal('#exampleModal')
          exampleModal.show()
          var data = new Object();
          data.id = $(this).attr("idData");
          console.log(data.id)

          $.ajax({
            url: "/data/" + data.id,
            type: "GET",
            cache: false,
            success: function (resp) {
              var result = resp.data;
              if (result[2] == 'Website Aman') {
                $("#iconFailed").css('display', 'none')
                $("#iconSuccess").css('display', 'block')
              }else if(result[2] == 'Website Tidak Aman'){
                $("#iconSuccess").css('display', 'none')
                $("#iconFailed").css('display', 'block')
              }
              $("#org").val(result[1])
              $("#notes").val(result[4])
              $("#idPhising").val(result[0])
            },
            error: function () {
              console.log("error");
            },
          });
        });
```

- Tombol update data notes Javascript

  Berfungsi untuk menyimpan data yang telah diedit

```
$(document).on("click", "#update", function () {
          var data = new Object();
          data.id = $("#idPhising").val();
          data.notes = $("#notes").val();

          $.ajax({
            url: "/data",
            type: "POST",
            dataType: 'json',
            contentType: 'application/json',
            data: JSON.stringify(data),
            cache: false,
            success: function (resp) {
              if(resp.message == 'success'){
                $("#exampleModal").modal("hide");
                const successModal = new
bootstrap.Modal('#successModal')
                successModal.show()
                $("#done").click(function () {
                  $("#successModal").modal("hide");
                })
                refreshTable()
              }else{
                $("#exampleModal").modal("hide");
                const failModal = new bootstrap.Modal('#failModal')
                failModal.show()
                $("#tryagain").click(function () {
                  $("#failModal").modal("hide");
                  $("#exampleModal").modal("show");
                })
                refreshTable()
              }
            },
            error: function () {
              console.log("error");
            },
          });
        });
```

- Tombol hapus data Javascript

  Berfungsi untuk menghapus data dari database

```
$(document).on("click", "#hapusData", function () {
          id = $(this).attr("idData");

          $.ajax({
            url: "/data/" + id,
            type: "DELETE",
            cache: false,
            success: function (resp) {
              if (resp.message == 'success') {
                refreshTable()
              }else if(result[2] == 'Website Tidak Aman'){
                refreshTable()
              }
            },
            error: function () {
              console.log("error");
            },
          });
        });
```

Lampiran 6. Sertifikat HKI

REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN
CIPTAAN

Dalam rangka pelindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan	:	EC00202350672, 30 Juni 2023
Pencipta		
Nama	:	**Mohammad Zaidan Zufar, Dega Surono Wibowo dkk**
Alamat	:	Desa Harjosari Lor RT/RW 023/006, Kecamatan Adiwerna, Tegal, Jawa Tengah, 52194
Kewarganegaraan	:	Indonesia
Pemegang Hak Cipta		
Nama	:	**Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama**
Alamat	:	Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana, Tegal, JAWA TENGAH 52142
Kewarganegaraan	:	Indonesia
Jenis Ciptaan	:	**Program Komputer**
Judul Ciptaan	:	**Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naive Bayes**
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	:	30 Juni 2023, di Tegal
Jangka waktu pelindungan	:	Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan	:	000483607

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
Direktur Hak Cipta dan Desain Industri

Anggoro Dasananto
NIP. 196412081991031002

Disclaimer:
Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Mohammad Zaidan Zufar	Desa Harjosari Lor RT/RW 023/006, Kecamatan Adiwerna
2	Dega Surono Wibowo	Perumahan Sapphire Regency Blok H No.1 RT.004/RW.001, Kelurahan Pulosari, Kecamatan Brebes, Kabupaten Brebes, 52213
3	M. Nishom	Jalan Jepara Perum Griya Putri Land Blok A6, Kecamatan Margadana, Kota Tegal, 52147

Lampiran 7. Lembar Bimbingan

SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : **Mohammad Zaidan Zufar**
NIM : **19090027**
No. Ponsel : **085156059627**
Judul TA : **Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naïve Bayes**
Dosen Pembimbing I: **Dega Surono Wibowo, S.T., M.Kom.**

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1.	13/3 2023	* validasi data	dari kemenpo tegal * cek data nya sudah berklasifikasi / terkategorikan atau tidak.	
2	27/3 2023	* Data sudah lengkap tinggal validasi belum maksimal.		
3.	4/5 2023.	* develop mock up * develop aplikasi * sambil memperbaiki bug.		
4.	17/5 2023	HKI - segera membuat manual book - teknikal		
5.	15/6 2023	Bikin jutkan lagi.		

6	13/7 2023	• Sistematika penulisan. • Process = Activity = Sequence • Singkatan (kepanjangan disebut dulu)		✓
7.	14/7 2023	• abstrak single spasi • mention gambar		✓
8	14/7 2023 14:29	Laporan Selesai		✓

Silahkan.
Daftar Sidang

— done —

Tegal, Juli 2023
Dosen Pembimbing I

Dega Surono Wibowo, S.T., M.Kom.
NIPY. 06.014.183

SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : **Mohammad Zaidan Zufar**
NIM : **19090027**
No. Ponsel : **085156059627**
Judul TA : **Aplikasi Pendeteksi Situs Phising Berbasis Website Menggunakan Metode Naïve Bayes**
Dosen Pembimbing II : M. Nishom, M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1	10/3 - 2023	- Domain list	→ Lakukan / lanjutkan pengumpulan data ke "pandi".	
2.	30/3 - 2023	- domain list - Notar	- berikan tambahan catatan pada setiap domain (ex: database).	
3.	4/05 - 2023	- Result of Model.	- Lakukan testing dengan dataset yang sama menggunakan metode Naive bayes dan logistic regression untuk mendapatkan tingkat validitas hasil deteksi.	
4.	12/05 - 2023	- Metode	- Gunakan logistic. (pelajari cara kerja logistic).	
		- Pengujian Model.	- Uji model dengan daftar URL baru. - Lakukan validasi	

5	8/6 - 2023	- Model → Metode Uji akurasi metode dengan membandingkan metode lain (Logistic). → Tinjau perbedaan Naive Bayes & Linear dari sisi kelebihan. (dari jurnal)		
		→ Model	- Lakukan komparasi semua model klasifikasi yang biasa digunakan untuk kasus web phising. 1. SMO ✓ 2. Random forest ✓ 3. Naive Bayes ✓ 4. Logistic ✓	
6	14/6 - 2023	- Model.	Gunakan Naive bayes sebagai model yang akan di-implementasikan ke dalam projek.	

\

7	15/6-23	- Luaran Penelitian	- Persiapkan dokumen persyaratan pengajuan HKI	
8	14/7-2023	- Laporan Penelitian.	- perbaiki desain antarmuka.	
9	17/7-2023	- Pendaftaran Sidang	- lengkapi dokumen persyaratan pendaftaran Sidang Skripsi	

Tegal, Juni 2023
Dosen Pembimbing II

M. Nishom, M.Kom.
NIPY. 09.017.337