

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Perkembangan internet dan teknologi digital saat ini telah mengubah cara manusia berkomunikasi dan beraktivitas sehari-hari. Hanya dalam satu genggam, masyarakat kini bisa mengakses berbagai informasi dan layanan secara cepat selama 24 jam penuh [1]. Namun, kemudahan ini ibarat pedang bermata dua; di satu sisi sangat membantu, tetapi di sisi lain juga membuka celah bagi para pelaku kejahatan siber (*cybercrime*) untuk melancarkan aksinya [2]. Memasuki tahun 2026, taktik kejahatan siber terbukti menjadi semakin cerdas dan canggih. Para pelaku tidak lagi hanya menyerang sistem komputer yang lemah, melainkan memanfaatkan layanan *cloud* resmi yang lazim dipercaya masyarakat guna menyembunyikan aktivitas kejahatan tersebut agar tidak terdeteksi oleh antivirus atau sistem keamanan biasa [3].

Dari berbagai jenis kejahatan siber yang ada, penipuan lewat tautan atau *phishing* menjadi ancaman yang paling sering memakan korban. *Phishing* sendiri adalah trik penipuan berbasis manipulasi psikologis (*social engineering*), di mana pelaku menyamar sebagai instansi resmi atau pihak tepercaya untuk memancing korban agar bersedia memberikan informasi sensitif, seperti *username*, *password*, PIN, hingga data perbankan [5]. Di Indonesia, masalah ini sudah sangat darurat. Badan Siber dan Sandi Negara (BSSN) mencatat ada ratusan juta serangan siber yang terjadi, dengan *phishing* sebagai jenis laporan yang paling mendominasi [4].

Di Indonesia, masalah ini sudah sangat darurat. Badan Siber dan Sandi Negara (BSSN) mencatat ada ratusan juta serangan siber yang terjadi, dengan *phishing* sebagai jenis laporan yang paling mendominasi [4]. Korban yang terjebak tidak hanya mengalami kerugian finansial yang besar, tetapi juga kehilangan kendali atas data pribadi tersebut yang berujung pada kebocoran data massal [6].

Meskipun pemerintah telah mengesahkan Undang-Undang Perlindungan Data Pribadi (UU PDP) untuk melindungi masyarakat, angka penipuan ini tetap tinggi karena pelaku selalu menemukan celah baru. Banyak korban secara tidak sadar menyerahkan data sensitif karena panik akibat pesan ancaman palsu atau tergiur hadiah gratis. Selain kelengahan manusia, para pelaku kini menggunakan teknik manipulasi alamat situs yang sangat rapi. Salah satunya adalah serangan *homograf*, yaitu trik membuat nama domain palsu menggunakan huruf dari alfabet asing (seperti huruf *Cyrillic* 'a' atau 'o') yang bentuk fisiknya sangat mirip dengan huruf Latin biasa. Akibatnya, mata manusia akan melihat alamat situs tersebut sebagai situs asli yang aman, padahal peramban (*browser*) sebenarnya sedang diarahkan ke halaman jebakan [7].

Celakanya, sistem pertahanan yang dipakai masyarakat saat ini masih bergantung pada metode daftar hitam (*Blacklist*) statis yang mencatat situs-situs berbahaya setelah dilaporkan. Cara lama ini terbukti kalah cepat karena situs *phishing* masa kini biasanya hanya aktif selama beberapa menit lalu menghilang dan berganti dengan nama baru [3]. Ketika benteng keamanan utama jebol, peramban (*browser*) di perangkat pengguna menjadi pertahanan terakhir yang

menentukan aman atau tidaknya sebuah akses . Oleh karena itu, dibutuhkan sebuah sistem pengaman langsung di dalam peramban (*client-side defense*) yang mampu memeriksa alamat situs secara otomatis dan cerdas saat itu juga baik dengan membongkar keaslian karakter hurufnya maupun menganalisis keanehan pola teks pada tautan tersebut sebelum halamannya sempat terbuka [7].

Berangkat dari masalah tersebut, penelitian ini mengembangkan sebuah ekosistem keamanan digital yang dinamakan *Secure Link Guardian*, yang diperkuat oleh ekstensi peramban bernama *XGBOOSTGuard*. Ekosistem ini bekerja lewat tiga pilar utama: *Backend API* sebagai otak pemroses data, *Web Dashboard* sebagai halaman kendali bagi admin, serta *Chrome Extension* sebagai garda terdepan yang langsung memblokir tautan berbahaya di peramban pengguna. Sistem ini mendeteksi ancaman secara ganda, yaitu melalui modul khusus yang bisa membongkar manipulasi huruf (*homograf*) dan model kecerdasan buatan (*Machine learning*) dengan algoritma *XGBOOST* yang dilatih untuk mengenali ciri-ciri tautan palsu secara instan.

Berdasarkan urutan masalah di atas, penelitian ini mengajukan sebuah hipotesis bahwa penerapan sistem pembongkar manipulasi huruf (*homograf*) yang digabungkan dengan kecerdasan buatan algoritma *XGBOOST* di dalam ekstensi peramban (*client-side*), mampu mendeteksi dan langsung memblokir tautan *phishing* baru secara akurat sebelum halaman tersebut sempat terbuka, sekaligus memudahkan pengelola untuk melatih ulang sistem keamanan secara mandiri tanpa perlu mengubah kode pemrograman.

## 1.2. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah diuraikan, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang dan mengintegrasikan arsitektur tiga pilar yang terdiri atas *Backend API* berbasis *Flask*, *Web Dashboard* berbasis *React*, dan *web Extension Manifest V3* menjadi satu kesatuan ekosistem keamanan siber yang mampu mendeteksi dan memblokir tautan *phishing* secara terintegrasi.
2. Mengimplementasikan modul *Homograph Detection* yang mampu melakukan proses *decode* karakter *Unicode* secara otomatis untuk mendeteksi dan mengungkap manipulasi alamat situs berbasis serangan *homograf (homograph attack)* secara *real-time* pada peramban.
3. Menerapkan model *Machine learning* menggunakan algoritma *Extreme Gradient Boosting (XGBOOST)* dengan memanfaatkan ekstraksi 11 fitur leksikal *URL* untuk memprediksi tingkat bahaya tautan *phishing* secara cepat dan akurat.
4. Merancang dan mengembangkan fitur manajemen model secara mandiri pada *Dashboard* administrator, meliputi proses *retraining* model *XGBOOST* serta pembaruan versi ekstensi, sehingga sistem pertahanan dapat diperbarui secara dinamis tanpa memerlukan perubahan pada kode program.

## 1.3. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat bagi berbagai pihak, baik bagi peneliti, mahasiswa, masyarakat, maupun institusi pendidikan, khususnya

dalam bidang keamanan siber dan penerapan teknologi *machine learning*. Adapun manfaat penelitian ini adalah sebagai berikut:

#### 1.3.1 Bagi Peneliti

Penelitian ini memberikan kesempatan bagi peneliti untuk memperdalam pemahaman mengenai penerapan algoritma *Extreme Gradient Boosting (XGBOOST)* dalam deteksi *URL phishing* serta proses integrasinya ke dalam *BROWSER EXTENSION*. Selain itu, penelitian ini juga menjadi sarana untuk mengasah kemampuan analisis, riset, dan pengembangan sistem berbasis *machine learning* yang relevan dengan perkembangan teknologi keamanan siber saat ini.

#### 1.3.2 Bagi Mahasiswa Lain

Penelitian ini dapat menjadi referensi dan bahan pembelajaran bagi mahasiswa lain yang ingin mengembangkan penelitian di bidang serupa, khususnya mengenai penerapan *machine learning* dalam keamanan data dan sistem. Hasil penelitian ini diharapkan dapat menginspirasi mahasiswa lain di Universitas Harkat Negeri untuk terus berinovasi dalam menghadapi tantangan dunia digital yang semakin kompleks.

#### 1.3.3 Bagi Masyarakat atau Pengguna Internet

Secara praktis, hasil penelitian ini dapat memberikan manfaat langsung bagi masyarakat luas dalam bentuk sistem *SMART PHISHING URL DETECTION* yang dapat membantu mendeteksi situs berbahaya secara *real-time*. Dengan demikian, pengguna internet dapat lebih terlindungi dari

ancaman pencurian data, penipuan *online*, serta serangan siber yang menggunakan teknik *phishing*.

#### 1.3.4 Bagi Institusi (Universitas Harkat Negeri)

Penelitian ini diharapkan dapat memberikan kontribusi positif bagi Universitas Harkat Negeri dalam pengembangan riset dan inovasi di bidang keamanan siber dan kecerdasan buatan. Selain memperkuat reputasi akademik universitas, hasil penelitian ini juga dapat digunakan sebagai bahan ajar, referensi penelitian lanjutan, dan contoh penerapan teknologi yang bermanfaat bagi masyarakat luas.

#### 1.4. Tinjauan Pustaka

Perkembangan teknologi informasi dan komunikasi internet telah membawa perubahan besar pada tatanan sosial, ekonomi, dan budaya masyarakat. Di satu sisi, internet memberikan efisiensi akses informasi dan mengubah model bisnis, namun di sisi lain internet memicu peningkatan kejahatan dunia maya (*cybercrime*) [8]. *Cybercrime* sendiri mencakup aktivitas ilegal seperti *carding*, *hacking*, *cracking*, penipuan, hingga penyebaran informasi berbahaya yang mengeksploitasi kerentanan sistem demi keuntungan finansial pelaku. Salah satu bentuk kejahatan siber yang paling sering terjadi dan menimbulkan kerugian materiil maupun non-materiil berskala besar adalah *phishing* [9]. *Phishing* merupakan serangan rekayasa sosial (*social engineering*) di mana penyerang memanipulasi psikologis korban dengan memanfaatkan rasa penasaran, urgensi, atau keserakahan agar mereka mengklik tautan berbahaya (*malicious URL*). Tautan ini dirancang untuk

mengarahkan korban ke situs web palsu yang dibuat semirip mungkin dengan situs resmi guna mencuri data sensitif seperti *username*, *password*, nomor kartu kredit, atau kredensial akun keuangan lainnya [10]. Serangan *phishing* saat ini menyebar luas melalui berbagai media komunikasi digital, termasuk email, SMS, pesan instan, dan *Platform* media sosial. Di Indonesia, salah satu tren ancaman siber spesifik menasar *Platform* dompet digital populer melalui manipulasi fitur "DANA Kaget". Popularitas fitur ini dimanfaatkan oleh penjahat siber untuk menyebarkan tautan *phishing* duplikat di media sosial demi memikat korban agar memasukkan nomor telepon serta kode OTP, yang berujung pada pengambilalihan akun dompet digital mereka [11].

Dalam perkembangannya, teknik pertahanan siber untuk mengidentifikasi situs web *phishing* telah mengalami beberapa fase perbaikan. Metode tradisional yang paling awal dan umum digunakan dalam sistem waktu nyata (*real-time*) adalah pendekatan berbasis daftar (*list-based approach*), yang mencakup daftar hitam (*Blacklist*) atau daftar putih (*whitelist*). Layanan populer seperti Google *SafeBrowsing*, *PhishTank*, dan *OpenPhish* mengumpulkan serta memverifikasi *URL* berbahaya untuk dimasukkan ke dalam daftar hitam agar ketika pengguna meminta akses ke sebuah *URL*, sistem dapat langsung mencocokkannya dengan basis data [9]. Meskipun metode ini memiliki keunggulan berupa biaya komputasi yang sangat rendah dan tingkat kesalahan deteksi (*false positive*) yang hampir nol, pendekatan berbasis daftar memiliki kelemahan kritis berupa ketidakmampuannya mendeteksi tautan *phishing* yang baru dibuat atau *zero-day phishing*. Mengingat siklus hidup rata-rata situs *phishing* sangat pendek dan penyerang terus-menerus

memodifikasi domain mereka, metode berbasis daftar menuntut pembaruan basis data secara terus-menerus dan sering kali gagal melindungi pengguna pertama yang mengunjungi tautan baru tersebut [10].

Untuk mengatasi keterbatasan metode berbasis daftar tersebut, para peneliti beralih ke solusi otomatis berbasis Kecerdasan Buatan, khususnya Pembelajaran Mesin (*Machine learning*) dan Pembelajaran Mendalam (*Deep Learning*). Algoritma pembelajaran mesin mampu mempelajari pola historis dari karakteristik *URL* untuk mengklasifikasikan secara otomatis apakah suatu tautan baru bersifat aman atau berbahaya [12]. Berdasarkan jenis data masukan yang digunakan, model deteksi berbasis pembelajaran mesin terbagi menjadi beberapa kategori fitur utama . Pertama adalah fitur leksikal yang diekstrak langsung dari *string URL* seperti panjang *URL*, jumlah titik, simbol khusus, dan tingkat keacakan domain, di mana pendekatan ini sangat disukai karena proses komputasinya cepat, aman, dan tidak mengharuskan sistem mengunduh konten situs web berbahaya yang berisiko. Kedua adalah fitur berbasis konten yang diekstrak dari kode sumber HTML halaman web, teks, serta gambar menggunakan teknik tertentu untuk mendeteksi kemiripan visual dengan situs asli. Ketiga adalah fitur eksternal yang diperoleh dari layanan pihak ketiga seperti data pendaftaran WHOIS dan peringkat popularitas halaman [9].

Berbagai algoritma klasifikasi terbukti efektif diterapkan dalam tugas klasifikasi biner ini, di antaranya *Support Vector Machine (SVM)*, *Random Forest (RF)*, *Decision Tree (DT)*, *Logistic Regression (LogR)*, *Naïve Bayes (NB)*, *Gradient Boosting (GB)*, dan *Extreme Gradient Boosting (XGBOOST)* . Sebagai contoh,

pengujian metode SVM menunjukkan kekuatan yang sangat baik dalam memisahkan data kompleks dan mencari bidang pemisah (*hyperplane*) optimal untuk mengklasifikasikan *Link* "DANA Kaget" palsu dengan tingkat akurasi pengujian mencapai 88% melalui skema *10-fold cross-validation*. Di samping itu, optimasi *hiperparameter* pada model seperti *XGBOOST* dan *Random Forest* terbukti dapat memaksimalkan kinerja deteksi pada dataset berskala besar [12].

Namun, salah satu tantangan terbesar dalam melatih model klasifikasi *malicious URL* adalah ketidakseimbangan kelas (*class imbalance*) pada dataset dunia nyata, di mana jumlah tautan normal biasanya jauh lebih melimpah dibandingkan dengan sampel tautan berbahaya. Ketidakseimbangan ini dapat menyebabkan model menjadi bias terhadap kelas mayoritas, sehingga menurunkan kemampuan mendeteksi serangan yang sebenarnya. Untuk mengatasi masalah tersebut, teknik pengambilan sampel ulang (*resampling*) seperti *Adaptive Synthetic Sampling* (ADASYN) dan *Synthetic Minority Over-sampling Technique* (SMOTE) diintegrasikan ke dalam tahap pra-pemrosesan data. Implementasi metode ADASYN yang secara adaptif menciptakan sampel sintetis baru pada kelas minoritas berdasarkan tingkat kesulitan pembelajaran sampel, yang kemudian dipadukan dengan algoritma berbasis aturan asosiasi seperti *Classification Based on Association* (CBA), mampu meningkatkan akurasi model secara signifikan dalam mengenali *URL* berbahaya dengan tingkat *false positive* yang sangat rendah [13]. Begitu pula dengan penggunaan SMOTE yang dipadukan dengan optimasi *hiperparameter* terbukti memberikan peningkatan stabilitas performa pada model klasifikasi sehingga sensitivitas terhadap kelas minoritas tetap terjaga.

Meskipun banyak model klasifikasi mencatatkan akurasi teoritis yang sangat tinggi, penelitian terbaru mengidentifikasi adanya celah penelitian terkait representasi data dalam skenario kasus nyata [10]. Mayoritas dataset publik ditemukan menggunakan halaman beranda (*homepage*) dari situs terkenal sebagai representasi kelas tepercaya. Padahal, dalam serangan nyata, penjahat siber meniru halaman formulir *login* spesifik. Ketika model yang dilatih menggunakan dataset *homepage* diuji dengan *URL* halaman *login* resmi yang karakternya cenderung lebih panjang dan rumit, tingkat pesan alarm palsu (*false-positive rate*) melonjak drastis karena model kebingungan membedakan struktur *URL login* resmi dengan situs *phishing*. Selain itu, terdapat tantangan stabilitas model terhadap faktor waktu karena teknik penipuan siber terus berevolusi. Penyerang secara konstan menyesuaikan struktur tautan mereka untuk menghindari aturan klasifikasi yang sudah dipublikasikan. Pengujian empiris membuktikan bahwa model yang dilatih menggunakan data *URL* lama mengalami penurunan akurasi yang signifikan saat diuji untuk memprediksi *URL* serangan baru dari tahun-tahun berikutnya. Oleh karena itu, literatur menegaskan pentingnya penggunaan dataset yang diperbarui secara berkala, ekstraksi fitur yang adaptif, serta pembaruan model secara kontinu agar sistem deteksi tetap andal dalam memitigasi serangan di lingkungan jaringan yang dinamis [9], [10].

Tabel 1. 1. Gap Penelitian

| No | Peneliti sebelumnya                                                                                              |                                                                                                                                                                                                                                                                                            | Kebaruan                                                                                                                                                                                                |
|----|------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Penulis                                                                                                          | Aspek yang dikutip                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                         |
| 1  | I Kadek Odie Kharisma Putra, dkk. (2022). Tindakan Kejahatan pada Dunia Digital dalam Bentuk <i>Phishing</i> [8] | Jurnal ini berhasil memetakan aspek psikologis social engineering dan dampak hukum akibat kejahatan siber. Namun, pendekatannya masih bersifat teoretis dan deskriptif kualitatif, sehingga tidak menawarkan solusi teknis atau sistem otomatis untuk memitigasi serangan secara langsung. | Menyediakan ekosistem proteksi aktif berbasis arsitektur 3-tier ( <i>Backend</i> Flask, <i>Dashboard</i> React, dan Ekstensi Browser) untuk tindakan preventif secara langsung pada perangkat pengguna. |

| No | Peneliti sebelumnya                                                                                                           |                                                                                                                                                                                                                                                                                         | Kebaruan                                                                                                                                                         |
|----|-------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Penulis                                                                                                                       | Aspek yang dikutip                                                                                                                                                                                                                                                                      |                                                                                                                                                                  |
| 2  | Lizhen Tang & Qusay H. Mahmoud (2021), <i>A Survey of Machine learning-Based Solutions for Phishing Website Detection</i> [9] | Studi literatur ini berhasil merangkum berbagai ekstraksi fitur (leksikal dan konten) yang efektif untuk deteksi otomatis. Kelemahannya, riset ini hanya berupa survei makro tanpa adanya implementasi sistem terintegrasi yang siap digunakan oleh pengguna akhir ( <i>end-user</i> ). | Menerapkan ekstraksi 11 fitur leksikal <i>URL</i> yang di <i>Validasi</i> langsung secara <i>real-time</i> oleh model cerdas berbasis algoritma <i>XGBOOST</i> . |

| No | Peneliti sebelumnya                                                                                               |                                                                                                                                                                                                                                                                                                                                                                                                        | Kebaruan                                                                                                                                                                                               |
|----|-------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Penulis                                                                                                           | Aspek yang dikutip                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                        |
| 3  | Manuel Sánchez-Paniagua, dkk. (2022), <i>Phishing URL Detection: A Real-Case Scenario Through Login URLs</i> [10] | Jurnal ini membuktikan bahwa deteksi pada level <i>URL</i> halaman <i>login</i> jauh lebih merepresentasikan kasus nyata dibanding halaman beranda ( <i>homepage</i> ). Namun, kelemahannya adalah model klasifikasi bersifat kaku/statis setelah dilatih, sehingga mengalami penurunan performa signifikan terhadap faktor waktu ( <i>time robustness</i> ) karena pola serangan siber terus berubah. | Menyediakan fitur <i>Dynamic Retraining</i> Model langsung dari Web <i>Dashboard</i> oleh Administrator untuk melatih ulang dan memperbarui versi model ML secara dinamis tanpa mengubah kode program. |

| No | Peneliti sebelumnya                                                                                                                                                         |                                                                                                                                                                                                                                                                                                   | Kebaruan                                                                                                                                                                                                                    |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Penulis                                                                                                                                                                     | Aspek yang dikutip                                                                                                                                                                                                                                                                                |                                                                                                                                                                                                                             |
| 4  | Mutiara Vebriani & Wiyli Yustanti (2023), Klasifikasi Deteksi <i>Link Phishing</i> DANA Kaget Menggunakan Metode <i>Support Vector Machine</i> Berbasis <i>Website</i> [11] | Penelitian ini sukses mendeteksi jenis ancaman <i>phishing</i> lokal spesifik di Indonesia, yaitu manipulasi tautan DANA Kaget. Batasannya, sistem ini hanya fokus pada pemisahan <i>URL</i> standar dan tidak mampu mengenali manipulasi teks tingkat lanjut, seperti serangan <i>homograf</i> . | Mengintegrasikan Analisis <i>Homograf</i> lewat modul <i>Backend CyrillicController.py</i> yang mendeteksi, mendecode Punycode, dan memetakan karakter <i>Cyrillic</i> kembali ke Latin untuk mengungkap manipulasi domain. |

| No | Peneliti sebelumnya                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                                                   | Kebaruan                                                                                                                                                                            |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Penulis                                                                                                                                              | Aspek yang dikutip                                                                                                                                                                                                                                                                                                                                                |                                                                                                                                                                                     |
| 5  | Saleem Raja Abdul Samad, dkk. (2023), <i>Analysis of the Performance Impact of Fine-Tuned Machine learning Model for Phishing URL Detection</i> [12] | Menemukan bahwa optimasi hiperparameter ( <i>fine-tuning</i> ) pada <i>XGBOOST</i> dan Random Forest menghasilkan akurasi yang optimal pada dataset besar. Celahnya, penelitian ini sebatas eksperimen performa kode di lingkungan lokal ( <i>notebook/code execution</i> ) dan belum diimplementasikan ke dalam arsitektur produk perangkat lunak terdistribusi. | Mengimplementasikan algoritma <i>XGBOOST</i> hasil optimasi ke dalam <i>production-ready web service</i> menggunakan <i>API Flask</i> yang terhubung ke basis data <i>MongoDB</i> . |

| No | Peneliti sebelumnya                                                                                                                      |                                                                                                                                                                                                                                                                                                         | Kebaruan                                                                                                                                                                                                                        |
|----|------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Penulis                                                                                                                                  | Aspek yang dikutip                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                                                                 |
| 6  | Gilang Dwi Setyawan, dkk. (2023), Implementasi Metode Adasyn dalam Deteksi <i>URL</i> Berbahaya Menggunakan <i>Machine learning</i> [13] | Membuktikan bahwa teknik pengambilan sampel ulang (resampling) dengan ADASYN sangat efektif dalam menangani ketidakseimbangan kelas (class imbalance) data <i>URL</i> . Sayangnya, sistem klasifikasi yang diusulkan hanya berjalan pasif sebagai alat pelapor, bukan sebagai pemblokir navigasi aktif. | Menerapkan Proteksi Garda Terdepan via <i>BROWSER EXTENSION</i> Chrome berbasis <i>Manifest V3</i> yang memantau navigasi <i>URL</i> secara <i>real-time</i> dan memotong akses jaringan dengan halaman pemblokiran interaktif. |

### 1.5. Data Penelitian

Dalam pelaksanaan penelitian ini, ketersediaan dan kualitas data merupakan aspek yang krusial untuk melatih serta menguji performa model deteksi yang dibangun. Data penelitian yang digunakan berupa data sekunder yang terdiri dari kumpulan alamat situs web atau *Uniform Resource Locator (URL)*. Data ini mencakup dua kategori utama, yaitu kelompok *URL* aman (legitimate) dan

kelompok *URL* berbahaya (*phishing*). Proses pengumpulan, klasifikasi, dan evaluasi data ini memerlukan sejumlah bahan serta alat bantu yang spesifik. Oleh karena itu, bagian ini akan menjabarkan secara rinci mengenai bahan-bahan data yang diproses serta instrumen pendukung (alat) yang digunakan selama penelitian berlangsung.

### 1.5.1 Bahan Penelitian

Bahan penelitian merujuk pada objek, material, atau data mentah yang diolah dan dianalisis selama proses penelitian berlangsung. Dalam konteks penelitian berbasis kecerdasan buatan dan pengembangan sistem deteksi *phishing* ini, bahan utama yang digunakan meliputi data *URL* mentah yang diperoleh dari Platform penyedia data terbuka (open dataset), fitur-fitur struktural hasil ekstraksi *string URL*, serta berkas biner model klasifikasi yang telah dilatih menggunakan *framework machine learning*. Rincian mengenai bahan penelitian tersebut beserta fungsi dan Platform sumbernya disajikan secara sistematis pada Tabel 1.2 berikut:

Tabel 1. 2. Rincian Bahan Penelitian beserta Sumber Platform

| No | Nama Bahan                       | Deskripsi /<br>Karakteristik                                                       | Fungsi /<br>Peranan                                                   | Sumber / Platform<br>Asal                             |
|----|----------------------------------|------------------------------------------------------------------------------------|-----------------------------------------------------------------------|-------------------------------------------------------|
| 1  | Dataset <i>URL</i><br><br>Mentah | Kumpulan 549.34<br>6 <i>string</i> tautan<br>( <i>URL</i> ) mentah<br>yang terbagi | Sebagai data<br>latih (training<br>set) dan data<br>uji (testing set) | Kaggle ( <i>Phishing</i><br>Site <i>URLs</i> Dataset) |

| No | Nama Bahan                      | Deskripsi /<br>Karakteristik                                                                                                                                                                                            | Fungsi /<br>Peranan                                                                                                                                  | Sumber / Platform<br>Asal                                             |
|----|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
|    |                                 | menjadi kelas <i>phishing</i> (positif/ <i>bad</i> ) dan legitimate (negatif/ <i>good</i> ).                                                                                                                            | dasar untuk melatih akurasi model <i>machine learning</i> .                                                                                          |                                                                       |
| 2  | Fitur<br>Leksikal<br><i>URL</i> | 11 atribut struktural numerik yang diekstraksi dari <i>string URL</i> (panjang <i>URL</i> , jumlah titik ., tanda hubung -, @, ?, =, /, <i>string</i> http, status protokol https, serta panjang domain dan subdomain). | Sebagai parameter parameter masukan (input features) bagi model klasifikasi untuk menilai tingkat keanehan pola <i>URL</i> secara <i>real-time</i> . | Python <i>String</i> Processing (ekstraksi panjang & karakter khusus) |

| No | Nama Bahan                       | Deskripsi /<br>Karakteristik                                                                                                                                                                               | Fungsi /<br>Peranan                                                                                                                                                                        | Sumber / Platform<br>Asal                                                   |
|----|----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| 3  | Model<br>Klasifikasi<br>Terlatih | Berkas biner<br>(serialized model)<br>hasil pelatihan<br>algoritma yang<br>disimpan dalam<br>format .joblib (nlp<br>_XGBOOST_phis<br>hing_pipeline.jobl<br>ib dan hybrid_XG<br>BOOST_phishing.<br>joblib). | Sebagai mesin<br>kecerdasan<br>buatan utama<br>di sisi server<br>(Backend API)<br>untuk<br>memproses<br>prediksi dan<br>klasifikasi<br>URL secara<br>cepat dalam<br>hitungan<br>milidetik. | <i>XGBOOST<br/>Framework (Extrem<br/>e Gradient Boosting<br/>Algorithm)</i> |

### 1.5.2 Alat Penelitian

Alat penelitian merupakan instrumen penunjang yang digunakan untuk memproses bahan penelitian, membangun arsitektur sistem, serta melakukan pengujian fungsionalitas program. Alat yang digunakan dalam penelitian ini dikelompokkan menjadi dua bagian utama, yaitu perangkat keras (*hardware*) untuk menjalankan proses komputasi fisik, dan perangkat lunak (*Software*) sebagai

lingkungan pengembangan kode program, basis data, dan antarmuka pengguna.

Rincian alat penelitian yang digunakan dalam pengerjaan skripsi ini disajikan pada

Tabel 1.3 di bawah ini:

Tabel 1. 3. Alat Dan Bahan

| No                                  | Nama Alat                    | Jenis             | Spesifikasi / Versi                       | Fungsi / Keterangan                                                                                      |
|-------------------------------------|------------------------------|-------------------|-------------------------------------------|----------------------------------------------------------------------------------------------------------|
| Perangkat Keras ( <i>Hardware</i> ) |                              |                   |                                           |                                                                                                          |
| 1                                   | Laptop/PC<br>Pengembang      | Fisik             | AMD Ryzen 3,<br>RAM 8GB, SSD<br>512GB     | Lingkungan fisik<br>utama untuk<br>menulis kode,<br>melatih model, dan<br>menjalankan server<br>lokal.   |
| 2                                   | Perangkat<br>Komputer<br>Uji | Fisik             | Laptop/PC<br>bersistem operasi<br>Windows | Media pengujian<br>ekstensi peramban<br>( <i>web browser</i> ) di<br>sisi pengguna<br>( <i>client</i> ). |
| Perangkat Lunak ( <i>Software</i> ) |                              |                   |                                           |                                                                                                          |
| 3                                   | Windows 10<br>/ 11           | Sistem<br>Operasi | 64-bit                                    | <i>Platform</i> dasar<br>untuk menjalankan<br>seluruh perangkat                                          |

| No | Nama Alat                            | Jenis                              | Spesifikasi / Versi           | Fungsi / Keterangan                                                                                        |
|----|--------------------------------------|------------------------------------|-------------------------------|------------------------------------------------------------------------------------------------------------|
|    |                                      |                                    |                               | lunak<br>pengembangan.                                                                                     |
| 4  | Python                               | Bahasa<br>Pemrograman              | Versi 3.x                     | Bahasa utama untuk<br>pengembangan<br>model <i>machine learning</i> dan<br>RESTful API<br><i>Backend</i> . |
| 5  | JavaScript /<br>TypeScript           | Bahasa<br>Pemrograman              | ES6+ / TypeScript             | Bahasa utama untuk<br>membangun<br>antarmuka web<br>ekstensi<br>dan <i>Landing Page</i> .                  |
| 6  | Flask                                | Library<br><i>Backend</i>          | Versi 3.1.0                   | Micro-<br>framework untuk<br>membangun<br>endpoint API<br>komunikasi data.                                 |
| 7  | <i>XGBOOST</i><br>& Scikit-<br>Learn | Library<br><i>Machine learning</i> | Versi Terbaru /<br>Kompatibel | Untuk pelatihan<br>algoritma<br><i>XGBOOST</i> dan                                                         |

| No | Nama Alat                      | Jenis                        | Spesifikasi / Versi                         | Fungsi / Keterangan                                                                                                  |
|----|--------------------------------|------------------------------|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
|    |                                |                              |                                             | pemrosesan teks<br>(pipeline TF-IDF).                                                                                |
| 8  | <i>MongoDB</i>                 | Basis Data                   | NoSQL<br>( <i>MongoDB</i> Atlas /<br>Lokal) | Menyimpan data<br>riwayat scanning,<br>manajemen akun<br>pengguna, dan data<br>pendukung lainnya.                    |
| 9  | Vite &<br>React.js             | Framework<br><i>Frontend</i> | Versi Terbaru                               | Mengembangkan<br>antarmuka <i>Landing</i><br><i>Page</i> dan <i>Popup</i><br>ekstensi secara cepat<br>dan responsif. |
| 10 | Tailwind<br>CSS &<br>shadcn/ui | Library UI<br>Styling        | Versi Terbaru                               | Merancang tampilan<br>antarmuka agar<br>modern, rapi, dan<br>responsif.                                              |
| 11 | Visual<br>Studio Code          | Code Editor                  | Versi Terbaru                               | Lingkungan<br>penulisan ( <i>coding</i> )<br>dan penyuntingan<br>kode sumber<br>proyek.                              |

| No | Nama Alat                       | Jenis              | Spesifikasi / Versi | Fungsi / Keterangan                                                                        |
|----|---------------------------------|--------------------|---------------------|--------------------------------------------------------------------------------------------|
| 12 | Git &<br>GitHub                 | Version<br>Control | Versi Terbaru       | Melakukan<br>pelacakan<br>perubahan kode<br>(versioning) dan<br>kolaborasi<br>repositori.  |
| 13 | Postman /<br><i>Web browser</i> | Alat<br>Pengujian  | Versi Terbaru       | Melakukan uji coba<br>request API<br><i>Backend</i> dan uji<br>fungsionalitas<br>ekstensi. |