JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 12 No. 3, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v12i3.4858

RANCANG BANGUN SISTEM PENDETEKSI LINK *PHISHING* MENGGUNAKAN ALGORITMA *RANDOM FOREST* BERBASIS WEB

Ali Dongan Harahap[1], Didi Juardi[2], Agung Susilo Yuda Irawan[3]

[1,2,3]Universitas Singaperbangsa Karawang; Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur Karawang, Indonesia; (0267) 641177

Received: 11 Juli 2024
Accepted: 31 Juli 2024
Published: 7 Agustus 2024

Keywords:
Phishing, Random Forest, Web Application, SDLC, Python, Flask, MySQL

Corespondent Email:
alidonganharahap21@gmail.com

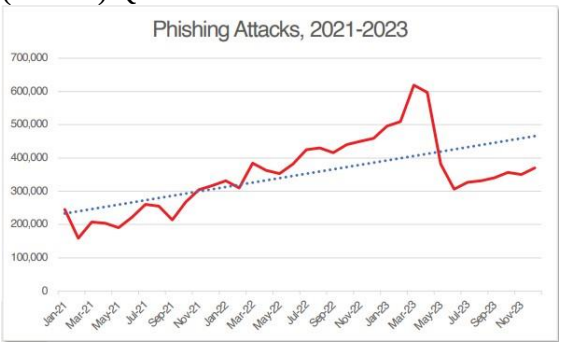
Abstrak. Phishing merupakan teknik penipuan yang memanfaatkan media internet untuk memperoleh informasi sensitif seperti kata sandi dan nomor kartu kredit dengan menyamar sebagai pihak yang tepercaya. Penelitian ini bertujuan untuk merancang dan membangun sistem pendeteksi link phishing berbasis web menggunakan algoritma Random Forest. Algoritma Random Forest dipilih karena kemampuannya dalam menangani jumlah data yang besar dan variabel yang banyak, serta mampu memberikan akurasi yang tinggi dalam proses klasifikasi. Sistem ini diimplementasikan dalam bentuk aplikasi web yang dapat digunakan oleh pengguna untuk memeriksa keaslian suatu link. Pengembangan sistem ini menggunakan metode SDLC (Software Development Life Cycle) dengan model Waterfall yang terdiri dari tahapan-tahapan: analisis kebutuhan, desain sistem, implementasi, pengujian, dan pemeliharaan. Bahasa pemrograman yang digunakan adalah Python dengan framework Flask untuk pengembangan backend, serta HTML, CSS, dan JavaScript untuk frontend. Basis data yang digunakan adalah MySQL. Hasil pengujian menunjukkan bahwa sistem yang dibangun mampu mendeteksi link phishing dengan tingkat akurasi yang memuaskan. Dengan demikian, diharapkan sistem ini dapat membantu pengguna dalam mengidentifikasi link phishing dan mengurangi risiko penipuan di dunia maya.

Abstract. *Phishing is a fraud technique that exploits the internet to obtain sensitive information such as passwords and credit card numbers by masquerading as a trustworthy entity. This study aims to design and develop a web-based phishing link detection system using the Random Forest algorithm. The Random Forest algorithm was chosen for its ability to handle large amounts of data and numerous variables, and its high accuracy in classification processes. This system is implemented as a web application that users can utilize to verify the authenticity of a link. The system development follows the Software Development Life Cycle (SDLC) method with the Waterfall model, consisting of stages: requirement analysis, system design, implementation, testing, and maintenance. The programming language used is Python with the Flask framework for backend development, and HTML, CSS, and JavaScript for frontend development. The database used is MySQL. The testing results indicate that the developed system is capable of detecting phishing links with satisfactory accuracy. Therefore, it is hoped that this system can assist users in identifying phishing links and reducing the risk of online fraud.*

1. PENDAHULUAN

Kemajuan berkembangnya penggunaan internet di Indonesia semakin meluas, perkembangan ini dimulai terutama pada saat era pandemic covid-19 yang membatasi aktivitas masyrakat secara langsung. Kebiasaan masyarakat yang sebelumnya selalu melakukan aktivitas secara fisik atau luring (luar jaringan) perlahan berubah melakukan aktivitas secara daring (dalam jaringan) melalui berbagai platform [1]. Karena aktivitas sekarang lebih sering dilakukan secara daring (dalam jaringan), internet menjadi akses sebagai media yang memudahkan dan menyediakan berbagai fasilitas yang mendukung aktivitas secara online seperti website. Banyak sekali website baru yang menjadi media untuk kolaborasi antar pengguna internet. Awalnya website hanya berfungsi sebagai alat penyampaian informasi satu arah. Namun, seiring dengan kemajuan teknologi informasi dan komunikasi, perubahan ini membawa website menjadi platform interaktif, media komunikasi, bahkan tempat untuk melakukan berbagai transaksi online. Meskipun perkembangan ini memberikan kemudahan, disayangkan bahwa kemunculan fitur baru pada website juga diimbangi oleh meningkatnya potensi ancaman siber, salah satunya adalah serangan web phishing [2].

Phising menjadi ancaman serius dalam dunia kejahatan siber, dimana tujuannya adalah mencuri informasi penting dari korban. Contohnya data login, seperti username dan password, serta detail pribadi termasuk rincian kartu kredit yang menjadi sasaran utama [1], [3]. Phising biasanya dilakukan dengan menyamar sebagai pihak tepercaya, seperti bank, artis, pemerintah, dan lainnya. Menurut sumber global dari Anti-Phising Working Grup (APWG) pada Q1 tahun 2023, APWG mengamati ada 1.624.144 serangan phishing. Jumlah ini merupakan jumlah tertinggi ketiga yang pernah dicatat oleh APWG. APWG melacak peningkatan yang kuat dalam penipuan berbasis telepon seluler. Volume voice-phishing, atau vishing, ini membengkak lebih dari 40% dibandingkan total pada kuartal keempat tahun 2022, dan mewakili peningkatan hampir sepuluh kali lipat dibandingkan dengan kuartal pertama tahun 2022. Bahkan setelah penurunan drastis pada kuartal kedua, phishing meningkat kembali di akhir tahun 2023, dan APWG mengamati 1.077.501 serangan phishing pada kuartal keempat tahun 2023. APWG menemukan bahwa peningkatan serangan phishing terjadi terhadap platform media sosial dengan mencakup 42,8% dari seluruh serangan phishing, meningkat dari 18,9% dari semua serangan di Q3. Berikut data yang diperoleh dari Anti-Phising Working Grup (APWG) Q4 tahun 2023.



Gambar 1 Data APWG

Banyaknya serangan phising yang terjadi dan peningkatan terjadi pada platform sosial media yang menjadi bagian dari aktivitas masyarakat sekarang, maka diperlukan suatu sistem yang mampu mendeteksi link phising. Tujuannya agar dapat mencegah masyarakat terkena serangan phising. Sebuah sistem deteksi yang efektif menjadi sebuah solusi yang dapat mengurangi resiko masyarakat terkena serangan phising.

Banyak penelitian yang melakukan pengembangan sistem yang dapat mendeteksi link phising, seperti pada bidang data mining ataupun machine learning. Namun masih sedikit yang mengembangkan algoritma machine learning menjadi sebuah sistem aplikasi deteksi phising berbasis website. Pada penelitian [4] dengan menggunakan algoritma Naïve Bayes, Decision Tree, dan Random Forest. Hasil dari penelitian ini melakukan testing data sebanyak 30% dan tingkat akurasi dari ketiga algoritma ini adalah 60,4% menggunakan naïve bayes, 94,4%

menggunakan decision tree, dan 96,3% dengan algoritma random forest. Penerapan algoritma random forest sangat efektif karena memiliki tingkat akurasi yang paling besar yaitu sebesar 96,3%.

Kemudian untuk mengembangkan sistem pendeteksi kedalam website menggunakan pendekatan Software Development Life Cycle (SDLC) yaitu metode waterfall. Menurut peneletian [5], penggunaan metode waterfall memiliki tahapan yang sistematis. Tahapan tersebut meliputi analisis, design, implementasi, standarisasi (integration), maintenance.

Berdasarkan peneletian sebelumnya maka penggunaan algoritma random forest sangat efektif untuk sebuah sistem pendeteksi dan metode waterfall sangat cocok untuk mengembangkan sistem ini kedalam website. Oleh karena itu, penulis mengangkat topik penelitian dengan judul "RANCANG BANGUN SISTEM PENDETEKSI LINK PHISING MENGGUNAKAN ALGORITMA RANDOM FOREST BERBASIS WEB" dengan harapan agar mempermudah masyarakat untuk mendeteksi link phising sebagai keamanan data mereka.

2. TINJAUAN PUSTAKA

2.1 Phishing

Salah satu jenis serangan siber yang dikenal sebagai phishing mencoba mendorong korban untuk mengakses situs web palsu yang mirip dengan situs aslinya. Salah satu metode yang biasa digunakan adalah dengan mengirimkan URL palsu melalui surel, tetapi seiring waktu, penyerang juga mulai mempertimbangkan penggunaan aplikasi pesan instan.

Beberapa tanda situs phishing yang mudah ditemukan termasuk URL yang terlalu panjang, penggunaan alamat IP dalam URL, dan fitur lain yang dapat ditemukan secara langsung. Namun, ada juga URL phishing yang sangat mirip dengan URL asli karena menggunakan karakter khusus. Komponen penting yang tidak dapat diubah oleh hacker, seperti sertifikat, usia domain, dan peringkat situs, harus diperiksa untuk memastikan bahwa situs tersebut asli. [6]

2.2 Deteksi

Deteksi, sebagai hal yang diinginkan manusia dalam situasi tertentu. Adanya alat deteksi memungkinkan manusia mendapatkan informasi yang dapat mengidentifikasi penyebab suatu kerusakan [7]. Proses deteksi ini dapat diartikan sebagai suatu metode pemeriksaan tertentu yang bermanfaat dalam membantu manusia menyelesaikan masalah yang dihadapinya.

2.3 *Website*

Website adalah kumpulan halaman informasi berisi teks, gambar, dan elemen lainnya, dibangun dengan bahasa pemrograman dasar seperti HTML, CSS, dan JavaScript [8]. Sebagai hasil perkembangan zaman dan teknologi komputer, website telah menjadi media efisien untuk menyampaikan informasi, mempromosikan produk atau layanan, serta menyediakan berbagai konten yang dapat diakses melalui browser web.

2.4 *Machine Learning*

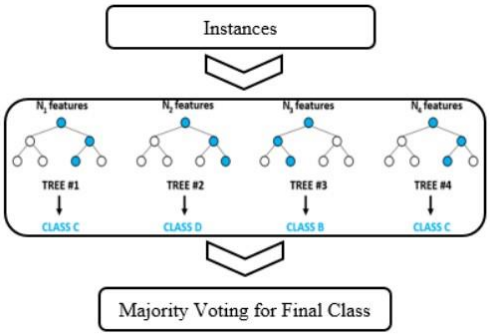
Suatu program komputer atau algoritma yang memiliki kemampuan untuk meningkatkan kecerdasan sistem melalui pemahaman pola dalam data saat ini dikenal sebagai machine learning (ML). Untuk menangani masalah tertentu, penggunaan pembelajaran mesin memerlukan tiga kondisi yaitu, memerlukan kapasitas memori yang signifikan, melibatkan proses pelabelan, dan seringkali menghasilkan prediksi yang tidak 100% akurat. [4]

2.5 Python

Python adalah bahasa pemrograman open-source yang dapat digunakan di banyak sistem operasi, termasuk Linux, Windows, dan MacOS. Bahasa pemrogaman ini juga fleksibel dan mudah dipelajari. Python biasanya memiliki modul standar yang memiliki banyak algoritma dan fungsi untuk menyelesaikan tugas seperti mengurai data teks dan mengunduh data dari web server. Pemograman Python biasanya lebih mudah dibaca dan lebih sederhana dibandingkan dengan bahasa C. Programer dapat dengan mudah menggunakan teknik komputasi tingkat lanjut, seperti pemrograman berorientasi objek, dengan menggunakan bahasa pemograman python. [9]

2.6 *Random Forest*

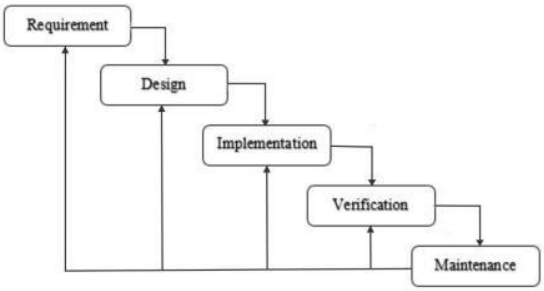
Random Forest adalah teknik dalam *machine learning* yang masuk dalam kategori *ensemble learning*, suatu pendekatan yang menggabungkan beberapa model pembelajaran mesin untuk meningkatkan performa dan keakuratan prediksi. *Random Forest* adalah algoritma yang menggabungkan beberapa pohon keputusan. Setiap pohon keputusan dilatih secara terpisah, dan model dasarnya diintegrasikan menggunakan skema pembobotan yang canggih. Umumnya, setiap pohon dilatih secara independen, dan prediksi dari setiap pohon digabungkan melalui rata-rata [10].



Gambar 2 Alur *Random Forest*

2.7 Metode SDLC (*Waterfall*)

SDLC adalah proses pengubahan dan pembuatan sistem, model, serta metodologi yang digunakan untuk mengembangkan software. Waterfall merupakan bagian dari SDLC (*Software Development Life Cycle*) yang memiliki karakteristik pada setiap fase pengerjaannya. Metode waterfall adalah pendekatan pengembangan sistem yang melibatkan serangkaian tahapan yang dilakukan secara berurutan dan sistematis. Proses ini dimulai dari tahap penjabaran kebutuhan sistem, diikuti dengan perancangan sistem, pengkodean, pengujian, dan akhirnya pemeliharaan. Setiap tahap harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya. [11]



Gambar 3 Alur *Waterfall*

2.8 MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal. MySQL menggunakan bahasa SQL untuk mengakses databasenya, sehingga memudahkan pengguna dalam melakukan operasi CRUD (Create, Read, Update, Delete) pada data. Lisensi MySQL adalah FOSS License Exception, yang berarti dapat digunakan secara gratis untuk proyek open-source. [12]

3. METODE PENELITIAN

Metode penelitian adalah rangkaian langkah-langkah yang penting untuk sebuah penelitian yang bertujuan untuk mengumpulkan dan mengevaluasi informasi dan data.

3.1 Metode Pengumpulan Data

Ada 2 metode pengumpulan data yaitu dengan observasi dan studi literatur.

3.1.1 Observasi

Pada tahap ini mengamati beberapa sistem deteksi phising pada penelitian sebelumnya dan website deteksi phising yang ada di internet untuk mencari informasi saat penelitian ini dilakukan, apakah memberikan nilai akurasi deteksi phising yang baik.

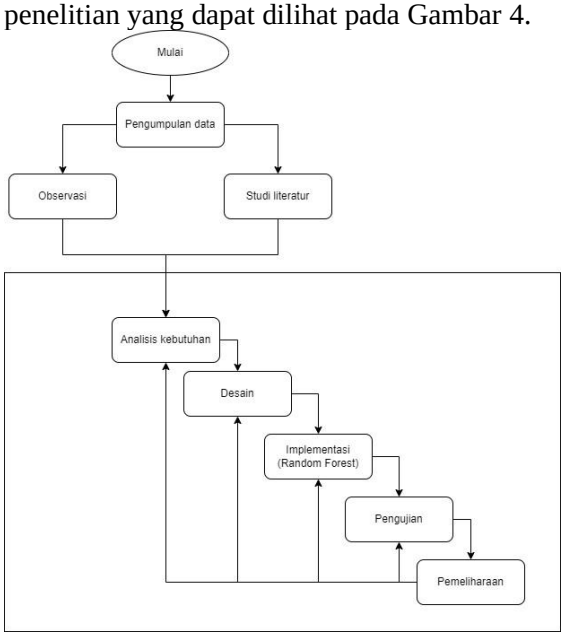
3.1.2 Studi Literatur

Pada tahapan ini akan mengumpulkan data atau informasi tentang rancang bangun sistem deteksi phising. Informasi atau data ini didapatkan dari berbagai sumber, seperti buku, penelitian terdahulu, jurnal, dan lainnya.

3.2 Metode Pengembangan *Software*

Metode yang digunakan untuk pengembangan sistem adalah metode waterfall. Metode ini akan memberikan struktur organisasi dengan langkah-langkah yang jelas mulai dari perencanaan, analisis, desain,

implementasi, dan pengujian. Berikut tahapan penelitian yang dapat dilihat pada Gambar 4.



Gambar 4 Metode *Waterfall*

Berikut adalah penjelasan dari tahapan-tahapan pada gambar 4:

3.2.1 Analisis Kebutuhan

Pada tahapan ini penulis melakukan analisis kebutuhan. Penulis akan menentukan kebutuhan apa saja yang akan digunakan untuk software yang akan dikembangkan, antara lain seperti kebutuhan fungsional dan nonfungsional.

3.2.2 Desain

Langkah berikutnya adalah merancang sistem berdasarkan informasi yang telah dikumpulkan setelah penulis menganalisis data sebelumnya. Pada tahap ini, penulis akan menjelaskan secara rinci struktur dan fungsi sistem yang akan dibuat, termasuk rancangan sistem, arsitektur perangkat lunak, dan user antarmuka, sesuai dengan kebutuhan yang telah diidentifikasi. Desain ini dibuat untuk memberikan gambaran tentang tindakan yang akan diambil.

3.2.3 Implementasi

Pada tahap implementasi, setelah melakukan perancangan tahap selanjutnya penulis akan mengimplementasikan konsep dan struktur yang sudah diatur dalam tahapa desain menjadi kode pemrograman yang dapat digunakan. Selama proses ini, aplikasi dibuat sesuai dengan spesifikasi yang telah ditentukan sebelumnya. Ini juga memastikan bahwa setiap komponen sistem beroperasi dengan benar sesuai dengan rancangan yang telah dibuat.

3.2.4 Pengujian

Pada tahapan ini penulis melakukan pengujian fungsionalitas kesuluruhan fitur hasil dari implementasi yang telah terintegrasi. Dalam proses ini, berbagai elemen fungsionalitas, keamanan, dan kinerja sistem dievaluasi secara menyeluruh. Dengan menjalankan pengujian, penulis dapat memastikan bahwa setiap fitur beroperasi sesuai dengan yang diharapkan.

3.2.5 Pemeliharaan

Dalam model waterfall, tahap terakhir adalah menjalankan perangkat lunak yang telah dibuat dan kegiatan pemeliharaan. Pada tahapan ini, perangkat lunak yang sudah berfungsi akan terus dipantau dan diperbarui jika diperlukan. Selain itu, kegiatan pemeliharaan akan sangat penting untuk memastikan bahwa perangkat lunak tetap berfungsi dan memenuhi kebutuhan yang berkembang seiring waktu. Proses ini memungkinkan peningkatan berkelanjutan dalam kinerja, fungsionalitas, dan keamanan perangkat lunak.

4 HASIL DAN PEMBAHASAN

4.1 Analisis

4.1.1 Analisis Kebutuhan Sistem

Pada tahapan analisis ini menentukan kebutuhan pengguna untuk software yang akan dikembangkan. Terdapat 2 analisis kebutuhan yaitu, kebutuhan fungsional mencakup fungsi sistem yang harus dilakukan, dan kebutuhan nonfungsional mencakup seperti kinerja, keamanan, dan kompatibilitas.

4.1.2 Analisis Kebutuhan Fungsional

Kebutuhan fungsional merupakan fitur-fitur yang disediakan oleh software yang bisa diakses oleh user secara langsung melalui interface yang disediakan oleh aplikasi atau sistem yaitu:

1. Halaman registrasi agar user bisa mendaftar. Halaman ini memungkinkan pengguna baru untuk membuat akun dengan memasukkan informasi yang diperlukan seperti nama, email, dan kata sandi.
2. Halaman login agar user bisa masuk pada aplikasi. Halaman ini memastikan bahwa hanya pengguna yang terdaftar yang bisa mengakses fitur dashboard.

3. Halaman pendeteksi phishing. Halaman ini menyediakan alat bagi pengguna untuk memasukkan URL yang mencurigakan, yang kemudian akan dianalisis oleh sistem untuk mendeteksi adanya indikasi phishing.
4. Halaman hasil deteksi phishing. Halaman ini menampilkan hasil analisis yang dilakukan oleh sistem, memberikan pengguna informasi apakah URL atau informasi yang diperiksa terindikasi sebagai phishing atau tidak.
5. Halaman dashboard. Halaman ini menampilkan URL apa saja yang sudah pernah dideteksi pada sistem ini dan juga hasil dari deteksinya.

4.1.3 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional mencakup software dan hardware yang digunakan dalam pengembangan aplikasi sebagai batasan fitur. Pada pengembangan aplikasi deteksi phishing, penulis menggunakan berbagai perangkat lunak sebagai berikut:

1. Visual Studio Code
2. Phyton, html, dan css
3. Mysql
4. Google Chrome
5. Xampp

Dan perangkat keras yang digunakan oleh penulis pada pengembangan aplikasi deteksi phising ini yaitu sebagai berikut:

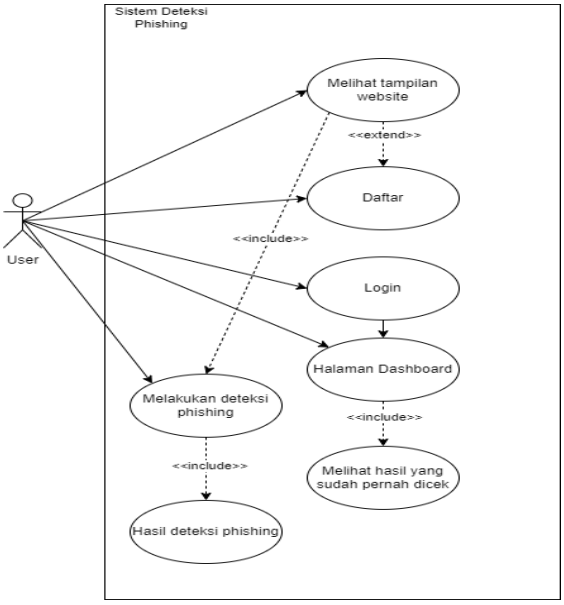
1. Laptop Aspire5
2. Ram 12GB
3. SSD 512.

4.2 Desain

Pada tahapan selanjutnya setelah melakukan analisis kebutuhan adalah desain. Pada tahap ini, kebutuhan yang dikumpulkan selama proses analisis dianalisis secara menyeluruh. Hasil analisis ini kemudian diterapkan dalam desain pengembangan, untuk memberikan gambaran yang jelas tentang apa yang akan dilakukan selama proses pengembangan aplikasi.

4.2.1 Use Case Diagram

Use case diagram adalah sebuah diagram yang dirancang untuk menggambarkan interaksi dari sistem yang sedang dikembangkan dengan pengguna.

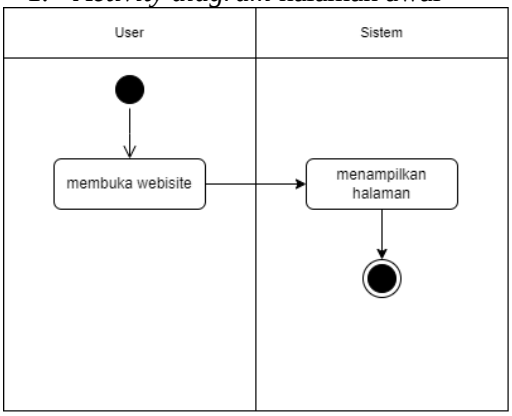


Gambar 5 *Use case Diagram*

4.2.2 Activity Diagram

Activity diagram dirancang seseuai use case yang telah dibuat sebelumnya untuk pengembangan aplikasi deteksi phishing berbasis website. Diagram ini menggambarkan langkah-langkah yang diambil oleh sistem dan pengguna.

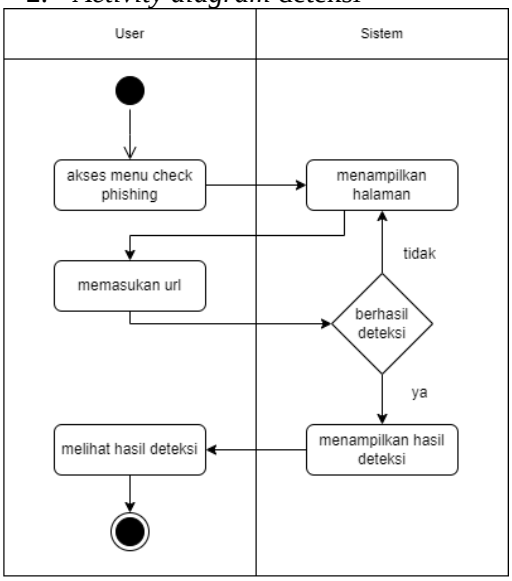
1. *Activity diagram* halaman awal



Gambar 6 Activity Halaman awal

Pada gambar 4.2 user memulai interaksinya dengan membuka website. Ini berarti pengguna memasukkan URL website ke dalam browser atau mengklik sebuah link yang mengarah ke website.

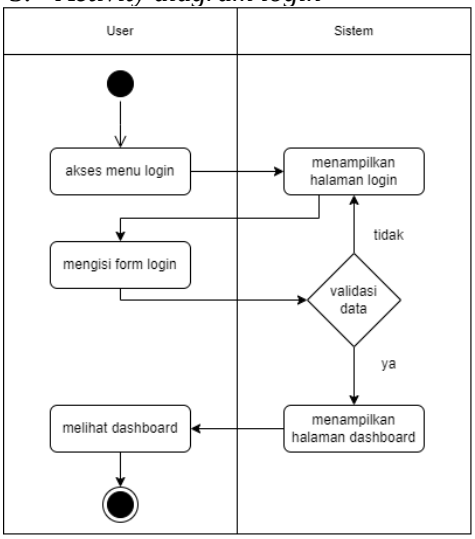
2. *Activity diagram* deteksi



Gambar 7 Activity Deteksi

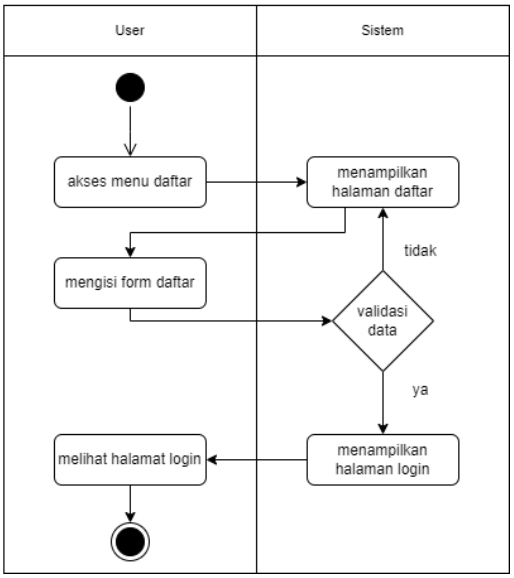
Pada gambar 7 user memulai dengan membuka website check phishing yang menyediakan layanan untuk mengecek apakah sebuah URL adalah phishing. Setelah itu, User memasukkan URL yang ingin diperiksa ke dalam sistem melalui form atau input field. Setelah sistem memproses URL yang dimasukkan, user akan melihat hasil deteksi yang ditampilkan oleh sistem, yang menunjukkan apakah URL tersebut termasuk phishing atau tidak.

3. *Activity diagram login*



Gambar 8 *Activity Login*

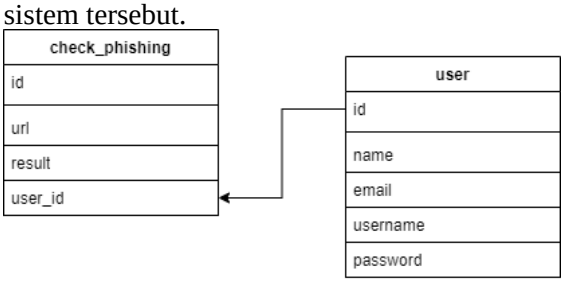
4. *Activity diagram* daftar



Gambar 9 Activity Daftar

4.2.3 Class Diagram

Class diagram merupakan gambaran dari hubungan antara objek-objek dan struktur yang ada dalam sebuah aplikasi. Diagram tersebut mencakup atribut-atribut dan metode-metode yang dimiliki oleh masing-masing kelas dalam sistem tersebut. Untuk aplikasi deteksi phishing berbasis website, Class Diagram memberikan gambaran yang jelas tentang bagaimana objek-objek seperti user dan check phishing berinteraksi satu sama lain dalam sistem tersebut.



Gambar 10 Class Diagram

4.3 Implementasi

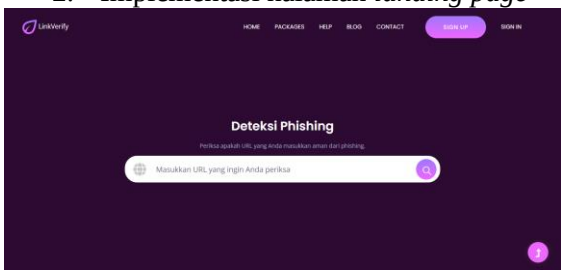
Pada tahap implementasi atau pengkodean mengubah desain menjadi sebuah website agar fungsi perangkat lunak dapat berjalan dengan baik. Untuk mewujudkan perubahan desain ini, implementasi (pengkodean) menggunakan Visual Studio Code sebagai lingkungan implementasi. Pada sisi front end, implementasi menerapkan pemrograman CSS dan HTML untuk membangun antarmuka pengguna yang

responsif dan menarik menggunakan framework flask. Sedangkan pada sisi back end, digunakan bahasa pemrograman Python dan algoritma random forest untuk menangani logika dan integrasi dengan basis data.

1. Implementasi algoritma *random forest*

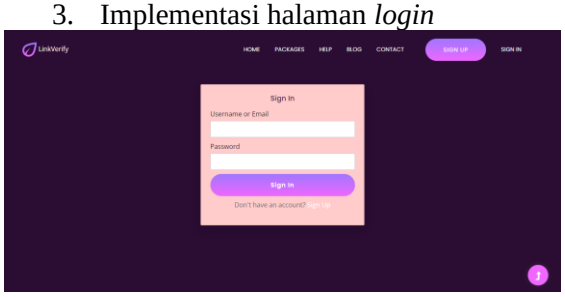
Gambar 11 *Random Forest*

2. Implementasi halaman *landing page*



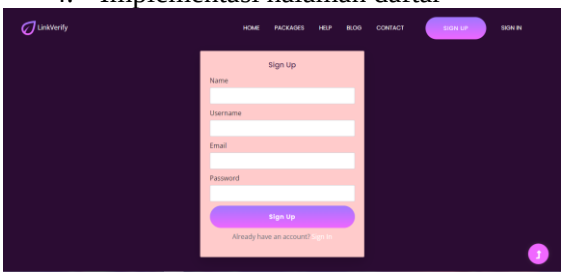
Gambar 12 Halaman Landing Page

3. Implementasi halaman *login*



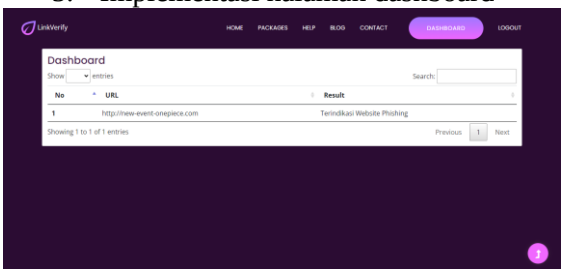
Gambar 13 Halaman Login

4. Implementasi halaman daftar



Gambar 14 Halaman Daftar

5. Implementasi halaman dashboard



Gambar 15 Halaman Dashboard

4.4 Pengujian

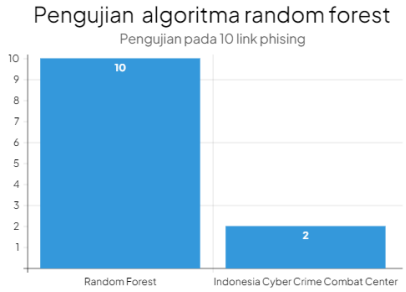
. Metode pengujian berkonsentrasi pada pengujian fungsionalitas fitur yang telah terintegrasi secara keseluruhan. Tujuan pengujian black box ini adalah untuk memastikan bahwa fungsi, masukan, dan keluaran perangkat lunak memenuhi spesifikasi, tanpa memperhatikan desain dan kode program [13]. Tujuan dari pengujian end-to-end adalah untuk menguji semua fitur yang diterapkan dalam sebuah iterasi dari sudut pandang pengguna secara end-to-end untuk memastikan bahwa aplikasi berfungsi sesuai harapan dalam situasi penggunaan nyata.

Tabel 1 Blackbox Testing

Feature	User Story	Test Case	Result
Search	Deteksi Phishing	Deteksi URL phishing	Behasil
Authentication	Daftar	Daftar dengan data valid	Behasil
	Login	Login dengan data valid	Berhasil

4.5 Pengujian algoritma random forest

Di bawah ini adalah perbandingan hasil pengujian dari 2 website deteksi phishing, menggunakan 10 URL phishing dan dilakukan sebanyak 5 kali percobaan. Pengujian ini bertujuan untuk mengevaluasi kinerja dan akurasi masing-masing website dalam mendeteksi URL phishing. Detail lengkap pengujian dapat dilihat pada tabel di bawah ini, yang menunjukkan performa setiap website berdasarkan percobaan yang dilakukan.



Gambar 16 Pengujian Random Forest

4.6 Pemeliharaan

Pada tahapan pemeliharaan ini, adalah untuk meningkatkan kinerja dan memastikan kelancaran operasional sistem yang sudah ada. Tahapan ini mencakup berbagai aktivitas menyeluruh, mulai dari analisis kebutuhan awal hingga pengujian, yang memungkinkan pengembangan dan peningkatan sistem tanpa harus memulai dari awal. Tujuan utama pemeliharaan adalah menangani dan menyelesaikan masalah. Diantaranya memperbaiki kesalahan, mengoptimalkan kinerja, dan memastikan sistem tetap aman. Selain itu, pemeliharaan adalah proses terus-menerus untuk meningkatkan sistem dan memperbaiki bug. Proses ini diuji secara menyeluruh untuk memastikan bahwa semua fitur yang ada berfungsi dengan baik tanpa adanya kesalahan ataupun error. Pemeliharaan ini akan berlangsung terus dilakukan agar memastikan sistem tetap berfungsi dengan baik.

5 KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dari tahapan analasis hingga pemeliharaan dapat disimpulkan bahwa:

1. Sistem deteksi phishing berbasis web yang dikembangkan berhasil memenuhi tujuan penelitian, yaitu mengembangkan sistem pendeteksi link phishing berbasis web.
2. Algoritma Random Forest berhasil diimplementasikan dalam sistem pendeteksi link phishing ini. Algoritma random forest terbukti efektif dalam fitur deteksi phishing pada sistem ini. Algoritma ini memberikan hasil yang akurat, sehingga meningkatkan keamanan pengguna dari ancaman phishing. Sistem ini juga berhasil mengatasi kekurangan dari sitem deteksi phishing berbasis web yang ada sebelumnya, terutama dalam hal keberhasilan deteksi URL. Hasil evaluasi menunjukkan bahwa sistem ini memiliki tingkat akurasi keberhasilan deteksi yang lebih baik dibandingkan dengan aplikasi sebelumnya.

Secara keseluruhan, sistem ini tidak hanya memenuhi tujuan penelitian, tetapi juga memberikan solusi yang lebih baik dan lebih aman bagi masyarakat dalam menghadapi ancaman phishing. Implementasi algoritma random forest dalam deteksi phishing menunjukkan potensi besar untuk pengembangan lebih lanjut di bidang keamanan siber.

Berikut adalah beberapa saran untuk pengembangan dan perbaikan lebih lanjut:

1. Peningkatan Algoritma: Meskipun algoritma random forest telah terbukti efektif, disarankan untuk mengeksplorasi dan mengimplementasikan algoritma machine learning lainnya.
2. Pembaruan Data: Untuk memastikan bahwa data yang digunakan untuk melatih model deteksi phishing selalu diperbarui secara berkala. Ancaman phishing berkembang dengan cepat, dan dataset yang up-to-date akan membantu sistem tetap efektif dalam mendeteksi ancaman terbaru.
3. User Experience: Terus tingkatkan antarmuka pengguna (UI) dan pengalaman pengguna (UX) agar lebih mudah digunakan. Pengguna lebih menerima teknologi baru jika antarmukanya sederhana dan mudah dipahami.

UCAPAN TERIMA KASIH

Penulis menyampaikan terima kasih kepada kedua orang tua yang selalu memberikan *support* dan doa restu dalam setiap langkah penelitian ini, serta kepada dosen pembimbing yang telah membantu dalam penelitian yang dilakukan.

DAFTAR PUSTAKA

[1] A. Ferdita Nugraha, R. F. A. Aziza, and Y. Pristyanto, "Penerapan metode Stacking dan Random Forest untuk Meningkatkan Kinerja Klasifikasi pada Proses Deteksi Web Phishing," *J. Infomedia*, vol. 7, no. 1, p. 39, 2022, doi: 10.30811/jim.v7i1.2959.

[2] A. N. S. Charan, Y. H. Chen, and J. L. Chen, "Phishing Websites Detection using Machine Learning with URL Analysis," *Proc. - 2022 IEEE World Conf. Appl. Intell. Comput. AIC 2022*, pp. 808–812, 2022, doi: 10.1109/AIC55036.2022.9848895.

[3] R. Kumar, R. Kumar, R. K. Sahu, R. Patra, and A. Ghosh, "Detection of Phishing Websites Using Machine Learning," *Smart Innov. Syst. Technol.*, vol. 313, no. 05, pp. 317–330, 2023, doi: 10.1007/978-981-19-8669-7_29.

[4] V. A. Windarni, A. F. Nugraha, S. T. A. Ramadhani, D. A. Istiqomah, F. M. Puri, and A. Setiawan, "Deteksi Website Phishing Menggunakan Teknik Filter Pada Model Machine Learning," *Inf. Syst. J.*, vol. 6, no. 01, pp. 39–43, 2023, doi: 10.24076/infosjournal.2023v6i01.1268.

[5] Fisa Wisnu Wijaya and D. Lomban, "Sistem Informasi Inventory Barang Menggunakan Metode Waterfall," *J. Inform. Teknol. dan Sains*, vol. 4, no. 3, pp. 247–254, 2022, doi: 10.51401/jinteks.v4i3.1963.

[6] R. R. Pradana, P. Sukarno, and E. M. Jadied, "Pendeteksi Phishing Menggunakan Metode Rule Based Attribute Checking Pendahuluan Studi Terkait," vol. 6, no. 1, pp. 2219–2226, 2019.

[7] M. Syahrizal and H. Haryati, "Perancangan Aplikasi Sistem Pakar Deteksi Kerusakan Mesin Alat Berat (Beko) Dengan Menerapkan Metode Teorema Bayes," *J. Media Inform. Budidarma*, vol. 2, no. 2, pp. 23–33, 2018, doi: 10.30865/mib.v2i2.596.

[8] A. Febriyani and M. Martanto, "Rancang Bangun Aplikasi Penjualan Kebutuhan Pokok Berbasis Web Pada Toko Khansaa," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 510–515, 2023, doi: 10.36040/jati.v7i1.6353.

[9] Muhammad Romzi and B. Kurniawan, "Pembelajaran Pemrograman Python Dengan Pendekatan Logika Algoritma," *JTIM J. Tek. Inform. Mahakarya*, vol. 03, no. 2, pp. 37–44, 2020.

[10] O. Adiputra and E. Setiawan, "Klasifikasi Malicious URL Menggunakan Algoritma Improved Random Forest dan Random Forest Berbasis Web," *J. Sains dan Inform.*, vol. 9, no. 1, pp. 8–14, 2023, doi: 10.22216/jsi.v9i1.1378.

[11] R. Rohi, J. Pote, and A. Talakua, "Perancangan Dan Implementasi Sistem Informasi Perpustakaan Berbasis Website Menggunakan Metode Waterfall Di Sd Masehi Kambaniru 2," *J. Inform. dan Tek. Elektro Terap.*, vol. 10, no. 2, pp. 63–70, 2022, doi: 10.23960/jitet.v10i2.2437.

[12] R. F. Ramadhan and R. Mukhaiyar, "Penggunaan Database Mysql dengan Interface PhpMyAdmin sebagai Pengontrolan Smarthome Berbasis Raspberry Pi," *JTEIN J. Tek. Elektro Indones.*, vol. 1, no. 2, pp. 129–134, 2020, doi: 10.24036/jtein.v1i2.55.

[13] A. P. Putra, F. Andriyanto, K. Karisman, T. D. M. Harti, and W. P. Sari, "Pengujian Aplikasi Point of Sale Menggunakan Blackbox Testing," *J. Bina Komput.*, vol. 2, no. 1, pp. 74–78, 2020, doi: 10.33557/binakomputer.v2i1.757.