PERANCANGAN APLIKASI DETEKSI *PHISHING* BERBASIS WEB MENGGUNAKAN ALGORITMA *DECISION TREE* PADA KOMUNITAS PANGLIMA DI PONDOK PESANTREN SUNAN PLUMBON

Proposal Penelitian Disusun Sebagai Persyaratan Penyusunan Skripsi Guna Memperoleh Gelar Sarjana Strata Satu (S1)

Diajukan oleh

KURNIA AINUN NAJAH

203100088

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA

YOGYAKARTA

2024

LEMBAR PERSETUJUAN

Proposal Skripsi

PERANCANGAN APLIKASI DETEKSI *PHISHING* BERBASIS WEB MENGGUNAKAN ALGORITMA *DECISION TREE* PADA KOMUNITAS PANGLIMA DI PONDOK PESANTREN SUNAN PLUMBON

Diajukan Oleh:

KURNIA AINUN NAJAH

203100088

Telah Memenuhi Syarat dan Disetujui untuk Diseminarkan di Program Studi Sistem Informasi

Fakultas Komputer dan Teknik Universitas Alma Ata

Yogyakarta, ...

Pebimbing

Tri Rochmadi, S.Kom., M.Kom.

NIK. 12201720538

LEMBAR PENGESAHAN

Proposal Skripsi

PERANCANGAN APLIKASI DETEKSI *PHISHING* BERBASIS WEB MENGGUNAKAN ALGORITMA *DECISION TREE* PADA KOMUNITAS PANGLIMA DI PONDOK PESANTREN SUNAN PLUMBON

Yang dipersiapkan dan disusun oleh:

KURNIA AINUN NAJAH

203100088

Telah memenuhi syarat dan dinyatakan disetujui untuk dilakukan penelitian pada

Tanggal.....

Ketua Penguji
Tanggal, ...

Tri Rochmadi, S.Kom., M.Kom.

Mengetahui,
Ketua Program Studi Sistem Informasi

Tri Rochmadi, S.Kom., M.Kom.
NIK. 12201720538

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

PERNYATAAN KEASLIAN PENELITIAN

Saya yang bertandatangan di bawah ini:

Nama	:	KURNIA AINUN NAJAH
NIM	:	203100088
Program Studi	:	SISTEM INFORMASI
Fakultas	:	KOMPUTER DAN TEKNIK

Dengan ini, saya menyatakan dengan sesungguhnya bahwa Skripsi yang berjudul "**Perancangan Aplikasi Deteksi *Phishing* Berbasis Web Menggunakan Algoritma *Decision Tree* Pada Komunitas Panglima Di Pondok Pesantren Sunan Plumbon**" adalah hasil karya peneliti sendiri, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain kecuali yang tertulis di dalam naskah ini dan disebutkan dalam daftar pustaka sesuai dengan kriteria etika penulisan ilmiah yang baku. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa Skripsi ini merupakan hasil karya orang lain maka saya bersedia menerima resiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, (20/Mei/2024)

Yang Membuat Pernyataan,

Materai

(Kurnia Ainun Najah)

EKSEKUTIF SUMMARY

Ancaman siber semakin berkembang seiring dengan kemajuan teknologi digital. Menyadari pentingnya keamanan dalam lingkungan internet, kebutuhan akan keamanan digital semakin meningkat, terutama di lingkungan komunitas Panglima di Pondok Pesantren Sunan Plumbon. Anggota komunitas sering kali menjadi korban penipuan *phishing* melalui pesan yang mengarah ke situs web palsu. Teknologi seperti deteksi URL *phishing* menggunakan algoritma *decision tree* dapat menjadi salah satu langkah mengurangi risiko serangan siber. Teknologi deteksi URL *phishing* akan menggunakan algoritma *decision tree* yang berbasis website dengan menggunakan metodologi *waterfall*, di mana tahap-tahapnya meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan *deployment*. Aplikasi diimplementasikan menggunakan *framework Flask* dan di-*hosting* di platform Heroku untuk kemudahan akses dan penggunaan. Melalui aplikasi ini, anggota komunitas Panglima dapat memasukkan URL yang mencurigakan, dan aplikasi akan menganalisisnya untuk menentukan apakah itu situs web *phishing* atau tidak. Hasil analisis ditampilkan kepada pengguna, agar dapat membantu menghindari ancaman *phishing*. Dengan demikian, aplikasi ini diharapkan dapat memberikan perlindungan yang lebih baik bagi anggota komunitas Panglima terhadap serangan *phishing* yang semakin canggih dan merugikan.

Kata Kunci: Website Deteksi, Phishing, Decision tree, Waterfall, Black box

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam era globalisasi informasi, di mana konektivitas dan akses internet telah menjadi bagian tak terpisahkan dari kehidupan sehari-hari. Penggunaan internet ini juga dimanfaatkan dalam kegiatan bisnis secara online, termasuk Usaha Mikro, Kecil, dan, Menengah (UMKM). Berdasarkan Survei *E-commerce* yang dilakukan Badan Pusat Statistik (BPS) tahun 2021, terdapat 63,52% responden pelaku usaha menggunakan layanan internet untuk pemasaran digital, meskipun hanya 8% yang menerima pelatihan, sedangkan 55,5% menggunakan internet untuk memesan bahan mentah dan 44,3% untuk komunikasi internal perusahaan[1]. Kemudahan dalam mengakses informasi, transaksi online, hingga pemasaran digital menjadi keuntungan yang diperoleh UMKM. Komunitas Panglima di Pondok Pesantren Sunan Plumbon Temanggung, telah memanfaatkan internet untuk meningkatkan daya saing dan memperluas pasar. Namun, internet juga memberikan tantangan berupa semakin beragamnya kejahatan siber di ranah digital [2].

Saat popularitas penggunaan media sosial meningkat di seluruh dunia, pelaku kejahatan dunia siber mulai mengeksploitasi kesempatan tersebut untuk meraih keuntungan, yang umumnya melalui cara-cara lama seperti *phishing* untuk pencurian data atau informasi yang selanjutnya dapat dimanfaatkan dari informasi

yang diperoleh [3]. *Phishing* merupakan bagian dari kejahatan siber yang semakin sering terjadi dengan maraknya kegiatan kriminal melalui jaringan komputer [4]. Ketika serangan *phishing* berhasil, munculnya tanda-tanda konkrit seperti kasus kehilangan akun online, penipuan finansial, atau bahkan identitas yang dicuri [5]. Pengguna seringkali menyadari terlambat bahwa mereka telah terjebak dalam perangkap *phishing*, dan dampaknya dapat mencakup kerugian finansial, hilangnya data pribadi, dan berkurangnya kepercayaan pada keamanan daring secara keseluruhan. Karena itu, memahami gejala dan tanda serangan *phishing* menjadi kunci penting dalam melindungi diri dan organisasi dari konsekuensi yang merugikan.

Peningkatan frekuensi serangan *phishing* terhadap UMKM telah menjadi tren yang mencemaskan dalam lanskap keamanan siber. Pada studi Cisco yang berjudul "Keamanan Siber untuk UMKM: Bisnis Asia Pasifik Mempersiapkan Pertahanan Digital", yang melibatkan lebih dari 3.700 UMKM di 14 kawasan Asia-Pasifik, termasuk Indonesi, pada April hingga Juli 2021, menunjukkan 56% UMKM mengalami kejahatan siber dalam setahun[6].

Komunitas Panglima di Pondok Pesantren Sunan Plumbon merupakan sebuah komunitas yang terdiri dari sejumlah pedagang yang aktif berjualan di sekitar lingkungan pesantren tersebut. Pada masa pandemi COVID-2019, kegiatan bisnis online meningkat. Beberapa anggota komunitas telah mengalami insiden serangan *phishing* pada masa itu, di mana mereka menerima pesan yang mengandung tautan yang mengarah ke situs web palsu yang bertujuan untuk mencuri informasi pribadi atau keuangan. Salah satu insiden yang dialami adalah

ketika anggota komunitas menerima pesan yang menjanjikan bantuan dana usaha melalui pengisian data diri, namun ternyata itu adalah situs web palsu. Dengan hal ini, diperlukan adanya perlindungan digital untuk anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon dari ancaman serangan *phishing*.

Penelitian ini bertujuan merancang sebuah aplikasi deteksi *phishing* berbasis web yang dapat membantu anggota komunitas dalam mengidentifikasi dan menghindari serangan *phishing*. Dengan menggunakan algoritma *Decision Tree*. Diharap aplikasi ini nantinya mampu memberi perlindungan tambahan dan meningkatkan kesadaran akan resiko *phishing* di kalangan anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon.

Penelitian ini akan menggunakan pendekatan waterfall dengan implementasi pertama-tama, melakukan pengumpulan dataset fitur-fitur URL *phishing* dan non-*phishing* dari sumber yang terpercaya. Selanjutnya Perancangan model deteksi menggunakan *Decision Tree* dengan menggunakan Google Colab sebagai platform pengembang. Pembuatan aplikasi web menggunakan Flask sebagai framwork perancangan. Proses *deployment* aplikasi akan menggunakan layanan *hosting* Heroku setelah tahap pengujian selesai. Implementasi dan pengujian aplikasi menggunakan metode *blackbox* untuk memastikan fungsionalitas yang tepat. Melalui penelitian ini, diharap dapat menciptakan lingkungan online yang lebih aman dan terpercaya bagi anggota komunitas dalam menjalankan aktivitas bisnis mereka.

1.2 Rumusan Masalah

Pada perancangan sistem deteksi *phishing* menggunakan algoritma *decision tree* berbasis web, terdapat permasalahan yang perlu diidentifikasi dan diselesaikan. Adapun rumusan masalah yang akan dijawab dalam penelitian ini yaitu bagaimana merancang sistem deteksi *phishing* berbasis website menggunakan algoritma *decision tree*, sehingga dapat membantu anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon yang dapat menghindari serangan *phishing*.

1.3 Batasan Masalah

Dalam rangka menjaga fokus dan ruang lingkup penelitian, batasan masalah pada penelitian ini mencakup hal-hal berikut:

1. Penelitian ini terbatas pada penggunaan dataset yang diperoleh dari repositori publik.
2. Aplikasi hanya akan berbasis website.
3. Proses perancangan aplikasi akan menggunakan metode *waterfall*.
4. Bahasa yang digunakan dalam perancangan adalah bahasa pemrograman HTML dan CSS untuk *frontend*, sedangkan *backend* menggunakan Python
5. Perancangan aplikasi menggunakan algoritma *decision tree*
6. Aplikasi ini akan difokuskan hanya mendeteksi URL.
7. Output aplikasi berupa hasil prediksi (*phishing* atau aman).
8. Heroku akan digunakan sebagai platform *hosting*
9. Pengujian aplikasi dilakukan dengan *blackbox*.

10. Penelitian difokuskan pada anggota komunitas Panglima yang berjualan di lingkungan Pondok Pesantren Sunan Plumbon.

1.4 Tujuan Penelitian

Tujuan utama dari penelitian ini untuk menghasilkan sebuah aplikasi deteksi *phishing* berbasis web menggunakan algoritma *decision tree* untuk memberi perlindungan tambahan kepada anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon dalam menghadapi resiko serangan *phishing*, serta meningkatkan kesadaran akan pentingnya keamanan digital.

1.5 Signfikasi Penelitian

Penelitian ini memiliki beberapa signifikansi yang dapat memberikan kontribusi positif dalam bidang keamanan siber dan perlindungan pengguna daring:

1. Membantu anggota komunitas Panglima dalam menghindari serangan *phishing*
2. Meningkatkan kesadaran akan pentingnya keamanan data pribadi.
3. Mengurangi resiko pencurian informasi pribadi dan finansial.
4. Memanfaatkan algoritma *decision tree*, diharapkan dapat melakukan deteksi *phishing*.
5. Hasil penelitian dapat menjadi acuan dan sumber referensi untuk penelitian-penelitian selanjutnya.
6. Memberikan edukasi tentang penggunaan teknologi dan keamanan siber kepada komunitas Panglima di Pondok Pesantren Sunan Plumbon.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Dalam tinjauan pustaka ini, akan dibahas beberapa penelitian terdahulu yang relevan dan dapat menjadi acuan dalam Perancangan aplikasi deteksi *phishing*.

Tabel 2.1 Penelitian Terdahulu

No	Judul	Peneliti dan Tahun Penelitian	Metode	Dataset	Hasil Penelitian
1.	Implementasi Metode Adasyn Dalam Deteksi Url Berbahaya Menggunakan *Machine Learning*: Demi Meningkatkan Keamanan Siber di Era Digital	Gilang Dwi Setyawan, Andrie Yuswanto, et al. (2023)	*Random Forest, Decision Tree, Naive Bayes*	Terdiri dari 2 fitur, yaitu "url" dan "type". Selanjutnya, diperluas menjadi 11 fitur. Nilai fitur mencakup "phishing", "benign", dan "defacement"	Pada hasil experiment menunjukkan bahwa algoritma *Random Forest mendapat akurasi 99%, Decision Tree 98%, dan Naive Bayes 96%.* [7]
2.	Deteksi Website	Vikky Aprelia	*Random Forest,*	Website UCI Machine	Naïve bayes memiliki

No	Judul	Peneliti dan Tahun Penelitian	Metode	Dataset	Hasil Penelitian
	Phishing Menggunakan Teknik Filter Pada Model *Machine Learning*	Windarni, Anggit Ferdita Nugraha, et al. (2023)	*Decision Tree, Naive Bayes*	Learning (11055 data dengan 30 fitur)	nilai akurasi sebesar 60,4%, decision tree 94,4% dan random forest sebesar 96,3%.[8]
3.	Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi Website Phishing	Nabila Bianca Putri dan Arie Wahyu Wijayanto (2022)	*Random Forest, Decision Tree, Naive Bayes, Support Vector Machine*	Data website: 1.353 data dengan 702 data situs non-phishing, 103 data situs mencurigakan, dan 548 data situs phishing	*Random Forest* memiliki akurasi 90,77%, Naïve Bayes 82,31%, Decision Tree 85,77%, dan Support Vector Machine 86,25%[9]
4.	*Comparative Analysis of Phishing Website*	Muhammad Fandru Al Rifqi, Mauli	*Logistic Regression, Decission Tree,*	Source EKG Kaggle	*Logistic Regression* memiliki perfoma

No	Judul	Peneliti dan Tahun Penelitian	Metode	Dataset	Hasil Penelitian
	Prediction Classification Algorithm Using Logistic Regression, Decision Tree, and Random Forest	Dina, et al. (2022)	*Random Forest*		akurasi 92.76%, *Decission Tree* 94.57%, dan *Random Forest* 97.10% [10]
5.	Komparasi Machine Learning Memprediksi Phising Dalam Keamanan Website	Aswan Supriyadi Sunge (2022)	*Decision Tree, Naïve Bayes, Multilayer Perceptron (Neural Network), K-Nearest Neighbor, Support Vector Machines*	Data publik dengan jumlah 11.055 data, 30 atribut.	Seluruh metode yang diusulkan memiliki akurasi tinggi. Namun, *Neural Network* lebih tinggi untuk mendeteksi phishing [11]
6.	Identifikasi Website Phishing dengan	Agung Susilo Yuda Irawan,	*Support Vector Machine, Decision*	Data public dengan tiga kategori yaitu legitimate,	Seluruh algoritma yang diusulkan

No	Judul	Peneliti dan Tahun Penelitian	Metode	Dataset	Hasil Penelitian
	Perbandingan Algoritma Klasifikasi	Nono Heryana, et al. (2021)	*Tree, Random Forest, dan Multilayer Perceptron*	suspicious, dan phishing	memiliki Perfoma yang baik. Namun, *Multilayer Perceptron* mendapat nilai tertinggi dengan akurasi 93,15% dan AUC 0,976[12]
7.	Rekomendasi Jurusan Dengan Menggunakan Decision Tree Pada Sistem Penerimaan Peserta Didik Baru SMK Widya Dharma Turen	Lazuardi Noorca Rachmadi, Aji Prasetya Wibawa, Utomo Pujianto (2021)	Decision Tree dengan mengguna kan waterfall sebagai metode pengemban gan.	Nilai raport calon peserta didik baru yang masih berupa numeric.	Menggunaka n pohon keputusan dapat membantu proses penerimaan peserta didik baru pada SMK Widya Dharma Turen.

Dari tinjauan penelitian terdahulu, dapat dilihat bahwa pendekatan deteksi *phishing* menggunakan algoritma *decision tree* telah menunjukkan hasil yang baik dan stabil. Karena itu, penelitian ini bertujuan untuk memberikan kontribusi tambahan dengan mengembangkan aplikasi deteksi *phishing* berbasis web menggunakan metode *waterfall* yang diharap mampu mempermudah pengguna dalam mendeteksi *phishing*.

2.2 *Phishing*

Phishing, sebuah istilah yang berasal dari kata dalam bahasa Inggris, yang secara harfiah dapat diterjemahkan sebagai penangkapan ikan atau memancing, merujuk pada praktik penipuan daring yang bertujuan untuk memperoleh informasi sensitif dari korban tanpa sepengetahuan mereka. Kejahatan dunia maya akan terus berkembang dalam berbagai bentuk, memberikan peluang yang lebih besar bagi pelakunya [13]. Modus operandinya melibatkan upaya untuk meniru entitas resmi atau organisasi terpercaya dalam komunikasi elektronik dengan tujuan menipu pengguna untuk mengungkapkan informasi pribadi seperti nama pengguna, kata sandi, atau rincian kartu kredit [14].

Praktik *phishing* sering kali terjadi dalam konteks layanan perbankan daring, di mana pelaku menyamar sebagai lembaga keuangan yang sah untuk mengelabui pengguna. Misalnya, pelaku dapat menciptakan formulir *login* palsu yang menyerupai tampilan resmi situs web bank. Ketika pengguna memasukkan kredensial mereka ke dalam formulir tersebut, informasi tersebut akan diserahkan kepada pelaku tanpa sepengetahuan pengguna. Namun, *phishing* tidak terbatas

pada sektor perbankan saja, ia juga menyerang melalui media sosial, email, pesan teks, dan situs web.

Phising dapat dibagi menjadi berbagai jenis berdasarkan motivasi pelaku dan sasaran yang ingin dicapainya. [14]

A. *Spear Phishing*, mengacu pada upaya *phising* yang ditujukan secara spesifik kepada target tertentu. Dalam jenis ini, pelaku memiliki peluang keberhasilan yang lebih tinggi karena sasaran mereka lebih jelas.
B. *Whaling*, mirip dengan *Spear Phishing*, namun targetnya adalah individu dengan posisi tinggi dalam organisasi, seperti pejabat atau eksekutif perusahaan. Pelaku *phishing* dalam jenis ini menggunakan media seperti subpoena untuk menakuti korban agar menghadap ke pengadilan.
C. *Clone Phishing*, merupakan jenis *phishing* konvensional di mana pelaku menggunakan email resmi untuk mengirimkan pesan yang identik dengan email asli kepada korban, tetapi dengan mengganti lampiran pesan tersebut.
D. *Covert Redirect*, merupakan teknik *phising* yang sangat halus di mana pelaku mengarahkan korban ke link yang seolah-olah resmi, tetapi sebenarnya menuju link yang dibuat oleh pelaku melalui pop-up login. Dalam teknik ini, target sulit untuk dikenali karena pelaku menggunakan link dan situs resmi dengan pop-up yang dimodifikasi, sehingga korban sulit membedakan antara form login asli dan palsu.

2.1 *Uniform Resource Locator* (URL)

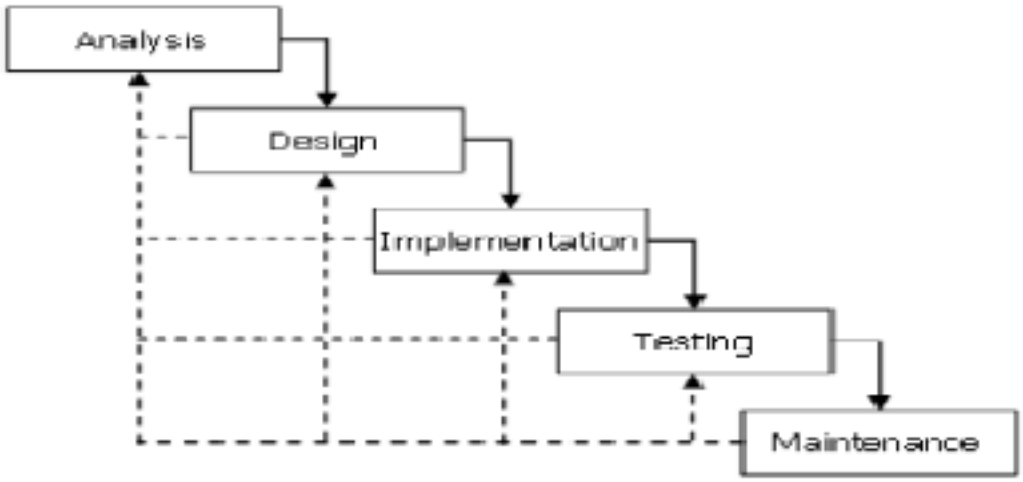
Uniform Resource Locator (URL) adalah pengidentifikasi unik yang terdiri dari protokol dan nama domain yang digunakan untuk menemukan halaman atau sumber daya di Internet. Namun, URL dapat dieksploitasi oleh penyerang untuk membuat URL berbahaya dan merugikan pengguna. URL berbahaya ini dirancang untuk melakukan serangan seperti spam, *phishing, malware*, dan *defacement*. Mereka dapat muncul di email, pesan teks, pop-up, atau iklan palsu dan dapat membahayakan komputer atau jaringan pengguna. Oleh karena itu, penting untuk mengembangkan sistem yang dapat mendeteksi URL berbahaya tersebut dan melindungi pengguna dari serangan dunia maya [15].

2.2 Website

Situs web merupakan sebuah koleksi halaman yang tergabung dalam satu domain, menyediakan beragam informasi yang dapat diakses oleh pengguna internet melalui mesin pencari. Konten yang disajikan dalam situs web mencakup gambar, ilustrasi, video, serta teks yang dirancang untuk memenuhi berbagai keperluan pengguna. Situs web menjadi sarana bagi pengguna untuk menjelajahi informasi, mengakses berita, mencari referensi, atau bahkan berkomunikasi secara daring. Melalui situs web, pengguna dapat memperoleh pengetahuan, mengeksplorasi produk dan layanan, atau bahkan terlibat dalam komunitas daring. Dengan demikian, situs web menjadi jendela virtual yang membuka akses ke berbagai jenis informasi dan interaksi di era digital [16].

2.3 Metode *Waterfall*

Siklus Hidup Pengembangan Perangkat Lunak atau *Software development* life cycle (SDLC) adalah metode yang digunakan dalam industri untuk merencanakan, membangun dan memelihara sistem informasi. Salah satu model SDLC yang paling umum adalah model Waterfall, yang terdiri dari lima tahapan yang harus diselesaikan secara berurutan. Model *Waterfall* adalah proses pengembangan perangkat lunak yang berurutan di mana kemajuan dipandang mengalir ke bawah seperti air terjun melalui tahapan-tahapan yang harus diselesaikan secara berurutan. Model ini pertama kali diusulkan oleh Winston W. Royce pada tahun 1970 untuk menggambarkan praktik perangkat lunak. Seluruh tahapan model *Waterfall* harus diselesaikan sebelum melanjutkan ke tahap berikutnya dan dapat diulangi hingga tahap tersebut selesai sepenuhnya. Secara umum model *Waterfall* terdiri dari analisis, desain, implementasi, pengujian dan pemeliharaan. Penjelasan mengenai metode *Waterfall* dapat dilihat pada Gambar 2.1 Metode Waterfall [17].



Gambar 2.1 Metode Waterfall [17]

Model SDLC Waterfall terdiri dari lima fase:

a) *Analysis*: Menentukan kebutuhan perangkat lunak secara lengkap dan komprehensif, melibatkan analis untuk mendefinisikan kebutuhan fungsional dan non-fungsional.
b) *Design*: Merencanakan solusi perangkat lunak, termasuk desain algoritma, arsitektur perangkat lunak, dan antarmuka pengguna.
c) *Implementation*: Merealisasikan kebutuhan bisnis dan desain menjadi program yang dapat dieksekusi melalui pemrograman dan deployment.
d) *Testing*: Memverifikasi dan memvalidasi perangkat lunak untuk memastikan sesuai dengan spesifikasi dan tujuan, serta melakukan debugging.
e) *Maintenance*: Memodifikasi perangkat lunak setelah pengiriman untuk memperbaiki kesalahan, meningkatkan kinerja, dan menyesuaikan dengan kebutuhan baru.

Setiap fase dalam model *Waterfall* harus diselesaikan sebelum melanjutkan ke fase berikutnya, dan dapat diulang hingga sempurna.

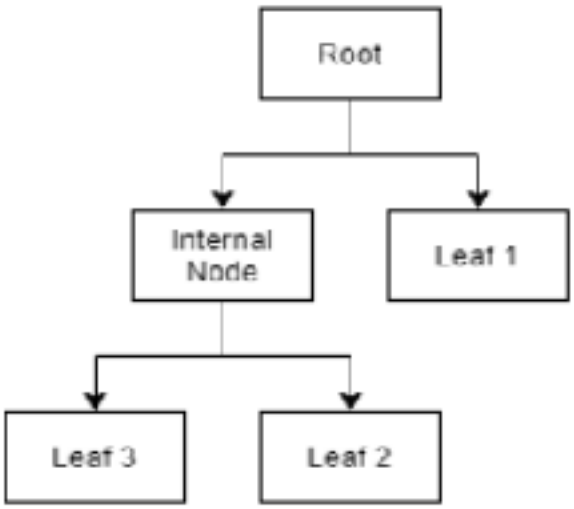
2.4 Data Sekunder

Data sekunder mengacu pada data yang sudah ada dan tersedia untuk dapat digunakan oleh peneliti lain. Dengan kata lain, data sekunder adalah data historis yang dikumpulkan pada masa lalu untuk tujuan tertentu. Misalnya, peneliti mengumpulkan data dan kemudian membagikannya kepada orang lain untuk digunakan dalam penelitian mereka sendiri. Data sekunder dapat berasal dari

berbagai sumber. Penting untuk dicatat bahwa data yang semula digunakan sebagai data primer dalam suatu penelitian, dapat menjadi data sekunder dalam penelitian lain jika data tersebut digunakan kembali oleh peneliti lain. Hal ini terjadi ketika data yang sama digunakan untuk tujuan penelitian yang berbeda [18].

2.5 Algoritma *Decision Tree*

Decision tree diperkenalkan pada tahun 1960-an oleh Fredkin. Singkatnya, *Decision tree* merupakan metode yang khusus dalam proses pengklasifikasian pada Text Mining [19]. *Decision tree* adalah sebuah struktur pohon, dimana setiap node internal (non-leaf) merepresentasikan pengujian atribut, setiap cabang merupakan suatu pembagian hasil uji, dan node daun (leaf) merepresentasikan kelompok kelas tertentu [9]. Model pohon keputusan banyak digunakan untuk pendeteksian web phishing [11]. Algoritma ini memiliki fungsi untuk mengeksplorasi data serta menemukan hubungan dari beberapa data yang ada di dalam dataset [8]. Metode pohon keputusan mengubah data menjadi bentuk pohon keputusan, kemudian mengubah pohon tersebut menjadi bentuk aturan, dan menyederhanakan aturan-aturan tersebut. Hasil pengujian ditetapkan pada setiap node internal, dimana setiap node internal mengembalikan hasil pengujian pada atribut [10].



Gambar 2.2 Decision Tree [9]

Root diambil dari dataset pada atribut yang paling berpengaruh atau memiliki kemampuan terbaik untuk memisahkan data menjadi kelompok yang berbeda. Pemilihan atribut untuk menjadi root biasanya dilakukan berdasarkan kriteria yang menghasilkan pemisahan yang paling baik antara kelas atau label data. Sedangkan internal root merupakan node internal (non-leaf) yang berada di bawah root. Node ini merepresentasikan pengujian atribut yang mengarahkan ke node-node cabang. Leaf 1, Leaf 2, Leaf 3 merupakan node daun (leaf) dalam struktur pohon keputusan. Node daun ini merupakan hasil dari pengujian atribut pada node internal, yang menunjukkan kelas atau label yang diprediksi untuk sampel data tertentu.

Algoritma decision tree sering digunakan untuk deteksi phishing karena bisa mengubah masalah kualitatif menjadi numerik. Data biasanya diubah menjadi nilai -1 dan 1, yang mewakili ketiadaan atau kehadiran fitur tertentu. Decision tree menggunakan struktur pohon biner untuk menganalisis data ini. Setiap langkah dalam pohon membagi data berdasarkan fitur-fitur, dan proses ini terus berlanjut

hingga prediksi akhir tercapai. Dengan cara ini, *decision tree* membantu mengidentifikasi pola dalam data *phishing* dengan mudah dan menghasilkan prediksi yang tepat.

2.6 SQLite

SQLite adalah sebuah library yang menerapkan mesin database self-contained, serverless, zero-configuration, dan transactional. Self-contained berarti SQLite dapat beroperasi tanpa memerlukan dukungan tambahan dari library eksternal atau sistem operasi. Serverless artinya akses ke database, baik itu untuk membaca maupun menulis, dapat dilakukan langsung dari file database tanpa perlu melalui proses server, dan tidak mendukung akses remote. Zero-configuration berarti SQLite dapat digunakan tanpa perlu proses instalasi sebelumnya. Transactional berarti SQLite adalah database transaksional yang menerapkan prinsip Atomic, Consistent, Isolated, dan Durable (ACID). Aplikasi database yang menggunakan SQL server memiliki kekurangan seperti memerlukan instalasi sebelum penggunaan, memerlukan server untuk memproses file database, dan membutuhkan ukuran memori yang lebih besar dibandingkan dengan penggunaan SQLite [20].

2.7 Python

Pencipta dari Python adalah Guido Van Rossum di Belanda pada tahun1990 dan dinamai berdasarkan acara televisi favoritnya "*Monty Python's Flying Circus*".

Pada awalnya Perancangan Python merupakan sebuah hobi, namun Python kemudian menjadi Bahasa pemrograman yang banyak digunakan karena kesederhanaannya, singkat, sintaksnya yang intuitif, dan Pustaka yang luas. Secara umum, Python mendukung pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Kelebihan Python adalah [21]:

a) Banyak koleksi perpustakaan dengan modul siap pakai untuk berbagai keperluan.
b) Struktur bahasanya jelas, sederhana dan mudah dipelajari.
c) Objek.
d) Sistem memori otomatis.
e) Bersifat modular.

2.8 *HyperText Markup Language* (HTML)

HTML (*Hyper Text Markup Language*) adalah bahasa yang digunakan untuk mengatur tata letak halaman web. Saat membuat website, HTML sangatlah penting karena merupakan fondasi dari website. File HTML disimpan dengan ekstensi .html dan dapat dibuka dengan browser web. HTML adalah dasar untuk membuat halaman web, di mana dapat mengatur elemen halaman dan membuat judul [22].

2.9 *Cascading Style Sheets* (CSS)

Cascading Style Sheets atau CSS, adalah bahasa pemrograman yang digunakan untuk mendesain halaman web. Dalam proses desain web, CSS digunakan untuk menambahkan gaya dan tampilan pada elemen halaman menggunakan tag seperti ID dan kelas. CSS direkomendasikan oleh World Wide Web Consortium (W3C) pada tahun 1996, dan browser seperti Internet Explorer dan Netscape telah merilis versi terbaru yang mendukung standar CSS. Terdapat tiga versi utama CSS: CSS1, CSS2, dan CSS3. Selain digunakan pada HTML dan XHTML, CSS juga dapat digunakan untuk mendesain tampilan aplikasi Android. CSS memungkinkan untuk mengonfigurasi dan memilih tampilan elemen dalam dokumen web [22].

2.10 *Unified Modeling Language* (UML)

Unified Modeling Language (UML) adalah bahasa standar yang digunakan untuk menjelaskan dan menggambar rencana serta desain sistem secara visual. Dikembangkan oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson, UML memberikan cara yang jelas untuk berkomunikasi antara pengguna dan pengembang, serta antara tim pengembangan perangkat lunak. UML memungkinkan untuk melihat struktur dan interaksi sistem dengan lebih mudah, dan memastikan keselarasan antara ide desain dan pelaksanaannya. Meskipun UML tidak memberikan langkah-langkah spesifik untuk analisis dan desain [23]. Berikut notasi dan artifak penting dalam UML:

A. *Use Case* Diagram

Diagram *use case* (UCD) menggambarkan apa yang dilakukan sistem yang sedang dibangun dan siapa yang berinteraksi dengan sistem. UCD bertindak sebagai dokumen kontrak antara pelanggan, pengguna, dan pengembang. Pengguna menggunakan UCD untuk memahami sistem dan memastikan bahwa sistem benar-benar menyelesaikan masalah yang telah atau sedang terjadi. Pengembang menggunakan UCD sebagai acuan tepat dalam pengembangan sistem. Secara umum diagram use case terdiri dari elemen-elemen seperti aktor, use case, dependensi, generalisasi, dan hubungan. UCD ini memberikan pandangan statis dari sistem yang sedang dibangun dan merupakan artefak dari proses analisis [23]. Notasi pada use case dapat dilihat pada tabel 2.2 Notasi Diagram *Use Case*.

Tabel 2.2 Notasi Diagram *Use Case* [24]

No.	Nama	Notasi	Keterangan
1.	*Actor*		Segala hal yang berinteraksi dengan sistem aplikasi komputer
2.	*Ekstends*		Relasi untuk menambahkan fitur tambahan walau tanpa usecase utama
3.	*Include*		Relasi untuh menambahkan fitur tambah dan hanya dapat dijalankan ketika usecase utama terpenuhi.
4.	*Use Case*		Tindakan aktor dalam sistem untuk mencapai tujuan tertentu
5.	*Association*		Relasi yang menghubungkan satu objek dengan objek lainnya
6.	*System*		Menampilkan atau menggunakan data paket untuk menyajikan

No.	Nama	Notasi	Keterangan
			informasi atau fitur tertentu kepada pengguna.

B. Activity Diagram

Diagram aktivity disebut juga diagram fungsional merupakan representasi visual dari aliran kerja atau aktivitas dalam suatu sistem atau bisnis. Diagram ini sering digunakan untuk berbagai tujuan, termasuk desain proses bisnis [24]. Notasi pada activity diagram dapat dilihat pada tabel 2.3 Notasi *Activity Diagram.*

Tabel 2.3 Notasi *Activity* Diagram [24]

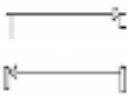
No.	Nama	Notasi	Keterangan
1.	*Actifity*		Menunjukkan interaksi antar kelas antarmuka
2.	*Action*		Kondisi sistem saat melakukan aksi tertentu
3.	*Initial Node*		Menandai awal pembentukan atau proses suatu objek
4.	*Actifity Final Node*		Menunjukkan akhir dari pembentukan objek.
5.	*Fork Node*		Aliran sistemyang pada tahp tertentu bercabang menjadi beberapa aliran.

C. Sequence Diagram

Sequence diagram menunjukkan urutan proses sistem untuk mencapai tujuan tertentu, seperti interaksi antar kelas, aktivitas yang terlibat, urutan aktivitas, dan informasi yang diperlukan. Membuat diagram sequence sangat penting dalam perancangan karena memandu program dan menunjukkan aliran kendali program. Desain urutan biasanya terdiri dari elemen objek,

interaksi, dan pesan. Interaksi menghubungkan objek dengan pesannya yang mengungkapkan karakteristik dinamis dari sistem yang dibangun [23]. Notasi pada diagram Sequence dapat dilihat pada tabel 2.4 Notasi Diagram Sequence.

Tabel 2.4 Notasi *Sequence* Diagram [24]

No.	Nama	Notasi	Keterangan
1.	*Life Line*		Representasi dari objek atau entitas yang berinteraksi satu sama lain dalam sebuah diagram.
2.	*Messager*		Detail komunikasi antar objek yang mencakup informasi tentang kegiatan atau aktivitas yang terjadi.

2.11 Framework Flask

Framework Flask merupakan framwork web python yang ringan dan fleksibel yang digunakan untuk membangun aplikasi web. Flask sering disebut "*microframework*" karena ringkas dan mudah dipahami, sehingga memudahkan pengembang untuk mempelajari kode sumbernya dan memahami seluk-beluknya. Flask sengaja tidak memiliki dukungan bawaan untuk tugas-tugas Tingkat tinggi seperti akses database dan otentikasi pengguna, sehingga memberikan ruang bagi pengembang untuk memasukkan ekstensi yang selaras dengan kebutuhan proyek [25].

2.12 Heroku

Heroku adalah salah satu platform pertama yang menawarkan Service as a Service (PaaS) dan telah ada sejak tahun 2007. Heroku sangat fleksibel dan mendukung banyak bahasa pemrograman. Untuk menyebarkan aplikasi ke Heroku, pengembang menggunakan Git untuk mendorong aplikasi ke server Heroku Git, yang secara otomatis memulai instalasi, konfigurasi, dan penerapan. Heroku menggunakan unit dyno untuk mengukur penggunaan dan menentukan biaya layanan. Ada dua jenis dyno: Web Dyno untuk server web dan Work Dyno untuk tugas latar belakang. Heroku juga menawarkan banyak plug-in dan add-on untuk berbagai layanan, seperti dukungan database dan email, yang mempermudah pengelolaan dan pengembangan aplikasi [25].

2.13 *Google Interactive Notebook* (Google Colab)

Bahasa pemrograman Python yang berkembang pesat membuat Google tertarik untuk mengembangkan *Integrated Development Environment* (IDE) secara daring yang dikenal Google Colab. Situs resmi ini dapat diakses di www.colab.research.google.com. Google Colab menggunakan jenis lingkungan Jupyter Notebook dengan eksensi file *ipynb. Google Colab menyediakan beragam perpustakaan yang penting untuk penelitian, termasuk Keras, TensorFlow, NumPy, Pandas, dan Matplotlib untuk membuat visualisasi data. Selain itu, Google Colab dilengkapi dengan fitur penyimpanan terintegrasi dengan Google Drive [26].

2.14 Visual Studio Code

Visual Studio Code adalah sebuah software editor kode yang ringan namun kuat yang berjalan didesktop dan tersedia untuk Windows, macOS,dan Linux. Muncul dengan dukungan built-in untuk JavaScript, TypeScript dan Node.js dan memiliki ekosistem ekstensi yang kaya untuk bahasa lain (seperti C ++, C #, Java, Python,PHP, Go) dan runtime (seperti .NET danUnity) [21].

2.15 *Black Box Testing*

Pengujian Black Box merumakan teknik pengujian perangkat lunak yang digunakan untuk menentukan fungsionalitas dari aplikasi. Teknik ini berfokus pada input yang diberikan ke aplikasi dan output yang dihasilkan untuk setiap nilai input. Pengujian ini tidak perlu mengetahui cara kerja internal aplikasi.

Teknik pengujian Black Box ada berbagai macam, yaitu [27]:

a) *Equivalence Partitioning*: merupakan teknik untuk merancang kasus uji. Teknik ini membagi semua nilai input ke dalam beberapa partisi, termasuk partisi yang valid dan tidak valid. Kasus uji dirancang dari setiap partisi untuk menemukan kesalahan.

b) *Boundary Value Analysis*: Teknik ini digunakan untuk merancang kasus uji untuk menemukan kesalahan. Teknik ini menggunakan nilai batas atau nilai yang mendekati batas dari domain input sebagai data uji. Kasus uji

dirancang untuk nilai batas yang valid dan tidak valid. Satu kasus uji dipilih dari setiap nilai batas.

c) *Cause Effect Graphing*: Ini adalah teknik desain pengujian perangkat lunak yang berfokus pada perilaku eksternal sistem. Teknik ini menentukan hubungan logis antara kondisi input dan output dengan bantuan operator Boolean. Nilai input mewakili 'Penyebab' dan nilai output mewakili 'Efek'. Hubungan antara Penyebab dan Efek membantu membuat kasus uji.

d) *Decision Table Based Testing*: Ini adalah teknik yang baik untuk menangani banyak input dan output yang sesuai. Tabel Keputusan memiliki sifat kelengkapan; ia mencakup semua kemungkinan nilai dari kondisi. Teknik ini sangat berguna untuk alur bisnis yang kompleks untuk diubah menjadi kasus uji.

e) *Error Guessing*: Ini adalah teknik berdasarkan asumsi dan dugaan. Penguji berpengalaman menemukan cacat berdasarkan pengalaman mereka. Keberhasilan teknik ini sepenuhnya bergantung pada keterampilan penguji, seorang penguji yang baik tahu di mana dan jenis cacat apa yang paling sering ditemukan.

2.16 Kerangka Pemikiran

Pada gambar 2.3 ini merupakan sebuah kerangka pemikiran yang digunakan untuk penulisan penelitian ini sebagai tahap dalam melakukan penelitian.

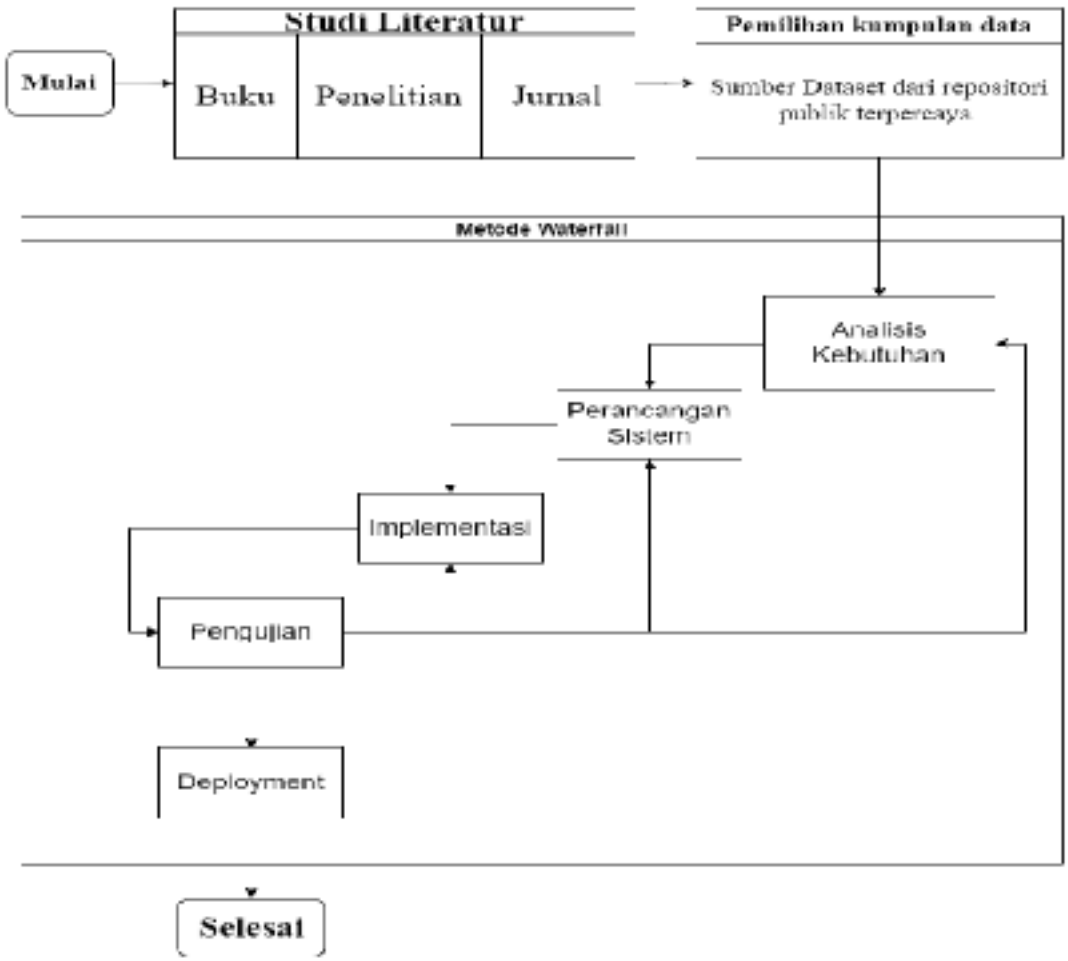
Gambar 2.3 Kerangka Pemikiran Penelitian

BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Desain penelitian digunakan agar penelitian terlaksana secara terstruktur. Berikut desain penelitian dari proyek Perancangan aplikasi deteksi *phishing* berbasis web menggunakan algoritma *decision tree* pada komunitas Panglima di Pondok Pesantren Sunan Plumbon dapat dilihat pada gambar 3.1 Desain Penelitian.



Gambar 3.1 Desain Penelitian

Penelitian ini dimulai dengan studi literatur dari berbagai buku, penelitian, dan jurnal terkait penelitian ini. Setelah itu, dataset yang relevan dipilih untuk melatih model deteksi phishing. Metode Perancangan yang digunakan adalah metode waterfall, dimulai dari tahap analisis kebutuhan, lalu perancangan sistem, hingga implementasi aplikasi deteksi phishing dengan integrasi algoritma *Decision Tree* dan pembuatan aplikasi menggunakan Flask. Setelah selesai, dilakukan pengujian fungsionalitas aplikasi menggunakan black box, dan jika terdapat kesalahan, akan kembali ke tahap sebelumnya untuk perbaikan. Akhirnya, aplikasi akan didistribusikan dan di-deploy untuk digunakan oleh anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon.

3.2 Subjek Penelitian

Penelitian ini akan difokuskan pada pengembangan aplikasi deteksi phishing berbasis web menggunakan algoritma decision tree. Subjek penelitan ini adalah pedagang yang menjadi anggota komunitas Panglima dan berada di lingkungaan Pondok Pesantren Sunan Plumbon yang akan menggunakan aplikasi ini untuk melindungi diri dari serangan phishing.

3.3 Lokasi Penelitian

Lokasi penelitian ini di lingkungan sekitar Pondok Pesantren Sunan Plumbon, Krajan, Tembarak, Temanggung, Jawa Tengah, Indonesia, dimana pedagang yang menjadi anggota Panglima beroperasi.

3.4 Metode Pengumpulan Data

Dalam penelitian ini, metode pengumpulan data yang digunakan adalah menggunakan data sekunder. Data akan diambil dari dataset penelitian terdahulu dari sumber terpercaya. Dataset ini telah dikumpulkan dan disusun oleh peneliti sebelumnya dan terdiri dari sampel-sampel URL yang digunakan untuk mendeteksi halaman web phishing. Penggunaan dataset yang telah teruji dan terverifikasi akan memastikan konsistensi dan validasi hasil penelitian.

Dataset tersebut akan terdiri dari dua jenis URL, yaitu URL *phishing* dan URL sah (*legitimate*). URL phishing akan mencakup sampel-sampel URL yang telah terverifikasi sebagai situs *phishing*, sedangkan URL sah akan mencakup sampel-sampel URL yang terverifikasi sebagai situs web yang aman dan dapat dipercaya.

Penggunaan dataset yang kredibel dan terpercaya akan memastikan keakuratan dan keandalan pelatihan model *decision tree* dalam mendeteksi pola-pola serangan phishing. Selain itu, penggunaan dataset yang telah ada juga akan meminimalkan waktu dan upaya yang diperlukan untuk pengumpulan data, sehingga mempercepat proses penelitian.

Dengan mengambil dataset dari penelitian terdahulu yang telah terbukti validitasnya, diharapkan hasil penelitian ini dapat memberikan kontribusi yang berarti dalam Perancangan sistem deteksi URL *phishing*.

3.5 Metode dan Tahap Perancangan Aplikasi

Aplikasi berbasis web ini akan menggunakan algoritma *decision tree* dengan menggunakan model pengembangan waterfall dan dengan pengujian black box untuk memastikan fungsionalitas aplikasi berjalan sesuai dengan yang diharapkan. Berikut rincian tahapan dari model pengembangan waterfall di aplikasi ini.

3.5.1 Analisis Kebutuhan

Analisis merupakan tahap awal dari metode *waterfall* dalam pengembangan aplikasi ini. Tahap ini digunakan untuk mengidentifikasi dan menentukan kebutuhan pengguna akan software yang dirancang.

A. Kebutuhan Fungsional

Analisis kebutuhan fungsional ini mencakup empat halaman utama dalam aplikasi: *Home page*, *searching page*, dan *result page, history page*. Berikut adalah penjelasan dari masing-masing halaman beserta kebutuhan fungsionalnya.

1) *Home page*: Menampilakan ringkasan layanan deteksi phishing yang ditawarkan oleh aplikasi. Menyediakan navigasi ke halaman *Searching page.*
2) *Searching page*: Menyediakan form input untuk pengguna memasukkan URL yang ingin diperiksa. Melakukan validasi terhadap input URL untuk memastikan formatnya benar sebelum dianalisis. Menyediakan tombol submit untuk mengirim URL yang telah dimasukkan untuk diproses oleh sistem deteksi *phishing*.

3) *Result page*: Menampilkan hasil deteksi apakah URL tersebut terindikasi phishing atau tidak.
4) *History page*: Menampilkan histori deteksi yang dilakukaan oleh pengguna.

B. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional berkaitan dengan batasan atau fungsi yang ditawarkan sistem [28]. Berikut kebutuhan non-fungsional dari aplikasi berbasis web deteksi *phishing*.

Tabel 3.1 Kebutuhan Non-Fungsional

Kategori	Kebutuhan	Deskripsi
Kinerja (*Performance*)	Efisiensi Algoritma	Algoritma *decision tree* harus dioptimalkan agar dapat melakukan deteksi dengan efisien.
Kemudahan Penggunaan (*Usability*)	Antarmuka Pengguna (*User Interface*)	Antarmuka aplikasi harus intuitif dan mudah digunakan oleh pengguna tanpa memerlukan pelatihan khusus.

C. Alat dan Bahan Penelitian

Alat dan bahan diperlukan untuk menunjang penelitian Perancangan aplikasi deteksi *phishing* berbasis web menggunakan algoritma *decision tree* pada komunitas panglima di Pondok Pesantren Sunan Plumbon.

1) Alat Penelitian

- Perangkat Keras:

Berikut adalah perangkat keras yang digunakan dalam penelitian ini:

Tabel 3.2 Perangkat Keras

Laptop:	Lenovo ideapad 330	*Device name:*	KurniaAinunN
		Processor:	*Intel(R) Celeron(R) N4000 CPU @ 1.10GHz 1.10 GHz*
		RAM:	*8,00 GB*
		System type:	*64-bit operating system, x64-based processor*
		Operating System:	*Windows 11 Pro, Version 21H2, OS build 22000.2538*
Mouse:	*Fantech Blake X17 mouse*		

- Perangkat Lunak:

Berikut adalah perangkat lunak yang digunakan dalam penelitian ini:

Tabel 3.3 Perangkat Lunak

Nama	Keterangan
Visual Studio Code:	Digunakan untuk perancangan kode dan penyusunan skrip Python untuk implementasi model deteksi phishing.
Google Colab:	Platform cloud yang digunakan untuk melatih model jaringan saraf menggunakan GPU yang tersedia secara gratis.
Kaggle	Website penyedia dataset

Nama	Keterangan
Figma:	Alat untuk membuat desain antarmuka pengguna (UI).
Drawio:	Digunakan untuk merancang dan membuat diagram kerangka pemikiran, alur penelitian, dan desain lainnya.
Microsoft Excel:	Melihat dataset

2) Bahan Penelitian

Berikut adalah beberapa bahan penelitian yang digunakan untuk proyek ini:

Tabel 3.4 Bahan Penelitian

Dataset:	Dataset penelitian terdahulu dari sumber terpercaya. Dataset ini telah dikumpulkan dan disusun oleh peneliti sebelumnya dan terdiri dari sampel-sampel URL
Python:	Bahasa pemrograman utama yang digunakan untuk mengembangkan dan melatih model deteksi phishing menggunakan algoritma decision tree.
Libraries Python:	Berbagai pustaka Python akan digunakan untuk implementasi model dan pemrosesan data. a) scikit-learn: untuk machine learning, digunakan untuk melatih model Decision Tree. b) joblib: Library untuk menyimpan dan memuat model machine learning.
Flask:	Framework Python yang digunakan untuk membangun antarmuka web untuk sistem deteksi phishing yang akan diimplementasikan.
Heroku:	Platform cloud untuk deployment aplikasi web Flask.
SQLite	Database self-contained yang digunakan untuk menyimpan histori deteksi URL phishing.

3.5.2 Perancangan Sistem

Tahap kedua dari metode waterfall setelah analisis yaitu perancangan sistem yang bertujuan merancang bagaimana aplikasi bekerja dan ditampilkan.

A. Pengembangan Logika Aplikasi

Metode pengembangan logika aplikasi ini menggunakan Unified Modeling Language (UML). Berikut model UML yang akan digunakan:

1) *Use Case* Diagram

Sebuah diagram *use case* (diagram kasus penggunaan) adalah representasi visual dari interaksi antara sistem dan pemangku kepentingan (aktor) dalam sistem. Ini menunjukkan fungsionalitas sistem dari sudut pandang pengguna atau aktor eksternal yang terlibat. Berikut adalah contoh *use case* diagram untuk sistem deteksi URL phishing menggunakan decision tree:

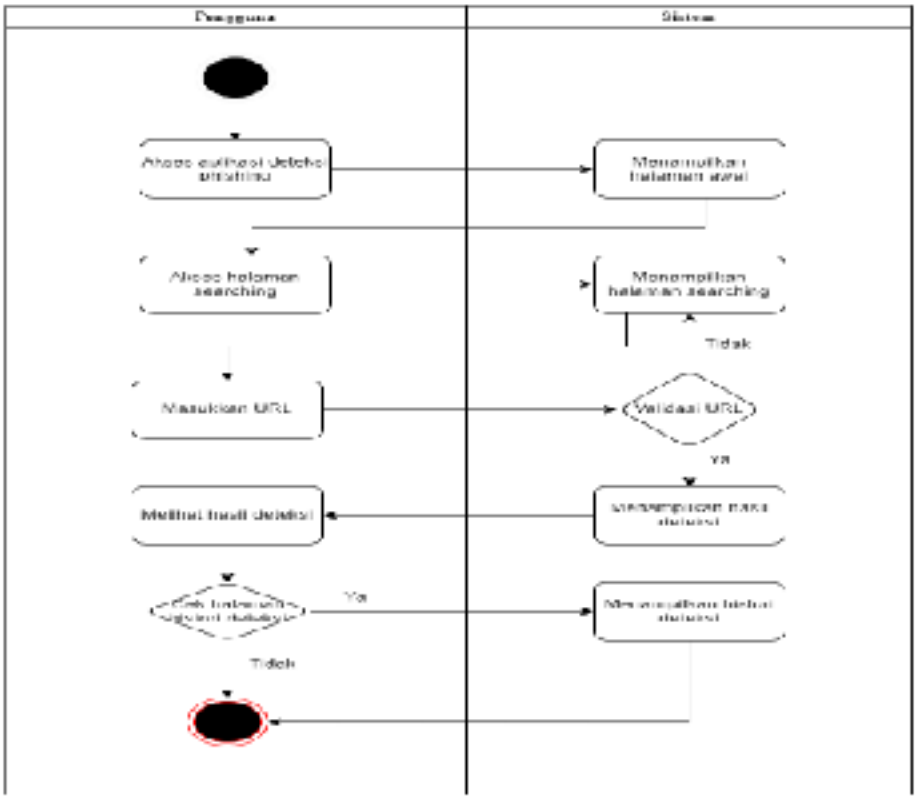
Gambar 3.2 Use Case

Dalam diagram *use case* di atas, terdapat satu aktor eksternal yaitu "Pengguna", yang berinteraksi dengan sistem. Pengguna berinteraksi dengan aplikasi web melalui halaman awal, memasukkan URL yang ingin mereka periksa. Aplikasi kemudian menganalisis URL tersebut menggunakan model decision tree yang telah dilatih sebelumnya. Hasil

dari analisis tersebut ditampilkan kepada pengguna. Pengguna dapat memilih aksi tambahan untuk memeriksa URL lainnya atau melihat histori deteksi.

2) *Activity* Diagram

Pada gambar 3.3 *Activity* Diagram menggambarkan interaksi antara pengguna dan sistem dalam proses deteksi phishing.



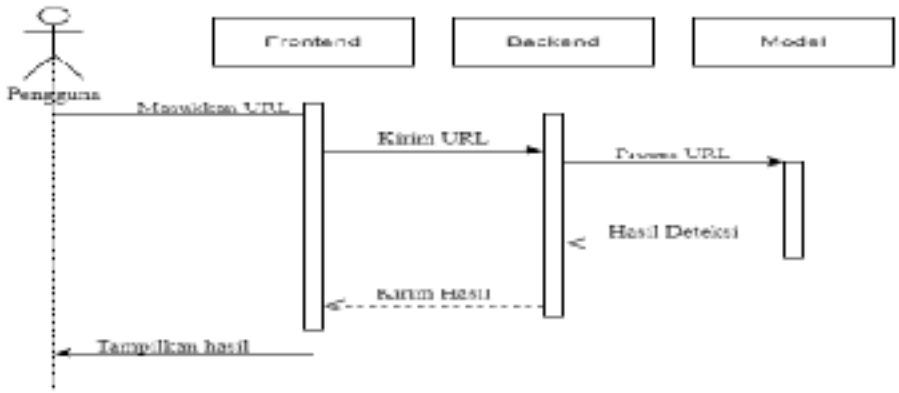
Gambar 3.3 *Activity* Diagram

Pengguna membuka aplikasi web deteksi phishing melalui browser. Sistem akan menampilkan halaman awal. Selanjutnya pengguna mengakses halaman searching dan halaman tersebut akan ditampilkan oleh sistem. Pengguna mengetikkan URL yang ingin diperiksa ke dalam input form di halaman searching. Selanjutnya sistem akan memvalidasi

URL yang sudah di input pengguna apakah sesuai format atau tidak. Jika tidak sesuai maka sistem akan tetap menampilkan halaman searching dan pengguna diharap memasukkan URL yang valid. Jika URL sesuai maka sistem akan kehalaman result untuk menampilkan hasil dan pengguna dapat melihatnya. Selanjutnya akses histori page jika pengguna ingin mengakses halaman histori untuk melihat histori deteksi.

3) *Squence* Diagram

Berikut adalah urutan diagram (sequence diagram) yang menggambarkan interaksi dari pengguna hingga menampilkan hasil deteksi phishing:



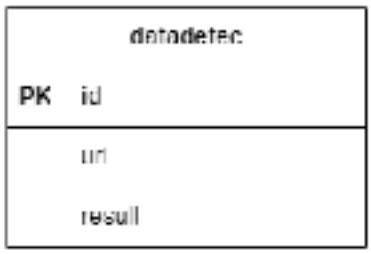
Gambar 3.4 Squence Diagram

Dalam sequence diagram, pengguna memasukkan URL yang ingin diperiksa ke dalam aplikasi. Selanjutnya frontend mengirimkan URL yang dimasukkan ke server backend. Server backend akan menerima URL dan memprosesnya menggunakan model Decision Tree yang telah dilatih. Model Decision Tree akan menganalisis URL dan mengembalikan hasil deteksi. Server backend mengirimkan hasil deteksi

kembali ke aplikasi frontend. Terakhir, hasil deteksi akan ditampilkan kepada pengguna.

B. Database SQLite

Aplikasi ini menggunakan sistem database SQLite dengan rincian dapat dilihat pada tabel 3.5 Desain Database.

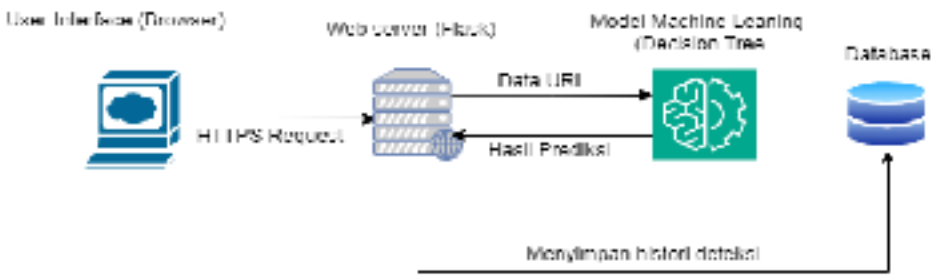


Gambar 3.5 Desain Database

- id: Kolom integer yang bertindak sebagai primary key, dengan auto-increment untuk menghasilkan nilai unik secara otomatis.
- url: Kolom teks yang menyimpan URL yang diperiksa.
- result: Kolom teks yang menyimpan hasil deteksi (misalnya, "phishing" atau "aman").

C. Arsitektur Aplikasi

Tahap arsitektur aplikasi dapat dilihat pada gambar 3.6 Arsitektur Aplikasi.



Gambar 3.6 Arsitektur Aplikasi

Pengguna berinteraksi dengan aplikasi melalui antarmuka web yang melalaui Browser, dimana pengguna dapat memasukkan URL yang ingin diperiksa. Permintaan dari pengguna diteruskan ke Web Server (Flask) yang menangani pemrosesan data dan memanggil Machine Learning Model (Decision Tree) untuk memprediksi apakah URL tersebut phishing atau tidak. Hasil dari prediksi akan dikembalikan ke web server dan ditampilkan kembali kepada pengguna, memberikan informasi apakah URL tersebut terindikasi sebagai phishing atau tidak. Dengan demikian, pengguna dapat dengan mudah mengetahui apakah suatu URL aman atau berpotensi phishing. Web server juga bertanggung jawab untuk menyimpan histori deteksi dalam database.

D. Tahap Desain Sistem

1) Desain Proses

Merancang proses deteksi URL pada website, perlu memperhatikan langkah-langkah utama yang diperlukan dari awal hingga akhir. Berikut adalah desain proses yang digunakan:

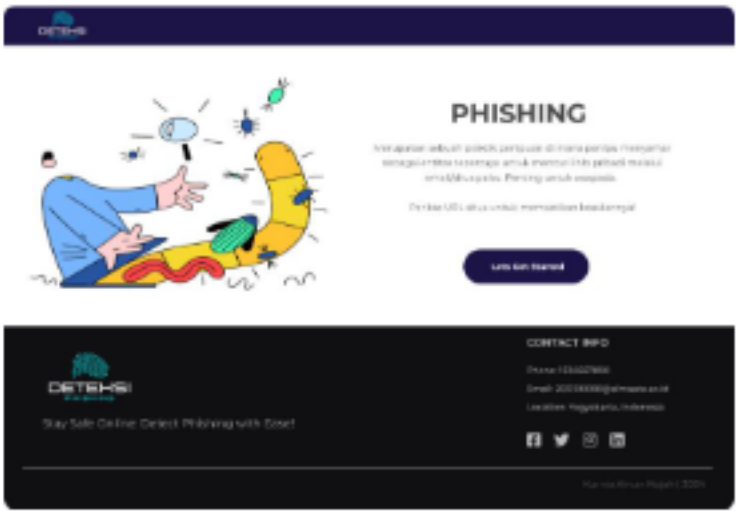
- Inisialisasi: Pengguna mengunjungi halaman utama website deteksi URL. Terdapat informasi tentang layanan yang ditawarkan di halaman pertama.
- Input URL: Pengguna memasukkan URL yang ingin diperiksa. Validasi input dilakukan untuk memastikan format URL benar.

- Proses Deteksi: URL yang telah divalidasi diproses untuk persiapan deteksi phishing. Disini model deteksi phishing diterapkan pada URL yang diproses.
- Evaluasi Hasil: Hasil deteksi phishing diterima dari model. URL diklasifikasikan sebagai phishing atau aman berdasarkan hasil deteksi.
- Tampilan Hasil: Hasil deteksi ditampilkan kepada pengguna. Pengguna diberikan informasi tentang status URL yang diperiksa.
- Interaksi Pengguna: Pengguna dapat memilih untuk melihat histori dari deteksi atau kembali dan memeriksa URL lain.
- Penyelesaian: Proses selesai dan sistem kembali ke kondisi awal untuk menerima permintaan deteksi URL berikutnya.

2) Rancangan User Interface

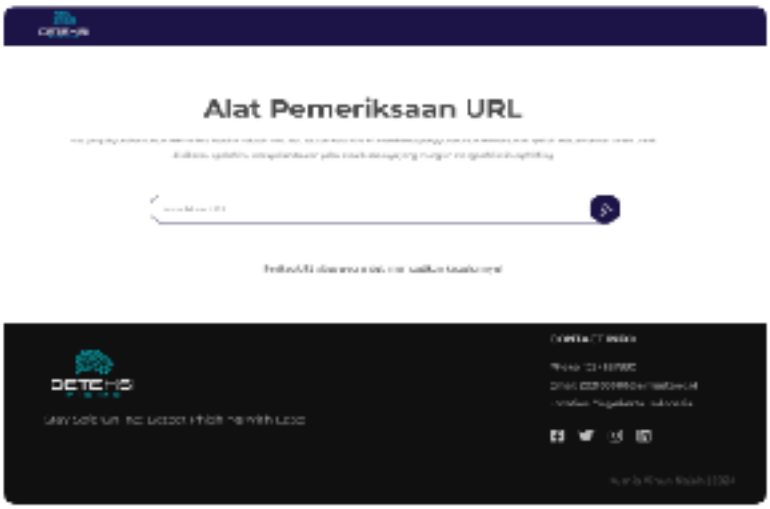
Desain UI akan dibuat menggunakan Figma untuk memastikan tampilan yang menarik dan mudah dipahami oleh pengguna. Berikut adalah gambaran singkat dati setiap halaman website yang ingin dibuat:

- *Home Page*: Ini merupakan halaman utama dari website. Berisi ringkasan tentang layanan yang ditawarkan. Serta tautan tombol navigasi untuk menuju halaman lainnya seperti halaman pencarian.



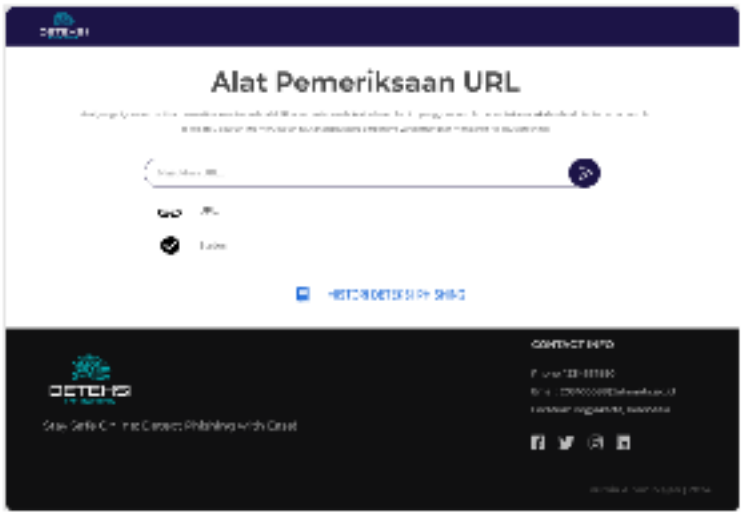
Gambar 3.7 Desain *Home Page*

- *Searching Page*: Halaman ini merupakan tempat untuk memasukkan URL yang ingin diperiksa. Berisikan tentang formulir atau kotak teks yang dapat digunakan untuk mengetik atau menempelkan URL yang ingin dideteksi. Setelah pengguna memasukkan URL dan mengirim permintaan, website akan mengarahkan pengguna pada halaman hasil.



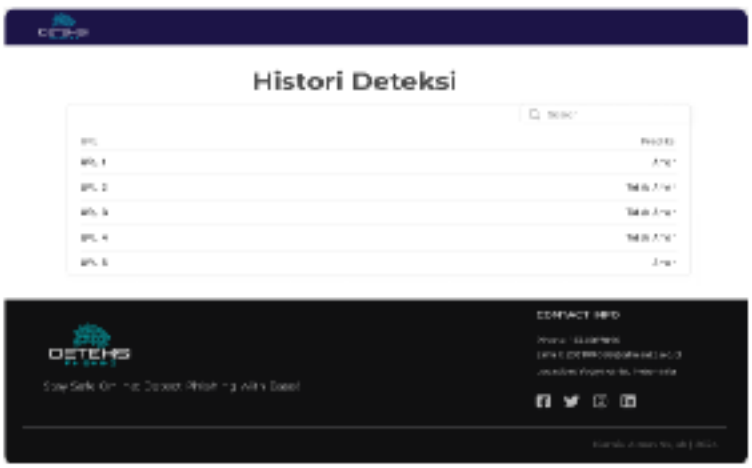
Gambar 3.8 Desain *Searching Page*

- *Results Page*: Halaman ini menampilkan hasil dari deteksi URL yang dilakukan oleh *system*. Hasil berupa label "phishing" atau "aman". Terdapat formulir atau kotak teks jika ingin mendeteksi URL yang lain. Pada halaman ini juga terdapat tautan menuju halaman histori jika ingin melihat histori deteksi.



Gambar 3.9 Desain *Result Page*

- *History Page*: Halaman ini menampilkan history dari deteksi URL yang sudah dilakukan para pengguna. Pengguna dapat mencari



Gambar 3.10 Desain *History Page*

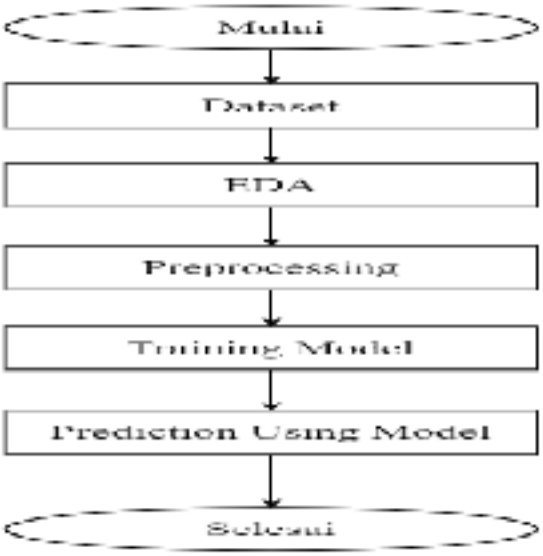
3.5.3 Implementasi

Implementasi adalah tahap di mana desain diubah menjadi sebuah aplikasi yang fungsional agar perangkat lunak dapat berjalan dengan baik. Dalam proses ini, digunakan Visual Studio Code sebagai lingkungan pengembangan utama. Pada sisi front end, pemrograman dilakukan menggunakan HTML dan CSS untuk membuat antarmuka pengguna yang interaktif dan mudah digunakan. Sedangkan untuk sisi back end, digunakan Python dengan framework Flask untuk menangani logika aplikasi dan interaksi dengan model machine learning. Setelah pengembangan selesai, aplikasi dapat dideploy ke Heroku untuk diakses oleh pengguna.

3.5.4 Algoritma Decision Tree

Dalam penelitian ini, metode yang diusulkan untuk deteksi phishing adalah menggunakan decision tree. Algoritma Decision tree menjadi metodw klasifikasi yang efektif dan mudah diimplementasikan. Dalam penelitian ini, akan digunakan sebuah paket atau pustaka perangkat lunak yang telah dikembangkan untuk menerapkan algoritma decision tree. Paket ini akan memudahkan implementasi algoritma decision tree dalam pemrosesan data dan analisis yang akan dilakukan. Dengan menggunakan paket ini, peneliti dapat dengan cepat membuat, melatih, dan mengevaluasi model decision tree untuk kebutuhan deteksi phishing berbasis URL. Paket yang akan digunakan dalam penelitian ini adalah scikit-learn dengan menggunakan kelas DecisionTreeClassifier. Gambar 3.10 Diagram Alur Algoritma *Decision Tree*

akan memperjelas langkah-langkah dalam pengolahan dataset hingga prediksi menggunakan model decision tree.



Gambar 3.11 Diagram Alur Algoritma *Decision Tree*

Alur pengolahan data dimulai dari dataset yang kemudian melalui tahap analisis data eksploratif (EDA) untuk memahami karakteristik dataset, dilanjutkan dengan preprocessing data yang mencakup pembersihan data, kemudian dilatih dengan decision tree model. Setelah dilatih, model digunakan untuk memprediksi keberadaan URL phishing berdasarkan fitur-fitur yang ada. Setelah itu, pengguna memberikan input berupa keyword yang dijadikan sebagai dasar untuk memprediksi keberadaan URL phishing.

A. Dataset

Penelitian ini memanfaatkan dataset yang diperoleh dari repositori public yang tersedia di website kaggle untuk analisis deteksi phishing. Dataset digunakan sebagai sumber data untuk melatih dan menguji model deteksi phishing yang akan dikembangkan. Dataset tersebut dibuat dari kumpulan

data yang berasal dari UCI yang dibuat oleh David Aha dan mahasiswa pascasarjana UB Irvine. UCI merupakan website arsip koleksi dataset yang dibuat sejak tahun 1987. Dataset akan digunakan sebagai dasar untuk melakukan analisis, preprocessing, serta evaluasi kinerja model deteksi phishing menggunakan decision tree. Bentuk file yang disediakan adalah .csv.

B. Exploratory Data Analisys (EDA)

Pada tahap Exploratory Data Analisys (EDA) bertujuan untuk memahami karakteristik pada dataset, seperti pola, hubungan variable sebelum melakukan preprocessing. Dalam dataset ini, akan dilihat kelas antara URL phishing dan non-phishing, serta melihat frekuensi fitur-fitur tertentu seperti panjang URL, jumlah tanda baca, dan lainnya.

C. Preprocessing

Pada tahap preprocessing, data dari dataset diolah untuk mempersiapkannya sebelum digunakan dalam pembuatan model deteksi phishing. Langkah-langkah preprocessing meliputi pembersihan data dari nilai-nilai yang hilang atau tidak valid, pengkodean ulang fitur kategorikal menjadi format yang dapat diproses oleh model, dan pemilihan fitur yang relevan. Preprocessing bertujuan untuk meningkatkan kualitas data dan memastikan bahwa model yang dibangun berdasarkan data yang telah disiapkan dengan baik.

D. Training Model

Pada tahap Training Model, dataset akan dibagi menjadi data latih dan data uji dengan perbandingan 70:30, di mana 70% data digunakan untuk melatih model dan 30% digunakan untuk menguji performa model. Pembagian data ini dilakukan untuk memastikan bahwa model dapat belajar dari data yang cukup dan diuji pada data yang belum pernah dilihat sebelumnya. Dengan menggunakan data latih, model akan belajar pola-pola yang ada dalam data, sedangkan data uji digunakan untuk menguji seberapa baik model dapat melakukan prediksi pada data baru. Dengan pembagian data ini, dapat diukur kinerja model secara objektif sebelum mengimplementasikannya dalam aplikasi deteksi phishing.

E. Prediction Using Model

Pada tahap Prediction Using Model, model yang telah dilatih akan digunakan untuk melakukan prediksi terhadap data baru. Data baru yang ingin diprediksi dapat berupa URL yang dimasukkan. Model akan mengambil fitur-fitur dari URL tersebut, kemudian melakukan prediksi apakah URL tersebut terindikasi sebagai phishing atau tidak. Prediksi dilakukan berdasarkan pembelajaran pola yang telah diperoleh. Hasil prediksi kemudian dapat disajikan kembali kepada pengguna sebagai informasi apakah URL tersebut aman atau berpotensi phishing.

3.5.5 Sistem Pengujian

Pengujian pada penelitian ini menggunakan metode pengujian blackbox di mana penguji mengevaluasi fungsi aplikasi tanpa melihat struktur internal atau kode sumber. Fokus utama adalah pada input yang diberikan ke aplikasi dan output yang dihasilkan, untuk memastikan bahwa aplikasi berfungsi sesuai dengan spesifikasi yang ditentukan. Berikut adalah langkah-langkah dan aspek yang diuji dalam pengujian blackbox untuk aplikasi deteksi phishing berbasis web dapat dilihat pada table 3.5 Sistem Pengujian Aplikasi.

Tabel 3.5 Sistem Pengujian Aplikasi

No.	Pengujian	Tujuan	Langkah Pengujian	Hasil yang diharapkan
1.	Home Page	Memastikan halaman menampilkan ringkasan layanan yang ditawarkan.	a) Akses halaman utama aplikasi. b) Verifikasi tampilan dan konten halaman, seperti deskripsi layanan, dan navigasi	Halaman memuat dengan benar, menampilkan semua informasi yang relevan dan navigasi berfungsi.
2.	Searching Page	Memastikan halaman input URL berfungsi dengan benar.	a) Akses halaman input URL. b) Masukkan URL yang ingin dideteksi dan kirimkan. c) Verifikasi bahwa input diterima dan diproses oleh sistem.	Halaman harus menerima input URL, memprosesnya, dan mengarahkan pengguna ke halaman hasil.
3.	Result Page	Memastikan hasil deteksi URL ditampilkan dengan benar.	a) Setelah memasukkan URL di halaman input, periksa halaman hasil.	Halaman harus menampilkan prediksi model dan menyediakan informasi yang mudah dipahami

No.	Pengujian	Tujuan	Langkah Pengujian	Hasil yang diharapkan
			b) Verifikasi bahwa hasil deteksi (phishing atau tidak phishing) ditampilkan dengan jelas. c) Periksa konsistensi hasil dengan input yang diberikan. d) Periksa navigasi ke halaman history	pengguna dan navigasi berfungsi.
4.	History Page	Memastikan tools search dan history prediksi pengguna ditampilkan dengan benar.	a) Akses halaman histori. b) Periksa tools search	Halaman harus menampilkan histori deteksi para pengguna dan tools search dapat digunakan dengan semestinya.
5.	Error Handling	Memastikan aplikasi menangani kesalahan dengan benar.	a) Masukkan URL yang tidak valid atau kosong dan kirimkan. b) Verifikasi bahwa sistem memberikan pesan kesalahan yang informatif.	Halaman harus menampilkan pesan kesalahan yang sesuai.

Pengujian blackbox pada aplikasi deteksi phishing berbasis web mencakup pengujian fungsionalitas utama seperti halaman utama, halaman input URL, dan halaman hasil, serta penanganan kesalahan. Dengan memastikan setiap bagian aplikasi berfungsi sesuai spesifikasi tanpa melihat struktur internal, pengujian blackbox membantu memastikan bahwa aplikasi siap digunakan oleh pengguna.

3.5.6 Deployment Sistem

Aplikasi Flask yang telah dibangun kemudian di-deploy ke Heroku. Heroku memungkinkan aplikasi berjalan secara terus-menerus di web, tanpa server lokal atau komputer pribadi. Proses deployment melibatkan beberapa Langkah. Pertama, aplikasi Flask dikonfigurasi agar dapat berjalan di lingkungan Heroku. Kemudian, aplikasi dan dependensinya dikemas dalam repositori Git, yang di-push ke Heroku menggunakan Heroku CLI. Setelah proses deployment selesai, aplikasi Flask akan aktif dan dapat diakses melalui URL yang disediakan oleh Heroku, memastikan aplikasi tersedia secara online dan dapat diakses oleh pengguna kapan saja tanpa harus dijalankan ulang dari komputer lokal.

BAB IV

HASIL YANG DIHARAPKAN

4.1 Luaran Yang Diharapkan

Dalam penelitian ini, diharapkan beberapa hasil yang dapat memberikan kontribusi dalam bidang Perancangan aplikasi deteksi *phishing* berbasis web menggunakan algoritma *decision tree* pada komunitas Panglima di Pondok Pesantren Sunan Plumbon. Luaran yang diharapkan termasuk:

a) Aplikasi Deteksi Phishing Berbasis Web:

   Diharapkan dapat diakses dengan mudah oleh anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon.

b) Peningkatan Kesadaran Keamanan Digital:

   Diharap penelitian ini dapat meningkatkan kesadaran dan pengetahuan anggota komunitas Panglima di Pondok Pesantren Sunan Plumbon tentang ancaman Phishing dan cara mengidentifikasinya.

c) Dokumentasi dan Lapoan Penelitian:

   Penyusunan dokumentasi lengkap perancangan aplikasi.

d) Publikasi Ilmiah:

   Diharap hasil penelitian ini dapat dipubliksikan dalam jurnal bereputasi yang terindeks di SINTA sebagai bentuk penyebaran ilmiah dan kontribusi pada perkembangan pengetahuan di bidang industri keamanan siber dengan menyediakan pendekatan deteksi yang efektif.

Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi yang berarti dalam meningkatkan keamanan siber dan perlindungan pengguna daring dari ancaman phishing.

4.2 Jadwal Penelitian

Berikut adalah jadwal penelitian untuk proyek Perancangan aplikasi deteksi *phishing* berbasis web menggunakan algoritma *decision tree* pada komunitas Panglima di Pondok Pesantren Sunan Plumbon.

Tabel 4.1 Jadwal Penelitian

No.	Kegiatan	Pelaksanaan																							
		Februari				Maret				April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1.	Perancangan Proposal																								
2.	Seminar Proposal																								
3.	Pengambilan Data																								
4.	Pelaksanaan Penelitian																								
5.	Penyelesaian Laporan																								
6.	Seminar Hasil																								
7.	Publikasi Jurnal																								

DAFTAR PUSTAKA

[1] H. Fatiya Rumi, "Pemanfaatan Internet oleh Pelaku Usaha Digital," *KataData*, 2022. https://databoks.katadata.co.id/datapublish/2022/07/14/pemanfaatan-internet-oleh-pelaku-usaha-digital

[2] T. Rochmadi, "Live Forensik Untuk Analisa Anti Forensik Pada Web Browser Studi Kasus Browzar," *Indones. J. Bus. Intell.*, vol. 1, no. 1, p. 32, 2019, doi: 10.21927/ijubi.v1i1.878.

[3] T. Rochmadi and I. Y. Pasa, "Measurement of Risk and Evaluation of Information Security Using The Information Security Index in BKD XYZ Based on ISO 27001/SNI," *CyberSecurity dan Forensik Digit.*, vol. 4, no. 1, pp. 38–43, 2021, doi: 10.14421/csecurity.2021.4.1.2439.

[4] M. H. Wibowo and N. Fatimah, "Ancaman Phishing Terhadap Pengguna Sosial Media Dalam Dunia Cyber Crime," *JOEICT(Jurnal Educ. Inf. Commun. Technol.*, vol. 1, no. 1, pp. 1–5, 2017, [Online]. Available: https://www.jurnal.stkippgritulungagung.ac.id/index.php/joeict/article/view/69

[5] R. D. Hapsari, K. G. Pambayun, and A. Cybercrime, "ANCAMAN CYBERCRIME DI INDONESIA Sebuah Tinjauan Pustaka Sistematis," vol. 5, no. April, pp. 1–17, 2023, [Online]. Available: https://ejournal.ipdn.ac.id/konstituen/article/view/3208/1657

[6] C. M. Annur, "Serangan Malware Banyak Mengintai UMKM di Masa

Pandemi," *KataData*, 2021. https://databoks.katadata.co.id/datapublish/2021/10/23/serangan-malware-banyak-mengintai-umkm-di-masa-pandemi

[7] G. D. Setyawan, A. Yuswanto, A. M. Ridwan, and B. Wibowo, "Implementasi Metode Adasyn Dalam Deteksi Url Berbahaya Menggunakan Machine Learning: Demi Meningkatkan Keamanan Siber Di Era Digital," vol. 6, no. 2, pp. 123–126, 2023, doi: 10.31943/teknokom.v6i2.153.

[8] 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.

[9] N. B. Putri and A. W. Wijayanto, "Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi Website Phishing," *Komputika J. Sist. Komput.*, vol. 11, no. 1, pp. 59–66, 2022, doi: 10.34010/komputika.v11i1.4350.

[10] M. Fandru, A. Rifqi, M. Dina, M. N. K. Nababan, and S. Aisyah, "Comparative Analysis of Phishing Website Prediction Classification Algorithm Using Logistic Regression , Decision Tree , and Random Forest," *Infor.Seaninstitute.Org*, vol. 10, no. 2, pp. 859–869, 2022, [Online]. Available: http://infor.seaninstitute.org/index.php/infokum/article/view/425

[11] A. S. Sunge, "Komparasi Machine Learning Memprediksi Phising Dalam Keamanan Website," *Pros. SAINTEK Sains dan Teknol.*, vol. 1, no. 1, pp. 135–140, 2022.

[12] A. S. Y. Irawan, N. Heryana, H. S. Hopipah, and D. Rahma, "Identifikasi Website Phishing dengan Perbandingan Algoritma Klasifikasi," *Syntax J. Inform.*, vol. 10, no. 01, pp. 57–67, 2021, doi: 10.35706/syji.v10i01.5292.

[13] T. Rochmadi, "Deteksi Bukti Digital Pada Adrive Cloud Storage Menggunakan Live Forensik," *Cyber Secur. dan Forensik Digit.*, vol. 2, no. 2, pp. 65–68, 2019, doi: 10.14421/csecurity.2019.2.2.1455.

[14] V. F. P. Y, "Modus Operandi Tindak Pidana Phishing Menurut UU ITE," vol. 4, no. 6, pp. 2525–2548, 2021, doi: 10.20473/jd.v4i6.31857.

[15] Q. Abu Al-Haija and M. Al-Fayoumi, "An intelligent identification and classification system for malicious uniform resource locators (URLs)," *Neural Comput. Appl.*, vol. 35, no. 23, pp. 16995–17011, 2023, doi: 10.1007/s00521-023-08592-z.

[16] K. Immanuel Rui Costa, "Pengembangan dan Pembuatan Website: Sebuah Tinjauan Literatur," *ResearchGate*, no. April, pp. 1–4, 2022.

[17] Karambir and S. Thind, "A Simulation Model for the Spiral Software Development Life Cycle," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 03, no. 05, pp. 3823–3830, Jun. 2015, doi: 10.15680/ijircce.2015.0305013.

[18] A. Rahman *et al.*, *Metode Penelitian Ilmu Sosial*, no. October. CV.Widina Media Utama, 2022. [Online]. Available: https://www.asikbelajar.com/metode-pengumpulan-data-sekunder/

[19] L. N. Rachmadi, A. P. Wibawa, and U. Pujianto, "Rekomendasi Jurusan Dengan Menggunakan Decision Tree Pada Sistem Penerimaan Peserta Didik Baru SMK Widya Dharma Turen," *Belantika Pendidik.*, vol. 4, no. 1, pp. 29–

36, 2021, doi: 10.47213/bp.v4i1.95.

[20] A. SETIYAD and T. Harihayati, "Penerapan Sqlite Pada Aplikasi Pengaturan Waktu Ujian Dan Presentasi," *Maj. Ilm. UNIKOM*, vol. 13, no. 2, 2015, doi: 10.34010/miu.v13i2.128.

[21] M. Romzi and B. Kurniawan, "Implementasi Pemrograman Python Menggunakan Visual Studio Code," *J. Inform. Dan Komput.*, vol. 11, no. 2, pp. 1–9, 2020, [Online]. Available: www.python.org

[22] Wahyudi, *Pemrograman WEB : HTML dan CSS*, vol. 6, no. 1. CV. Eureka Media Aksara, 2022. [Online]. Available: http://repositorio.unan.edu.ni/2986/1/5624.pdf%0Ahttp://fiskal.kemenkeu.go.id/ejournal%0Ahttp://dx.doi.org/10.1016/j.cirp.2016.06.001%0Ahttp://dx.doi.org/10.1016/j.powtec.2016.12.055%0Ahttps://doi.org/10.1016/j.ijfatigue.2019.02.006%0Ahttps://doi.org/10.1

[23] J. Hermawan, *Analisa Desain dan Pemrograman Berorientasi Objek dengan UML dan Visual Basic . Net*. Andi Offset, 2005.

[24] S. D. Redemptus and M. Muhlis, "Sistem Informasi Penjualan Berbagai Macam Produk Berbasis Android Di Toko De Ari Palopo," *Indones. J. Educ. Humanit.*, vol. 2, no. 1, pp. 34–50, 2022.

[25] M. Grinberg, *Flask Web Development*, vol. 2. 2014.

[26] R. T. Handayanto and H. Herlawati, "Prediksi Kelas Jamak dengan Deep Learning Berbasis Graphics Processing Units," *J. Kaji. Ilm.*, vol. 20, no. 1, pp. 67–76, 2020, doi: 10.31599/jki.v20i1.71.

[27] A. Verma, A. Khatana, and S. Chaudhary, "A Comparative Study of Black

Box Testing and White Box Testing," *Int. J. Comput. Sci. Eng.*, vol. 5, no. 12, pp. 301–304, 2017, doi: 10.26438/ijcse/v5i12.301304.

[28] C. N. Sidauruk, A. Purnama, T. Zani, F. Ilmu, and T. Universitas, "Pembangunan Aplikasi Augmented Reality Dan Implementasi Video Alat Musik Tradisional Jawa Barat," *e-Proceeding Appl. Sci.*, vol. 6, no. 2, pp. 4161–4170, 2020.

Lampiran 1. Formulir Bimbingan Proposal dan Skripsi

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA YOGYAKARTA

Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183

Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Diskusi terkait judul yang diajukan.
2. Diskusi metode penelitian.

(Judul yang diajukan : Analisis dan Deteksi Platform Penipuan Berkedok Investasi di Lingkungan Judi Online Menggunakan Teknologi Data Mining: Pengembangan deteksi)

Pembahasan Pembimbing:

Berikan jurnal pendukung terkait judul dan metode yang diajukan.

Tanggal	Ttd mahasiswa	Ttd pembimbing
25 September 2023	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA YOGYAKARTA

Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183

Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Diskusi terkait metode pengambilan data

Pembahasan Pembimbing:

Cari datasetnya dari penelitian terdahulu agar lebih mudah dan aman

Tanggal	Ttd mahasiswa	Ttd pembimbing
26 September 2023	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

FAKULTAS KOMPUTER DAN TEKNIK
UNIVERSITAS ALMA ATA YOGYAKARTA
Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183
Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Belum menemukan data terkait objek (data aplikasi yang terindikasi investasi palsu)
2. Mengajukan metode pengambilan data

Pembahasan Pembimbing:

Ganti topik penelitian.

Tanggal	Ttd mahasiswa	Ttd pembimbing
3 Oktober 2023	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

FAKULTAS KOMPUTER DAN TEKNIK
UNIVERSITAS ALMA ATA YOGYAKARTA
Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183
Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Mengajukan judul "Perancangan Sistem Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web"
2. Pengajuan progres BAB 1: Dimulai dari latar belakang hingga signifikasi penelitian.
3. Progres pengerjaan BAB 2. Dimulai dari penelitian terkait hingga kajian teori.

Pembahasan Pembimbing:

1. Perbaiki beberapa kalimat pada paragraf di latar belakang dan kajian teori : Buat kalimatnya lebih beralur dengan kalimat sebelumnya.
2. Perbaiki tabel penelitian : Kolom header pada tabel dibuat repeat header dan diurutkan dari tahun terendah ke tahun terbaru.

Tanggal	**Ttd mahasiswa**	**Ttd pembimbing**
14 Maret 2024	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA YOGYAKARTA

Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183

Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Progres pengerjaan BAB 2. Dimulai dari penelitian terkait hingga kajian teori.

Pembahasan Pembimbing:

1. Perbaiki tabel penelitian : Kolom header pada tabel dibuat repeat header dan diurutkan dari tahun terendah ke tahun terbaru.

Tanggal	**Ttd mahasiswa**	**Ttd pembimbing**
14 Maret 2024	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA YOGYAKARTA

Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183

Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Pengajuan revisi: Kalimat sudah dibuat lebih beralur dengan kalimat sebelumnya.
2. Pengajuan revisi: Kolom header pada tabel penelitian sudah dibuat repeat header dan diurutkan dari tahun terendah ke tahun terbaru.
3. Progres pengerjaan BAB 2: Dimulai dari penelitian terkait sampai kerangka pemikiran.

Pembahasan Pembimbing:

1. Perbaiki beberapa tabel: Nama pada tabel jangan terpotong.

Tanggal	**Ttd mahasiswa**	**Ttd pembimbing**
15 Maret 2024	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA YOGYAKARTA

Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183

Telp. (0274) 4342288

Formulir Bimbingan Proposal Dan Skripsi

Nama	:	Kurnia Ainun Najah
NIM	:	203100088
Prodi	:	Sistem Informasi
Pembimbing	:	Tri Rochmadi, S.Kom., M.Kom.
Judul	:	Perancangan Deteksi Phishing Menggunakan Algoritma LSTM-CNN Berbasis Web

Pembahasan Mahasiswa:

1. Progres pengerjaan eksekutif summary.
2. Progres pengerjaan BAB 3. Dimulai dari desain penelitian hingga tahap desain sistem.
3. Progres pengerjaan BAB 4. Hasil yang diharapkan.

Pembahasan Pembimbing:

1. Perbaiki beberapa kalimat pada paragraf di eksekutif summary : Buat kalimatnya lebih mudah dipahami.
2. Perbaiki semua kesalahan penulisan (typo).
3. Cek label pada tabel dan gambar. Urutkan penomorannya.

Tanggal	**Ttd mahasiswa**	**Ttd pembimbing**
19 Maret 2024	Kurnia Ainun Najah	Tri Rochmadi, S.Kom., M.Kom.

Lampiran 2. Formulir Mahasiswa Mengikuti Seminar Proposal

FAKULTAS KOMPUTER
UNIVERSITAS ALMA ATA YOGYAKARTA
Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183
Telp. (0274) 4342288

PRESENSI MAHASISWA MENGIKUTI SEMINAR PROPOSAL

Nama : KURNIA AINUN NAJAH

NIM : 203100088

Prodi : SISTEM INFORMASI

NO	NAMA MAHASISWA TERUJI	JUDUL SKRIPSI/KTI	TTD KETUA DEWAN PENGUJI
1.	Heri Gunawan Yusuf	Pengaruh Push Pull Mooring sebagai Faktor Switching Intention Pada Pengguna Aplikasi Video On Demand Pada Masa Pandemi COVID-19	Asti Rafnasari, S.Kom., M.Kom.
2.	Siti Nurhasanah	Evaluasi Tingkat Kesiapan Pengguna Sistem Single Sign On Pada Portal Universitas Alma Ata Menggunakan Metode Technology Readiness Index (Tri)	Ayu Ilaila A. Harahap, M.Kom
3.	Yahya Mukhaemin	Penerapan Metode Design Thinking Pada Model Perancangan Ulang UI/UX Aplikasi Alma Ata Mobile	Asti R
4.	Elpa Siti Nora Nasution	Sistem Pendukung keputusan Pemilihan Aplikasi Layanan E-Ticketing Berbasis Website Menggunakan Metode SAW	
5.	Nur Aisah Amini	Analisis Tingkat Kepuasan Konsumen Menggunakan Metode Importance Performance Analysis pada Aplikasi Tiktok Shop	

FAKULTAS KOMPUTER DAN TEKNIK

UNIVERSITAS ALMA ATA YOGYAKARTA

Jl. Brawijaya No.99, Jalan, Tamantirto, Kec. Kasihan, Bantul, Daerah IstimewaYogyakarta 55183

Telp. (0274) 4342288

PRESENSI MAHASISWA MENGIKUTI SEMINAR PROPOSAL

Nama : **KURNIA AINUN NAJAH**

NIM : **203100088**

Prodi : **SISTEM INFORMASI**

NO	NAMA MAHASISWA TERUJI	JUDUL SKRIPSI/KTI	TTD KETUA DEWAN PENGUJI
6.	Imam Hanbali	Perancangan Sistem Informasi Akademik Berbasis Website Pada Sekolah Menengah Pertama Al-Ghuroba'	
7.	Cucu Sri Nopianti	Segmentasi Potensi Daerah Wisata Di Kabupaten Tasikmalaya Menggunakan K-Means Clustering	
8	M. Ikhsanul Ali	Perancangan Sistem Informasi Website Di SMK Negeri Tapango Menggunakan Metode Rapid Application Development (RAD)	

Yogyakarta,

Mahasiswa

KURNIA AINUN NAJAH

NIM. 203100088

Dosen Pembimbing Akademik

YANUAR WICAKSONO, S.Kom.,M.Kom

NIK. 1214020380