

SKRIPSI

**ANALISIS PERFORMA *WEBSITE* DENGAN
MENGUNAKAN *FORECAST INSTANCE LOAD BALANCER*
SKALA *ENTERPRISE* DENGAN MODEL *MULTI LINEAR*
REGRESSION DAN *XGBOOST REGRESSION***

Disusun dan diajukan oleh:

**NUR ISLAMIAH RIFAI
D121 20 1006**



**PROGRAM STUDI SARJANA TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
GOWA
2024**

LEMBAR PENGESAHAN SKRIPSI

**ANALISIS PERFORMA *WEBSITE* DENGAN MENGGUNAKAN
FORECAST INSTANCE LOAD BALANCER SKALA *ENTERPRISE*
DENGAN MODEL *MULTI LINEAR REGRESSION* DAN *XGBOOST*
*REGRESSION***

Disusun dan diajukan oleh

**NUR ISLAMIAH RIFAI
D121 20 1006**

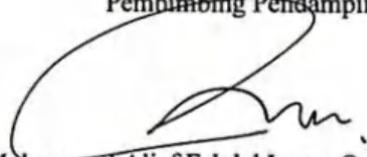
Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian
Studi Program Sarjana Program Studi Teknik Informatika
Fakultas Teknik Universitas Hasanuddin
Pada tanggal 19 Juli 2024
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,


Dr. Eng. Ir. Zulkifli Tahir, S.T., M.Sc.
NIP 19840403 201012 1 004


Ir. Muhammad Alief Fahdal Imran Oemar, S.T., M.Sc.
NIP 19940522 202204 3 001


Ketua Program Studi,
Prof. Dr. Ir. Inurabaya, S.T., MT., M.Bus.Sys., IPM, ASEAN. Eng.
NIP 19750716 200212 1 004



PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini ;
Nama : Nur Islamiah Rifai
NIM : D121201006
Program Studi : Teknik Informatika
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

“Analisis Performa Website dengan Menggunakan Forecast Instance Load Balancer Skala Enterprise dengan Model Multi Linear Regression dan XGBoost Regression”

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Semua informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala resiko.

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh Penulis di masa depan harus mendapat persetujuan dari Dosen Pembimbing.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Gowa, 15 Agustus 2024

Yang Menyatakan



Nur islamiah Rifai



ABSTRAK

NUR ISLAMIAH RIFAI. ANALISIS PERFORMA WEBSITE DENGAN MENGGUNAKAN FORECAST INSTANCE LOAD BALANCER SKALA ENTERPRISE DENGAN MODEL MULTI LINEAR REGRESSION DAN XGBOOST REGRESSION (dibimbing oleh Dr. Eng Zulkifli Tahir, S.T., M.SC. dan Ir. Muhammad Alief Fahdal Imran Oemar, S.T., M.Sc.)

Kemajuan teknologi telah membuat situs web menjadi pusat dari hampir setiap institusi, terutama dengan perkembangan *search engine* yang semakin pintar. Semakin banyak organisasi yang memberikan informasi melalui situs web mereka, semakin banyak pula pengunjung yang mengaksesnya. *Website* tidak hanya berfungsi sebagai media informasi, tetapi juga sebagai indikator kualitas suatu organisasi. Dalam lingkungan berskala *enterprise*, mengelola kinerja dan *fault tolerance website* menjadi tantangan yang kompleks. Oleh karena itu, organisasi semakin tertarik untuk menemukan solusi yang efektif dalam mengoptimalkan kinerja dan ketahanan situs web mereka. Salah satu pendekatan inovatif adalah menggunakan teknik prakiraan beban, untuk meramalkan beban permintaan masa depan dan mengalokasikan sumber daya secara efektif.

Penelitian ini bertujuan untuk menganalisis pengaruh penggunaan *Forecast Instance Load Balancer* pada performa situs web di lingkungan skala *enterprise*, dengan fokus pada peningkatan efisiensi dan penyebaran beban kerja yang merata di antara *server-instance*. Selain itu, penelitian ini juga bertujuan untuk meningkatkan tingkat *fault tolerance website* dengan menerapkan model *Multi Linear Regression* dan *XGBoost Regression* untuk meramalkan tren dan pola kesalahan yang mungkin terjadi, sehingga memungkinkan adopsi langkah-langkah pencegahan atau pemulihan yang lebih cepat.

Metode penelitian melibatkan penggunaan model *Multi Linear Regression* (MLR) dan *XGBoost Regression* untuk melakukan forecasting. Sistem ini diintegrasikan dengan *Amazon Web Service* (AWS) untuk optimalitas kinerja. Proses validasi dilakukan dengan menggunakan metrik evaluasi seperti R-squared, MAE, MSE, RMSE, dan MAPE.

Hasil penelitian menunjukkan bahwa model *Multi Linear Regression* menghasilkan nilai R-squared sebesar 99.98%, dengan MAE bernilai 8.19, MSE bernilai 107.43, RMSE bernilai 10.36 dan MAPE 0.28%. Sementara itu, model *XGBoost Regression* menghasilkan *R-squared* sebesar 98.12%, dengan MAE bernilai 34.18, MSE bernilai 2141.37, RMSE bernilai 46.27, dan MAPE 1.34%. Integrasi dengan layanan AWS menunjukkan bahwa simulasi selama 8 hari dengan rata-rata jumlah request antara 2.000 hingga 5.000 data dan penggunaan 2 hingga 4 *server instance* dapat menjaga kinerja yang stabil dan responsif. Proses simulasi juga membantu mengidentifikasi waktu puncak dan kapasitas server yang diperlukan untuk menjaga kinerja situs web.



ci: Performa Website, Forecast Instance, Load Balancer, Enterprise, Time

ABSTRACT

NUR ISLAMIAH RIFAI. *WEBSITE PERFORMANCE ANALYSIS USING ENTERPRISE SCALE FORECAST INSTANCE LOAD BALANCER WITH MULTI LINEAR REGRESSION DAN XGBOOST REGRESSION MODEL* (supervised by Dr. Eng Zulkifli Tahir, S.T., M.SC. and Ir. Muhammad Alief Fahdal Imran Oemar, S.T., M.Sc.)

Technological advancements have made websites the center of almost every institution, especially with the development of increasingly smart search engines. The more organizations that provide information through their websites, the more visitors access them. The website not only serves as a medium of information, but also as an indicator of the quality of an organization. In an enterprise-scale environment, managing website performance and fault tolerance becomes a complex challenge. Therefore, organizations are increasingly interested in finding effective solutions in optimizing the performance and resilience of their websites. One innovative approach is to use load forecasting techniques, to forecast future demand loads and allocate resources effectively.

This study aims to analyze the effect of using Forecast Instance Load Balancer on website performance in enterprise-scale environments, focusing on improving efficiency and evenly distributing workloads among server-instances. In addition, the study also aims to improve the fault tolerance rate of websites by applying time series regression models to forecast possible error trends and patterns, thus enabling faster adoption of prevention or recovery measures.

The research method involves the use of Multiple Linear Regression (MLR) and XGBoost Regression models to perform forecasting. The system is integrated with Amazon Web Services (AWS) for performance optimization. The validation process is done using evaluation metrics such as R-squared, MAE, MSE, RMSE, and MAPE.

The results showed that the Multi Linear Regression model produced an R-squared value of 99.98%, with MAE worth 8.19, MSE worth 107.43, RMSE worth 10.36, and MAPE 0.28%. Meanwhile, the XGBoost Regression model yields an R-squared of 98.12%, with MAE worth 34.18, MSE worth 2141.37, RMSE worth 46.27, and MAPE 1.34%. Integration with AWS services shows that an 8-day simulation with an average number of requests between 2.000 to 5.000 records and usage of 2 to 4 server instances can maintain stable and responsive performance. The simulation process also helps identify peak times and server capacity required to maintain website performance.

Keywords: Performance Website, Forecast Instance, Load Balancer, Enterprise, Time Series



DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI	i
PERNYATAAN KEASLIAN.....	ii
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	viii
DAFTAR SINGKATAN DAN ARTI SIMBOL	ix
DAFTAR LAMPIRAN.....	x
KATA PENGANTAR	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian/Perancangan.....	2
1.4 Manfaat Penelitian/Perancangan.....	2
1.5 Ruang Lingkup/Asumsi perancangan	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 <i>Website</i>	4
2.2 <i>Forecasting</i>	4
2.3 <i>Web Server</i>	5
2.4 <i>Instance Load Balancer</i>	6
2.5 <i>Amazon Web Service (AWS)</i>	7
2.5.1 <i>Amazon Elastic Compute Cloud (Amazon EC2)</i>	8
2.5.2 <i>Load Balancer dan Target Groups</i>	9
2.5.3 <i>Auto Scaling Group (ASG)</i>	9
2.5.4 <i>Virtual Private Cloud (VPC)</i>	10
2.6 <i>Enterprise</i>	10
2.7 <i>Regresi Time Series</i>	11
2.7.1 <i>Multiple Linear Regression</i>	13
2.7.2 <i>Extreme Gradient Boosting Regression</i>	15
BAB 3 METODE PENELITIAN/PERANCANGAN	17
3.1 Lokasi Penelitian.....	17
3.2 Tahapan Penelitian.....	17
3.3 Instrumen Penelitian	20
3.4 Perancangan Sistem	20
3.4.1 Rancangan Simulator	21
3.5 Implementasi Sistem	22
BAB 4. HASIL DAN PEMBAHASAN.....	35
4.1 Hasil Analisis <i>Trend</i> dan Pola Kinerja <i>Website</i>	35
4.2 Hasil <i>Forecasting</i>	38
4.2.1 <i>Forecasting Menggunakan Multi Linear Regression Model</i>	38
4.2.2 <i>Forecasting Menggunakan XGBoost Regression Model</i>	41
4.2.3 Integrasi dengan Layanan AWS untuk Optimalisasi Kinerja.....	43
4.3 KESIMPULAN DAN SARAN	46
4.3.1 Kesimpulan	46



5.2 Saran..... 46
DAFTAR PUSTAKA 47



DAFTAR GAMBAR

Gambar 1 Ilustrasi <i>Forecasting</i>	5
Gambar 2 Ilustrasi cara kerja web server.....	6
Gambar 3 Pola Data Konstan.....	12
Gambar 4 Pola Data Musiman.....	12
Gambar 5 Pola Data <i>Trend</i>	12
Gambar 6 Pola Data Siklik.....	13
Gambar 7 Representasi Konsep <i>XGBoost</i> Model.....	15
Gambar 8 Lab Ubicon Gedung Elektro.....	17
Gambar 9 Tahapan Penelitian.....	17
Gambar 10 <i>Flowchart</i> Penelitian.....	19
Gambar 11 Arsitektur Sistem <i>Auto Scaling</i> pada AWS.....	21
Gambar 12 Export <i>Dataset</i> ke Jupyter Notebook.....	24
Gambar 13 Variabel dalam <i>Dataset</i>	24
Gambar 14 <i>Flow Diagram</i> Sistem <i>Forecasting</i>	25
Gambar 15 Pembuatan <i>auto scaling group</i>	29
Gambar 16 Pembuatan <i>Instance EC2 Template</i>	30
Gambar 17 Otomatisasi Konfigurasi <i>Instance</i>	31
Gambar 18 <i>Scaling by Request</i>	33
Gambar 19 Jumlah Pengunjung Setiap Hari di Tahun 2020.....	36
Gambar 20 Dekomposisi Musiman Aditif.....	36
Gambar 21 <i>Autocorrelation Plot</i>	37
Gambar 22 <i>Partial Autocorrelation Plot</i>	38
Gambar 23 Hasil <i>Forecasting</i> Menggunakan <i>Multi Linear Regression</i> Model... 39	39
Gambar 24 Hasil <i>Forecasting</i> Menggunakan <i>XGBoost Regression</i> Model..... 42	42



DAFTAR TABEL

Tabel 1 Dataset.....	22
Tabel 2 Nilai Hasil <i>Forecasting</i> Menggunakan MLR Model.....	40
Tabel 3 Nilai Hasil <i>Forecasting</i> Menggunakan XGBR Model	42
Tabel 4 Hasil Simulasi <i>Website</i> menggunakan layanan AWS.....	43



DAFTAR SINGKATAN DAN ARTI SIMBOL

Lambang/Singkatan	Arti dan Keterangan
Y	Variabel Dependen
β_0	Konstanta
$\beta_1, \beta_2, \dots, \beta_n$	Koefisien Regresi Yang Mengukur Pengaruh Variabel Independen Terhadap Variabel Dependen
X_1, X_2, \dots, X_n	Variabel Independen
ε	Kesalahan Acak
MLR	<i>Multi Linear Regression</i>
XGBoost	<i>Extreme Gradient Boosting</i>
MAE	<i>Mean Absolute Error</i>
MSE	<i>Mean Squared Error</i>
RMSE	<i>Root Mean Squared Error</i>
MAPE	<i>Mean Absolute Percentage Error</i>
ACF	<i>Autocorrelation</i>
PACF	<i>Partial Autocorrelation</i>
GMM	<i>Gaussian Mixture Model</i>



DAFTAR LAMPIRAN

Lampiran 1 <i>Source Code Import Data</i>	49
Lampiran 2 Filter Data Tahun 2020	50
Lampiran 3 Dekomposisi Musiman Aditif untuk Data Kunjungan <i>Website</i>	50
Lampiran 4 Analisis Autokorelasi dan Autokorelasi Parsial untuk Prediksi Kunjungan <i>Website</i>	51
Lampiran 5 Prediksi Jumlah Kunjungan <i>Website</i> Menggunakan <i>Multi Linear</i> <i>Regression Model</i>	52
Lampiran 6 Prediksi Jumlah Kunjungan <i>Website</i> Menggunakan <i>XGBoost</i> <i>Regression Model</i>	53
Lampiran 7 Evaluasi Kinerja <i>Multi Linear Regression Model</i>	54
Lampiran 8 Evaluasi Kinerja <i>XGBoost Model</i>	54
Lampiran 9 <i>Schedule Automatic Jobs (Simulation)</i>	55
Lampiran 10 Lembar Perbaikan Skripsi	59



KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh.

Segala puji dan Syukur kami panjatkan kehadiran Allah S.W.T Tuhan Yang Maha Esa yang telah memberikan limpahan rahmat dan hidayah-Nya sehingga tugas akhir dengan judul “Analisis Performa *Website* dengan Menggunakan *Forecast Instance Load Balancer* Skala *Enterprise* dengan Model *Multi Linear Regression* dan *Xgboost Regression*” ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Dalam penyusunan penelitian ini disajikan hasil penelitian terkait judul yang telah diangkat dan telah melalui proses pencarian referensi dari berbagai sumber baik jurnal penelitian, buku maupun dari situs-situs di internet. Sehingga dapat terbentuk solusi dalam penyelesaian Tugas Akhir ini.

Penulis menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak mulai dari masa perkuliahan sampai dengan masa penyusunan tugas akhir, sangatlah sulit untuk menyelesaikan tugas akhir ini tanpa bantuan dan dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis menyampaikan ucapan terima kasih sedalam-dalamnya kepada:

- 1) Tuhan Yang Maha Esa atas semua berkat, karunia serta pertolongan-Nya yang tiada henti, yang telah diberikan kepada penulis disetiap langkah dalam pembuatan program hingga penulisan laporan skripsi.
- 2) Kedua orang tua penulis, saudara dan ponakan penulis, serta keluarga yang senantiasa berada di samping penulis untuk memberikan kekuatan, motivasi, inspirasi, materi, bimbingan moral, kepercayaan dan kasih sayang yang tiada batas kepada penulis.
- 3) Bapak Dr. Eng Zulkifli Tahir, S.T., M.SC., selaku pembimbing I yang telah memberikan banyak bimbingan, inspirasi, motivasi dan masukan yang bermanfaat kepada penulis.
- 4) Bapak Ir. Muhammad Alief Fahdal Imran Oemar, S.T., M.Sc., selaku pembimbing II yang telah banyak memberi keyakinan, perhatian, bimbingan, motivasi dan masukan yang berharga bagi penulis.
- 5) Bapak A. Ais Prayogi Alimuddin, S.T., M.Eng. serta Bapak Prof. Dr. Ir. Andani, M.T. selaku dosen penguji yang telah memberikan masukan serta saran sehingga laporan skripsi ini menjadi lebih baik.
- 6) Prof. Dr. Ir. Indrabayu, ST., MT., M.Bus.Sys., IPM, ASEAN. Eng., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin atas bantuan dan bimbingannya selama masa perkuliahan penulis. Bapak Robert dan Ibu Yuanita serta segenap dosen dan staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah bantu kelancaran penyelesaian tugas akhir penulis.



- 8) Kepada kakanda Teknik Informatika Angkatan 2018 (SYNCRONOUS'18), yang telah banyak membantu penulis pada saat perkuliahan baik akademik maupun non akademik.
- 9) Teman-teman Teknik Informatika Angkatan 2020 (REZOLVER'20), selaku rekan belajar, rekan berpikir, dan rekan dalam segala hal selama masa perkuliahan.
- 10) Yogi Satrya Putra, seseorang yang tiada henti memberi dukungan serta semangat kepada penulis sejak perkuliahan hingga penyelesaian tugas akhir.
- 11) Kepada sahabat-sahabat penulis, Iksan, Thoriq, Adit yang telah memberikan bantuan dalam proses penyelesaian tugas akhir ini.

Akhir kata, penulis berharap semoga Tuhan yang Maha Kuasa berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Penulis menyadari bahwa masih banyak kekurangan dalam penulisan tugas akhir ini. Oleh karena itu, penulis menerima segala masukan, kritik dan saran untuk menjadikan tugas akhir ini lebih baik lagi. Semoga Tugas Akhir ini dapat memberikan manfaat dan menambah wawasan bagi semua pihak yang membacanya.

Gowa, Juli 2024

Penulis



BAB I PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi telah membuat situs *website* berfungsi sebagai jantung dari hampir setiap institusi, salah satu yang sangat dirasakan adalah perkembangan *search engine*. Karena *search engine* menjadi lebih pintar dari sebelumnya, orang memilih untuk mencari informasi apapun melaluinya. Semakin banyak organisasi memberi informasi di *website*, semakin banyak pengunjung yang mengunjungi *website* tersebut (Rakhmawati dkk., 2017). *Website* sebagai salah satu bentuk teknologi informasi untuk mendapatkan informasi secara mudah dalam waktu singkat (Sirikitsathian dkk., 2017). Peran *website* dalam suatu organisasi tidak terbatas sebagai media informasi, tetapi juga sebagai salah satu indikator dalam menilai kualitas. *Website* suatu organisasi perlu mendapat perhatian secara berkala untuk menjaga konsistensi dan meningkatkan kualitas sehingga tercapai penilaian yang baik. Mengevaluasi situs web berdasarkan matrik kualitatif seperti kegunaan, fungsionalitas, keandalan, efisiensi, dan karakteristik (Cerqueira dkk., 2020). Dalam lingkungan berskala *enterprise*, di mana situs web harus mampu menangani lalu lintas tinggi dan beban kerja dinamis, mengelola kinerja dan *fault tolerance website* menjadi tantangan yang kompleks. Akibatnya, organisasi di lingkungan perusahaan semakin tertarik untuk menemukan solusi dan strategi yang efektif untuk mengoptimalkan kinerja dan ketahanan situs web mereka. Teori peramalan (*forecasting*) didasarkan pada premis bahwa pengetahuan saat ini dan masa lalu dapat digunakan untuk membuat prediksi tentang masa depan (Petropoulos dkk., 2022). Dalam *load forecasting* untuk situs web, berbagai metode dan model dapat digunakan. Pemodelan *time series*, umumnya digunakan untuk menganalisis data historis dan mengidentifikasi pola dan tren (Abd. Razak dkk., 2022). Metode *load balancing* skala *enterprise* yang menggunakan teknik prakiraan, yang dikenal sebagai "forecast instance *load balancing*", menjadi pendekatan yang inovatif dan

dalam mengoptimalkan performa *website* dan meningkatkan *fault website*. Dengan menerapkan model *Multi Linear Regression* dan *Regression* untuk memproyeksikan beban permintaan masa depan,



pendekatan ini memungkinkan pengelola sistem untuk merencanakan alokasi sumber daya dengan lebih efektif, sehingga menghindari potensi kelebihan beban pada server tertentu.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka identifikasi masalah yang ditemukan, yaitu:

1. Bagaimana performa sebuah *website* di lingkungan skala *enterprise* dapat dipengaruhi oleh penggunaan *Forecast Instance Load Balancer*?
2. Bagaimana *fault tolerance* sebuah *website* di lingkungan skala *enterprise* dapat ditingkatkan dengan mengimplementasikan model *Multi Linear Regression* dan *Xgboost Regression*?

1.3 Tujuan Penelitian/Perancangan

Tujuan dari penelitian ini diantaranya, yaitu:

1. Menganalisis pengaruh penggunaan *Forecast Instance Load Balancer* pada performa sebuah *website* di lingkungan skala *enterprise*, dengan fokus pada peningkatan efisiensi dan penyebaran beban kerja yang merata di antara *server-instance*.
2. Meningkatkan *fault tolerance* sebuah *website* di lingkungan skala *enterprise* dengan menerapkan model *Multi Linear Regression* dan *Xgboost Regression* untuk meramalkan tren dan pola kesalahan yang mungkin terjadi, sehingga memungkinkan adopsi langkah-langkah pencegahan atau pemulihan yang lebih cepat.

1.4 Manfaat Penelitian/Perancangan

Manfaat yang diharapkan dalam penelitian ini, yaitu:

1. Efisiensi beban kerja: Penelitian ini diharapkan dapat memberikan panduan praktis bagi organisasi atau perusahaan dalam skala *enterprise* untuk meningkatkan performa *website* mereka. Dengan mengadopsi metode *Forecast Instance Load Balancer* dan model *Multi Linear Regression* dan *Xgboost Regression*, mereka dapat mengoptimalkan distribusi beban kerja dan



meningkatkan efisiensi serta responsivitas website, menghadirkan pengalaman pengguna yang lebih baik.

2. Langkah pencegahan: Dengan menerapkan tingkat *fault tolerance website* yang lebih baik melalui model *Multi Linear Regression* dan *Xgboost Regression*, website di lingkungan skala enterprise dapat lebih siap menghadapi potensi kesalahan atau gangguan. Hal ini dapat membantu organisasi mengelola risiko downtime dan meminimalkan dampak negatif pada ketersediaan layanan, sehingga menjaga kepercayaan pelanggan dan reputasi perusahaan.

1.5 Ruang Lingkup/Asumsi perancangan

Agar penelitian ini tepat sasaran, batasan-batasan permasalahan terhadap topik penelitian ini ditentukan sebagai berikut:

1. *Website* dengan lingkup Skala *Enterprise*.
2. Ketergantungan pada Data Historis *Website*.
3. Fokus pada Load Balancing dan *Time Series Regression*.
4. Melakukan simulasi menggunakan AWS.



BAB II TINJAUAN PUSTAKA

2.1 Website

Website adalah sekumpulan halaman web yang saling berhubungan termasuk sebuah *homepage*, umumnya berada pada peladen yang sama, disiapkan, dan dipelihara sebagai kumpulan informasi oleh seseorang, kelompok, atau organisasi. *Website* dibuat menggunakan gabungan berbagai bahasa pemrograman seperti PHP, MySQL, HTML, dan lain sebagainya (Suliman, 2020).

Website merupakan tempat penyedia informasi yang digunakan dalam mencari informasi serta data melalui internet. Pengguna *website* tidak hanya dari kalangan pekerja maupun pelajar tetapi diakses juga oleh masyarakat umum (Huda, 2021).

Website merupakan suatu sistem informasi yang sangat mudah dijumpai pada saat ini seiring dengan pesatnya perkembangan dunia dan teknologi informasi, di mana dunia teknologi khususnya internet berkembang menyeluruh di seluruh belahan negara dengan memudahkannya manusia dalam bertukar informasi secara global. Perkembangan internet dalam hal ini *website* tidak hanya dirasakan dan dipergunakan oleh kalangan pebisnis, namun juga oleh institusi pendidikan dalam menjalankan aktivitas pendidikan seperti untuk penyampaian informasi, dan berbagi (*sharing*) dokumen antar departemen. Oleh karena itu, diperlukan *website* yang baik dan berkualitas. Sehingga mampu memberikan kegunaan sesuai dengan tujuan pembuatan *website*. Terdapat beberapa hal umum yang dipakai sebagai acuan sebagai bahan pertimbangan untuk menentukan baik buruknya kualitas sebuah *website*. Sebagai bahan pertimbangan yaitu: kecepatan download, isi mudah dibaca, dan tata letak/desain yang konsisten (Suliman, 2020).

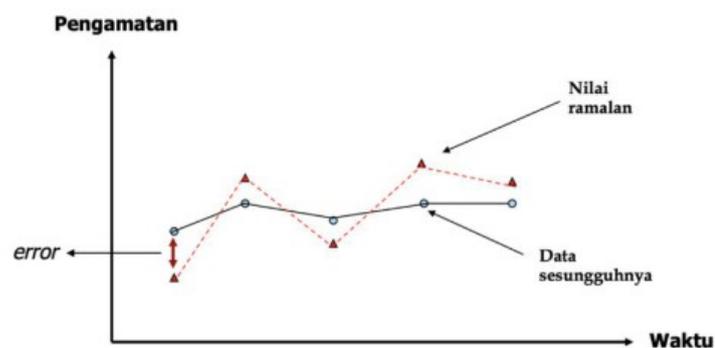
2.2 Forecasting

Forecasting mengacu pada proses membuat prediksi atau perkiraan tentang atau tren di masa depan berdasarkan data historis, pola, dan informasi lainnya. Ini biasanya digunakan di berbagai bidang seperti ekonomi, cuaca, dan bisnis untuk mengantisipasi hasil di masa depan dan membuat



keputusan yang tepat. Metode *forecasting* dapat berkisar dari analisis deret waktu sederhana hingga model statistik kompleks dan algoritma *machine learning*, bergantung pada sifat data dan domain spesifik penerapannya. Tujuan *forecasting* adalah untuk mengurangi ketidakpastian dan membantu dalam perencanaan, alokasi sumber daya, dan proses pengambilan keputusan (Addison, W., 2020).

Bidang *forecasting* mencakup berbagai teknik dan metodologi yang digunakan untuk memprediksi peristiwa atau tren masa depan berdasarkan data dan pola historis. Dalam konteks peramalan beban untuk sistem terdistribusi, model deret waktu linier telah digunakan untuk memproses data prediksi dan menganalisis karakteristik beban. Model ini menggabungkan pemrosesan dan analisis data, yang bertujuan untuk memprediksi pola beban secara akurat dan melakukan koreksi waktu nyata berdasarkan data beban aktual. Algoritma deret waktu linier adalah metode statistik yang dapat menangani faktor acak eksternal dan variabel tak terduga, seperti cuaca dan suhu, untuk memperkirakan model dan parameter teoritisnya, sehingga memperoleh hasil perkiraan atau lebih akurat (Naqas A., 2022).



Gambar 1 Ilustrasi *Forecasting*

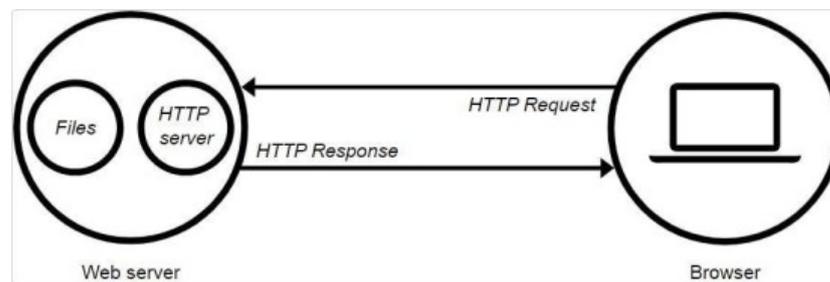
2.3 Web Server

Web Server adalah sistem *software* atau *hardware* yang menyajikan konten kepada pengguna melalui internet. Ia menggunakan *Hypertext Transfer Protocol* (HTTP) untuk mengirimkan halaman web dan konten lainnya ke browser web pengguna. Saat pengguna memasukkan alamat web ke browsernya, browser akan mengirimkan permintaan ke server web, yang kemudian memproses permintaan dan mengembalikan konten yang diminta ke browser pengguna. Server



web sangat penting untuk menghosting situs web dan aplikasi web, sehingga memungkinkannya diakses oleh pengguna di seluruh dunia (Thoyyib, A., 2022).

Secara sederhana, tugas web server adalah untuk menerima permintaan dari klien kemudian mengirimkan kembali data yang diminta oleh klien tersebut. Sama halnya dengan komputer klien, komputer server juga harus terhubung dengan jaringan internet untuk dapat diakses oleh klien. Ilustrasi cara kerja layanan web server dapat dilihat pada gambar 2.2 (What is web server: Developer Mozilla, 2022).



Gambar 2 Ilustrasi cara kerja web server

Berdasarkan gambar 2.1. pada saat browser (klien) melakukan permintaan data ke web server, maka instruksi permintaan data tersebut akan dikemas di dalam TCP yang merupakan protokol *transport* dan kemudian akan dikirimkan ke alamat yang dibutuhkan yaitu HTTP atau HTTPS. Data yang diminta oleh browser ke web server disebut dengan *HTTP request* yang kemudian akan dicari oleh web server di dalam komputer server. Jika ditemukan, data tersebut akan dikemas oleh web server dalam TCP dan dikirim kembali ke browser untuk ditampilkan. Data yang dikirim dari server ke browser dikenal dengan *HTTP response*. Jika data yang diminta oleh browser tersebut ternyata tidak ditemukan oleh web server, maka web server akan menolak permintaan tersebut dan browser akan menampilkan notifikasi *Page Not Found* atau Error 404.

2.4 Instance Load Balancer



Instance Load Balancer adalah jenis penyeimbang beban yang beroperasi di *application layer* jaringan dan dirancang untuk mendistribusikan lalu lintas ke beberapa server *instance*. Ini secara cerdas mengarahkan permintaan ke server yang paling sesuai berdasarkan faktor-faktor seperti kesehatan server,

kapasitas, dan beban saat ini. Jenis penyeimbang beban ini biasanya digunakan dalam lingkungan komputasi awan, di mana virtual server *instance* perlu menangani lalu lintas masuk secara efisien. Dengan mendistribusikan beban ke beberapa *instance*, maka *instance Load Balancer* membantu mengoptimalkan pemanfaatan sumber daya, meningkatkan *performance*, dan meningkatkan ketersediaan sistem secara keseluruhan (Rouse, M., 2019).

Penggunaan *Instance Load Balancer* sangat bermanfaat dalam lingkungan *cloud computing*, di mana virtual server *instances* perlu menangani lalu lintas masuk secara efisien. Dengan mendistribusikan beban ke beberapa *instance*, *Load Balancer* ini membantu mengoptimalkan pemanfaatan sumber daya, meningkatkan kinerja, dan meningkatkan ketersediaan sistem secara keseluruhan. Ini dapat menggunakan berbagai algoritma untuk menentukan cara mendistribusikan lalu lintas, seperti *round-robin*, *least connections*, or *weighted distribution*, memastikan bahwa setiap server digunakan secara efisien dan mencegah kelebihan beban atau kekurangan pemanfaatan (Minkenbergh, C., 2019).

2.5 Amazon Web Service (AWS)

Amazon Web Services (AWS) adalah platform komputasi awan komprehensif dan banyak digunakan yang menawarkan beragam layanan untuk membantu bisnis dan individu membangun dan mengelola infrastruktur TI mereka. AWS menyediakan layanan seperti daya komputasi, opsi penyimpanan, jaringan, database, pembelajaran mesin, dan banyak lagi, yang memungkinkan pengguna mengakses dan memanfaatkan sumber daya ini dengan basis bayar sesuai penggunaan. Dengan jaringan pusat data globalnya, AWS memungkinkan pengguna untuk menerapkan aplikasi dan layanan dengan cepat dan aman, sekaligus memberikan skalabilitas dan fleksibilitas untuk memenuhi kebutuhan bisnis yang terus berubah.

Salah satu pendekatan untuk mengimplementasikan server web di AWS adalah melalui penggunaan Teknik Formasi Cloud (CFT), yang memungkinkan in dan pengelolaan infrastruktur sebagai kode. Dengan menggunakan kode u YAML, pengguna dapat menentukan sumber daya dan konfigurasi yang n untuk server web mereka, memungkinkan otomatisasi dan pengulangan



dalam proses penerapan. Pendekatan ini menawarkan keuntungan seperti pengurangan beban kerja, penghematan waktu, dan penggunaan kembali kode, sehingga memudahkan pengelolaan dan pemeliharaan infrastruktur server web. Secara keseluruhan, AWS menyediakan platform yang kuat dan fleksibel untuk membangun dan menerapkan server web, menawarkan kinerja dan keunggulan lebih tinggi dibandingkan server tradisional di lokasi (Mangayarkarasi, M., 2021).

2.5.1 Amazon *Elastic Compute Cloud* (Amazon EC2)

Amazon Elastic Compute Cloud (Amazon EC2) adalah komponen kunci dalam penerapan server web yang sangat *scalable* dan memuat seimbang di *AWS cloud*. Instance EC2 digunakan untuk menghosting server web, yang menunjukkan kemampuan platform dalam menyediakan daya komputasi dengan cara yang terukur dan fleksibel. Hal ini memungkinkan penanganan permintaan masuk yang efisien dan keberhasilan instalasi server web.

Misalnya, *instance* AWS *Elastic Compute Cloud* (EC2) dapat digunakan untuk menghosting server web, sementara *Elastic File System* (EFS) menyediakan penyimpanan *file* yang dapat diskalakan untuk server. Selain itu, AWS menyediakan penyeimbang beban dan grup target untuk mendistribusikan lalu lintas masuk ke beberapa *instance* EC2, serta *Auto Scaling Group* (ASG) untuk secara otomatis menyesuaikan jumlah *instance* berdasarkan permintaan. Layanan ini, bersama dengan penggunaan *virtual private cloud* (VPC) untuk jaringan, memungkinkan terciptanya infrastruktur server web yang kuat dan efisien di *AWS cloud*.

Penggunaan *instance* EC2 dengan Nginx, bersama dengan layanan AWS lainnya, berkontribusi pada penciptaan infrastruktur server web yang kuat. Fleksibilitas dan skalabilitas *instance* EC2, dipadukan dengan kemampuan untuk mengkonfigurasi dan mengelolanya dengan mudah, menjadikan Amazon EC2 sumber daya berharga untuk menghosting server web dan menangani berbagai beban kerja (Mangayarkarasi, M., 2021).



2.5.2 Load Balancer dan Target Groups

Load Balancer dan *Target Groups* adalah dua komponen penting dalam infrastruktur web server yang dapat diandalkan dan *scalable* di *AWS cloud*. *Load Balancer* berperan dalam mendistribusikan beban kerja dari pengguna ke berbagai server, sehingga memastikan ketersediaan tinggi dan keseimbangan beban di seluruh zona. Terdapat tiga jenis *Load Balancer* yang tersedia di *AWS cloud*, yaitu *Network Load Balancer*, *Application Load Balancer*, dan *Classic Load Balancer*. *Network Load Balancer* cocok untuk mentransfer data pada level TCP dan UDP, mampu menangani jutaan permintaan per detik, dan menggunakan algoritma *Hash* untuk membagi beban berdasarkan sumber IP, nomor port, dan tujuan IP. Sementara itu, *Application Load Balancer* digunakan untuk mengimbangi lalu lintas HTTP dan HTTPS pada level Layer 7, dengan *routing* yang didasarkan pada *Target Groups*. *Classic Load Balancer*, meskipun merupakan *Load Balancer* tertua di AWS, tidak direkomendasikan untuk digunakan karena hanya melakukan *load balancing* dasar di antara server.

Target Groups merupakan kelompok server yang terhubung dengan *Load Balancer* dan memberikan petunjuk kepada *Load Balancer* untuk mengarahkan lalu lintas ke server mana. Setiap server harus terdaftar dalam *Target Groups* agar *Load Balancer* dapat mendistribusikan beban kerja di antara server yang terdaftar. *Target Groups* juga memungkinkan adanya pengecekan kesehatan otomatis, sehingga jika satu server tidak merespon, *Load Balancer* akan mengarahkan lalu lintas ke server lain yang sehat. Dengan menggunakan *Target Groups*, pengembang dapat mengatur *routing* lalu lintas secara lebih canggih dengan menentukan *port* dan alamat IP. Secara default, *Target Groups* menggunakan algoritma *Round Robin* untuk merutekan lalu lintas di antara target (server) yang terdaftar, dan lebih dari satu listener dapat ditambahkan ke sebuah *Target Group* untuk mengatur lalu lintas dengan lebih fleksibel (Mangayarkarasi, M., 2021).

2.5.3 Auto Scaling Group (ASG)



Auto Scaling Group (ASG) adalah komponen penting di AWS untuk otomatisasi dan pengelolaan server. ASG dapat membuat atau mengurangi server berdasarkan kebijakan penskalaan yang ditentukan,

menjadikan server dapat diskalakan secara dinamis dan ketersediaan tinggi. Ini meluncurkan server baru berdasarkan templat peluncuran tertentu atau konfigurasi yang melekat padanya.

ASG memungkinkan pengguna untuk mengatur kapasitas server minimum, maksimum, dan yang diinginkan, dan secara otomatis meluncurkan server baru hingga jumlah maksimum berdasarkan beban kerja, sehingga mengurangi beban kerja dan menghemat biaya. Kebijakan penskalaan, seperti penskalaan pelacakan target, penskalaan langkah, dan penskalaan sederhana, dilampirkan pada ASG untuk menentukan kapan meluncurkan atau mengakhiri server berdasarkan beban kerja (Mangayarkarasi, M., 2021).

2.5.4 *Virtual Private Cloud (VPC)*

Virtual Private Cloud (VPC) adalah komponen mendasar di AWS yang memungkinkan pengguna membuat jaringan virtual terisolasi mereka sendiri di dalam *AWS cloud*. VPC memberikan kontrol atas lingkungan jaringan virtual, termasuk pemilihan rentang alamat IP, pembuatan subnet, dan konfigurasi tabel rute dan *gateway* jaringan. Subnet dapat ditetapkan sebagai publik atau pribadi, dengan subnet publik mengizinkan akses publik dan subnet pribadi membatasi akses ke alamat IP tertentu, sehingga meningkatkan keamanan (Mangayarkarasi, M., 2021).

2.6 *Enterprise*

Enterprise, dalam konteks arsitektur perusahaan, merujuk pada suatu kumpulan organisasi yang memiliki tujuan dan garis bawah tunggal, seperti perusahaan secara keseluruhan, departemen tunggal, atau serangkaian organisasi yang dihubungkan. *Enterprise architecture (EA)* merupakan pengorganisasian secara logis untuk proses bisnis utama dan kemampuan yang mencerminkan kebutuhan integrasi dan standarisasi model operasi perusahaan. EA melibatkan sejumlah prinsip, metode, dan model yang digunakan dalam perancangan dan struktur organisasi perusahaan, proses bisnis, sistem informasi, dan tur (Sasmito, W. S., 2020).



Implementasi arsitektur perusahaan bertujuan untuk memberikan cetak biru dan usulan atau platform kerja kepada suatu organisasi, memberikan kerangka dasar dalam mengembangkan sistem informasi yang terintegrasi untuk mendukung kebutuhan organisasi. Pemodelan arsitektur perusahaan menggunakan metodologi pengembangan TOGAF ADM dapat menghasilkan model dan kerangka dasar dalam mengembangkan sistem informasi yang terintegrasi untuk mendukung kebutuhan organisasi (Yoganingrum, A., 2022).

2.7 Regresi *Time Series*

Regresi *time series* adalah metode statistik yang digunakan untuk memodelkan hubungan antara variabel dependen dengan variabel independen dalam suatu rentetan waktu. Metode ini memungkinkan untuk memprediksi nilai-nilai masa depan berdasarkan data historis yang telah tercatat. Salah satu contoh penerapan regresi *time series* adalah dalam memprediksi permintaan di sebuah perusahaan.

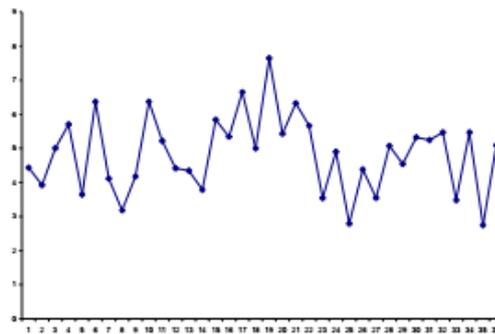
Penerapan regresi *time series* seringkali melibatkan analisis data historis untuk memahami pola dan tren yang mungkin mempengaruhi variabel dependen. Dengan memahami hubungan antara variabel-variabel yang relevan, perusahaan dapat menggunakan model *Multi Linear Regression* dan *Xgboost Regression* untuk membuat perkiraan yang lebih akurat terkait dengan permintaan, penjualan, atau kinerja operasional lainnya. Selain itu, regresi *time series* juga dapat membantu perusahaan dalam perencanaan kapasitas, pengelolaan rantai pasokan, dan pengambilan keputusan strategis lainnya (Abd. Razak, 2022).

Dengan melakukan analisis *Time Series*, maka pola gerakan data dapat diketahui. Menurut Makridakis & Wheelwright (1999), pola data deret waktu dapat dibedakan menjadi empat jenis, yaitu:

1. Pola Konstan

Pola horizontal terjadi bilamana nilai data berfluktuasi di sekitar nilai rata-rata konstan. Deret seperti ini stasioner terhadap nilai rata-ratanya. Berikut adalah ot data deret waktu yang mengikuti pola horizontal.

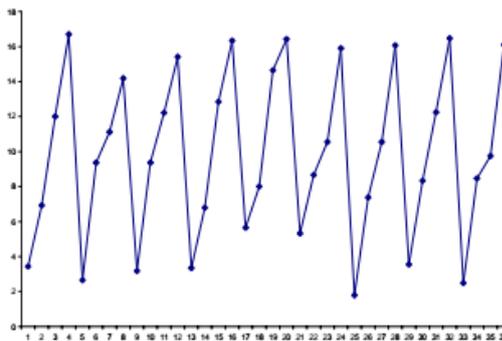




Gambar 3 Pola Data Konstan

2. Pola Musiman (*Seasonal*)

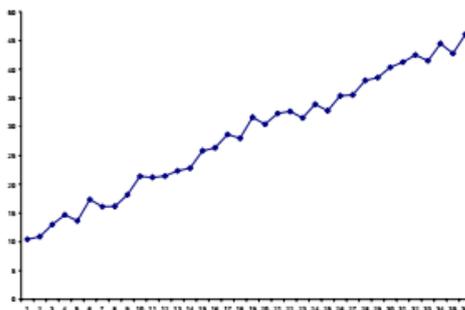
Pola musiman (*Seasonal*) terjadi bila suatu deret dipengaruhi oleh faktor musiman dan cenderung berulang, namun teratur dalam mingguan, bulanan, ataupun kuartalan. Berikut adalah bentuk plot data deret waktu yang mengikuti pola musiman.



Gambar 4 Pola Data Musiman

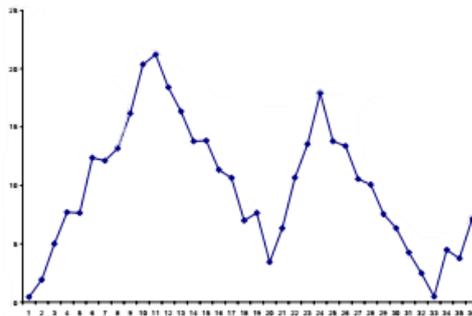
3. Pola *Trend*

Pola *trend* terjadi bila plot data menyebar membentuk suatu kecenderungan naik ataupun menurun secara konstan. Pola *trend* sangat berguna untuk membuat peramalan (*forecasting*) yang merupakan perkiraan masa depan. Berikut adalah bentuk plot data deret waktu yang mengikuti pola tren.

Gambar 5 Pola Data *Trend*

4. Pola Siklik

Pola siklis terjadi karena adanya fluktuasi bergelombang data yang terjadi di sekitar garis *trend*. Selain itu, pola siklis terjadi bilamana datanya dipengaruhi oleh fluktuasi ekonomi jangka panjang seperti yang berhubungan dengan siklus bisnis. Dalam kegiatan bisnis dan ekonomi, gerakan-gerakan data dianggap siklis apabila timbul kembali setelah jangka waktu lebih dari satu tahun. Berikut adalah bentuk plot data deret waktu yang mengikuti pola siklis.



Gambar 6 Pola Data Siklik

2.7.1 Multiple Linear Regression

Multiple Linear Regression (MLR) adalah metode statistik yang digunakan untuk memahami hubungan antara dua atau lebih variabel independen dengan variabel dependen. Dalam regresi linear berganda, variabel dependen diprediksi berdasarkan kombinasi linear dari variabel independen. Rumus umum regresi linear berganda dapat dituliskan sebagai berikut:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n + \varepsilon$$

Keterangan:

Y = variabel dependen.

β_0 = konstanta.

$\beta_1, \beta_2, \dots, \beta_n$ = koefisien regresi yang mengukur pengaruh variabel independen terhadap variabel dependen.

X_1, X_2, \dots, X_n = variabel independen.

ε = kesalahan acak.



am analisis regresi linear berganda, terdapat beberapa langkah yang perlu
1, seperti menentukan tujuan analisis, persiapan data, analisis eksploratif,
parameter, pemilihan prediktor, analisis residual, dan penarikan

kesimpulan. Langkah-langkah ini membantu dalam memahami hubungan antara variabel independen dan dependen serta membuat prediksi yang akurat berdasarkan data yang ada. Penting untuk mempertimbangkan tujuan analisis, signifikansi prediktor, dan potensi asosiasi non-linear dalam model regresi.

Regresi linear berganda digunakan dalam *forecasting* dengan cara memprediksi nilai variabel dependen berdasarkan kombinasi linear dari beberapa variabel independen. Dalam konteks *forecasting*, *Multi Linear Regression* memungkinkan untuk memperkirakan nilai variabel dependen di masa depan berdasarkan hubungan linier dengan variabel independen yang telah diamati. Dengan menggunakan *Multi Linear Regression*, kita dapat mengidentifikasi pengaruh variabel independen terhadap variabel dependen dan membuat prediksi yang lebih akurat berdasarkan data historis yang telah ada.

Selain itu, *Multi Linear Regression* juga dapat digunakan untuk memprediksi berbagai variabel lain, seperti cuaca, data lalu lintas, atau penjualan produk. Dengan memahami hubungan antara variabel independen dan dependen, *Multi Linear Regression* membantu dalam mengidentifikasi faktor-faktor kunci yang mempengaruhi variabel yang diprediksi, sehingga memungkinkan untuk membuat keputusan yang lebih baik berdasarkan prediksi yang dihasilkan (Ulgen, T., 2020).

Model *Multi Linear Regression* cocok untuk melakukan *forecasting* karena model ini memungkinkan untuk memasukkan beberapa prediktor yang mungkin berhubungan dengan variabel respons. Dengan mempertimbangkan beberapa faktor yang mempengaruhi variabel dependen, *Multi Linear Regression* dapat memberikan estimasi yang lebih baik daripada model regresi lainnya. Selain itu, *Multi Linear Regression* memungkinkan untuk memprediksi nilai variabel dependen berdasarkan kombinasi linear dari variabel independen, sehingga dapat memberikan gambaran yang lebih komprehensif tentang hubungan antara variabel-variabel tersebut.

Model *Multi Linear Regression* (MLR) bekerja dengan cara memodelkan hubungan linear antara satu variabel dependen dengan dua atau lebih variabel



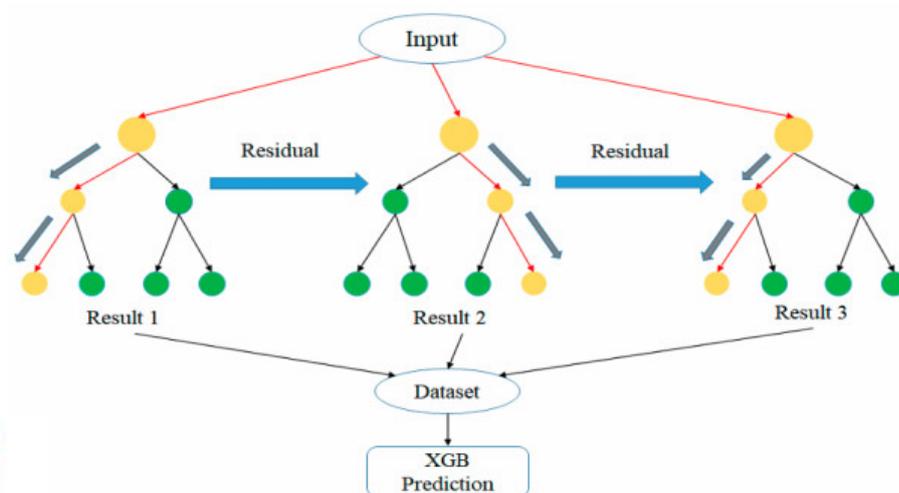
independen. Dalam *Multi Linear Regression*, variabel dependen diprediksi sebagai kombinasi linear dari variabel independen dengan bobot tertentu. Bobot ini menunjukkan seberapa besar pengaruh setiap variabel independen terhadap

variabel dependen. Proses ini melibatkan estimasi koefisien regresi yang optimal untuk setiap variabel independen, sehingga model dapat memberikan prediksi yang akurat berdasarkan data yang ada.

Selain itu, *Multi Linear Regression* juga melibatkan analisis residual untuk memeriksa sejauh mana model sesuai dengan data yang diamati. Residual adalah selisih antara nilai observasi aktual dan nilai prediksi dari model. Dengan menganalisis residual, kita dapat mengevaluasi sejauh mana model sesuai dengan data, apakah ada pola yang tidak terduga atau kekurangan dalam model yang perlu diperbaiki. Dengan demikian, *Multi Linear Regression* memungkinkan untuk mengidentifikasi hubungan antara variabel-variabel dan membuat prediksi yang akurat berdasarkan hubungan tersebut (Rivera, R., 2020).

2.7.2 Extreme Gradient Boosting Regression

Extreme Gradient Boosting (XGBoost) adalah salah satu algoritma *machine learning* yang digunakan untuk *regression* dan *classification*. Model XGBoost menggunakan pendekatan *ensemble learning* dengan menggabungkan beberapa pohon keputusan untuk membuat prediksi yang akurat. Dalam *regression*, XGBoost bekerja dengan cara menggabungkan prediksi dari beberapa model pohon keputusan yang lebih sederhana untuk menghasilkan prediksi akhir yang lebih kuat dan lebih tepat. Model ini memperhitungkan hubungan antara variabel *input* dan *output* untuk membuat prediksi yang optimal (Lingling, N., 2020).



Gambar 7 Representasi Konsep *XGBoost* Model



Gambar 7 dapat menggambarkan bagaimana *XGBoost* bekerja dalam proses training model. Dalam gambar tersebut, terlihat bahwa *XGBoost* menggabungkan *bagging* dan *boosting*. Model ini dibangun secara iteratif dengan menambahkan pohon baru untuk memperbaiki kesalahan yang dihasilkan oleh pohon sebelumnya, sehingga meningkatkan kekuatan keseluruhan model. Proses ini berlangsung secara berulang hingga tidak ada peningkatan lebih lanjut atau jumlah pohon mencapai batas maksimum yang ditentukan.

XGBoost menggunakan teknik *gradient boosting* untuk menghasilkan model yang lebih kuat dan akurat. Dalam gambar 7, terlihat bahwa model *XGBoost* dibangun dengan menambahkan pohon keputusan secara bertahap untuk mengoreksi kesalahan prediksi sebelumnya. Hal ini memungkinkan model untuk belajar dari kesalahan dan meningkatkan kinerja secara keseluruhan. Dengan pendekatan ini, *XGBoost* mampu mengatasi masalah *overfitting* dan *underfitting* yang sering terjadi dalam model *machine learning*. Proses training model *XGBoost* melibatkan parameter seperti *maximum depth of the tree structure*, *number of samples*, *learning rate*, dan jenis *booster* yang digunakan. (Wang, C., 2022).

Untuk melakukan *forecasting* dengan *XGBoost*, langkah pertama adalah membagi data menjadi training dan testing sets. Selanjutnya, dilakukan pemilihan fitur penting untuk dimasukkan ke dalam model. *XGBoost* akan melakukan *training* dengan data *training* yang telah dikelompokkan menggunakan *Gaussian mixture model* (GMM) untuk menghasilkan beberapa model *XGBoost* yang lebih spesifik. Kemudian, model-model tersebut akan digunakan untuk membuat prediksi berdasarkan probabilitas dari GMM (Lingling, N., 2020).

