



Jurnal Information System Development



Jurnal Information System Development

Mendeteksi Penyakit Jantung Menggunakan Machine Learning Dengan Algoritma Logistic Regression

Jefri Junifer Pangaribuan, Henry Tanjaya, Kenichi

Feature Addition to The Color Harmonization Application in Two Dimensional Image

Albert Antonio, Irene A. Lazarusli, Samuel Lukas, Petrus Widjaja

Aplikasi Marketplace Berbasis Website untuk Pencarian Jasa Musisi

Alya M. Amalia, Andree E. Widjaja, Hery, Calandra A. Haryani

Identifikasi Data Drifting pada Aplikasi Internet of Thing (IOT)

Alessandro Luiz Kartika, I Made Murwantara, Pujianto Yugopuspito

Rancangan CNC Plotter Untuk Menulis dan Menggambar

Christoper Tanato, Alfa Satya Putra

Implementasi Metode Analytic Hierarchy Process Untuk Pemilihan Saham Terbaik Berbasis Website

Rudolfo Rizki Damanik, Leonardy Khanady

Prediksi Kesembuhan Pasien Covid-19 di Indonesia Melalui Terapi Menggunakan Metode Naive Bayes

Okky Putra Barus, Anton Tehja

Kecerdasan Buatan Untuk Blokus Classic Menggunakan Heuristik, Floodfill, dan Algoritma Greedy

Hong Liang Cai, Sebastian Aldi, Winston Renatan



EDITORIAL BOARD



I S D

ISSN : 2528-5114

Jurnal Information System Development



Jurnal Information System Development

Mendeteksi Penyakit Jantung Menggunakan Machine Learning Dengan Algoritma Logistic Regression
Jefri Junifer Pangaribuan, Henry Tanjaya, Kenichi

Feature Addition to The Color Harmonization Application in Two Dimensional Image
Albert Antonio, Irene A. Lazarusli, Samuel Lukas, Petrus Widjaja

Aplikasi Marketplace Berbasis Website untuk Pencarian Jasa Musisi
Alya M. Amalia, Andree E. Widjaja, Hery, Calandra A. Haryani

Identifikasi Data Drifting pada Aplikasi Internet of Thing (IOT)
Alessandro Luiz Kartika, I Made Murwantara, Pujianto Yugopuspito

Rancangan CNC Plotter Untuk Menulis dan Menggambar
Christopher Tanato, Alfa Satya Putra

Implementasi Metode Analytic Hierarchy Process Untuk Pemilihan Saham Terbaik Berbasis Website
Rudolfo Rizki Damanik, Leonardy Khanady

Prediksi Kesembuhan Pasien Covid-19 di Indonesia Melalui Terapi Menggunakan Metode Naive Bayes
Okky Putra Barus, Anton Tehja

Kecerdasan Buatan Untuk Blokus Classic Menggunakan Heuristik, Floodfill, dan Algoritma Greedy
Hong Liang Cai, Sebastian Aldi, Winston Renatan

Penanggung Jawab : Dekan Fakultas Ilmu Komputer Universitas Pelita Harapan

Penasehat : Koordinator LPPM Universitas Pelita Harapan Medan

Mitra Bestari : Irene Astuti Lazarusli, S.Kom., M.T.
Arnold Ariwibowo, S.T., M.T.
I Made Murwantara, S.Si., M.Kom., Ph.D
Frans Panduwinata, S.Kom., M.T.
Kusno Prasetya, Ph.D.
Calandra Alencia Haryani, S.E., S.S.I., M.T.I.
Aditya Rama Mitra, S.Si., M.T.
Alfa Satya Putra, B.Sc., M.Sc.

Dewan Redaksi : Andree Emmanuel Widjaja, S.Kom., MBA., Ph.D. (Ketua)
Ir. Ju Sin, M.T.I. (Anggota)
Irene Astuti Lazarusli, S.Kom., M.T. (Anggota)
Arnold Ariwibowo, S.T., M.T. (Anggota)
Okky Putra Barus, S.Kom., M.M., M.T.I (Anggota)
Rudolfo Rizki Damanik, S.T., M.T. (Anggota)

Sekretaris Redaksi : Ferawaty, S.Kom., M.Kom.

Sekretariat : Mulia Dhamma, S.Kom., M.Kom.,
Yudhistira Adhitya Pratama, S. Kom., M.Kom.

Cover & Layout : William Lie

Alamat Redaksi : Universitas Pelita Harapan
Jl. Kapten Maulana Lubis No. 8 Lt. 1
Medan 20112 – Indonesia
email : ji.uphmedan@uph.edu

Jurnal Information System Development (ISD) hadir sebagai wadah bagi para Akademisi, Developer, Peneliti, dan Ilmuwan yang hendak menyumbangkan karya ilmiahnya bagi dunia ilmu pengetahuan di bidang Sistem Informasi.

Jurnal yang diterbitkan oleh Prodi Sistem Informasi Universitas Pelita Harapan ini menerima publikasi hasil pengembangan atau penelitian terbaru di bidang Sistem Informasi. Topik-topik meliputi pengembangan *software desktop, web, mobile, database system, artificial intelligence, data warehouse, data mining, UI/UX programming, IT infrastructure, Internet of Things, Game Development, Cyber Security*, dan topik-topik lainnya.

Setiap tahunnya, Jurnal ISD terbit dalam dua (2) periode yaitu pada Bulan Januari dan Juli



Jurnal Information System Development Universitas Pelita Harapan

Table of Contents

Vol. 6 No. 2 Juli 2021 | ISSN 2528-5114 (Online) | ISSN 2477-863X (Print)

Title	: Mendeteksi Penyakit Jantung Menggunakan Machine Learning Dengan Algoritma Logistic Regression	1 – 10
Author	: Jefri Junifer Pangaribuan, Henry Tanjaya, Kenichi	
Title	: Feature Addition to the Color Harmonization Application in Two Dimensional Image	11 – 17
Author	: Albert Antonio, Irene A. Lazarusli, Samuel Lukas, Petrus Widjaja	
Title	: Aplikasi Marketplace Berbasis Website Untuk Pencarian Jasa Musisi	18 – 32
Author	: Alya M. Amalia, Andree E. Widjaja, Hery, Calandra A. Haryani	
Title	: Identifikasi Data Drifting pada Aplikasi Internet of Things (IoT)	33 – 41
Author	: Pujianto Yugopuspito, Alessandro Luiz Kartika, I Made Murwantara	
Title	: Rancangan CNC Plotter Untuk Menulis dan Menggambar	42 – 47
Author	: Alfa Satya Putra, Christoper Tanato	
Title	: Implementasi Metode Analytic Hierarchy Process Untuk Pemilihan Saham Terbaik Berbasis Website	48 – 58
Author	: Rudolfo Rizki Damanik, Leonardy Khanady	
Title	: Prediksi Kesembuhan Pasien Covid-19 di Indonesia Melalui Terapi Menggunakan Metode Naïve Bayes	59 – 66
Author	: Okky Putra Barus, Anton Tehja	
Title	: Kecerdasan Buatan Untuk Blokus Classic Menggunakan Heuristik, Floodfill, dan Algoritma Greedy	67 – 75
Author	: Hong Liang Cai, Sebastian Aldi, Winston Renata	

MENDETEKSI PENYAKIT JANTUNG MENGGUNAKAN MACHINE LEARNING DENGAN ALGORITMA LOGISTIC REGRESSION

Jefri Junifer Pangaribuan^{1*)}, Henry Tanjaya²⁾, Kenichi³⁾

¹Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: jefri.pangaribuan@uph.edu¹⁾

²Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: ht6411@student.uph.edu²⁾

³Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: kk2848@student.uph.edu³⁾

**Penulis Korespondensi*

Abstract – Heart Disease or also called cardiovascular disease is one of the dangerous diseases that can cause death. As technology develops and the popularity of machine learning technology increases, machine learning technology can be used to help detect heart disease using patient data. There are various methods that can be used to diagnose whether a person has heart disease. This research implements the use of an algorithm called logistic regression, where the algorithm uses logistic functions to produce binary or zero and one as a classification determination. After the experiment is carried out with a logistic regression algorithm the results have different advantages over other methods based on the confusion matrix analysis model. In training data, logistic regression methods has the highest sensitivity value is 88.54% compared to other methods. In testing data, logistic regression method has the highest specificity value of 87.50% compared to other methods.

Keywords: Heart Disease, Cardiovascular Disease, Machine Learning, Logistic Regression

Abstrak – Penyakit Jantung atau disebut juga penyakit kardiovaskular merupakan salah satu penyakit berbahaya yang dapat menyebabkan kematian. Seiring berkembangnya teknologi dan peningkatan popularitas teknologi machine learning, teknologi machine learning tersebut dapat digunakan untuk membantu mendeteksi penyakit jantung dengan menggunakan data pasien. Terdapat berbagai jenis metode yang dapat digunakan untuk mendiagnosa apakah seseorang terkena penyakit jantung atau tidak. Penelitian ini mengimplementasikan penggunaan algoritma yaitu logistic regresi, dimana algoritma tersebut memakai fungsi logistik untuk menghasilkan binary atau nol dan satu sebagai penentuan klasifikasi. Setelah eksperimen dilakukan dengan algoritma logistik regresi memberikan hasil yang memiliki keunggulan yang berbeda-beda terhadap metode lainnya berdasarkan model analisa confusion matrix. Pada data training, metode logistik regresi mempunyai nilai sensitivity yang paling tinggi yaitu 88.54% dibanding metode lainnya. Pada data testing, metode logistik regresi mempunyai nilai kekhususan yang paling tinggi yaitu 87.50% dibanding metode lainnya.

Kata Kunci: *Penyakit Jantung, Penyakit Kardiovaskular, Machine Learning, Logistik Regresi*

PENDAHULUAN

Penyakit jantung atau dikenal juga sebagai penyakit kardiovaskular adalah semua penyakit yang terjadi akibat adanya gangguan fungsi jantung. Penyakit jantung merupakan hasil dari penumpukan plak di dalam arteri koroner, yang menghambat aliran darah ke jantung serta meningkatkan risiko serangan jantung dan komplikasi lainnya [1]. Penyakit jantung merupakan salah satu penyebab kematian tertinggi di Indonesia pada tahun 2017 menurut Kementerian Kesehatan Indonesia [2].

Tentunya dengan kemajuan teknologi angka tersebut dapat diminimalisir. Salah satu teknologi yang ada sekarang yaitu *machine learning* yang dapat digunakan untuk mendeteksi sebuah penyakit. Seperti penelitian yang pernah dilakukan untuk mendiagnosis penyakit Diabetes Melitus dengan menggunakan *Extreme Learning Machine* [3], maupun penelitian yang dilakukan untuk mendiagnosis penyakit jantung koroner dengan membandingkan algoritma C4.5 dan *Extreme Learning Machine* [4].

Machine learning sendiri bukanlah sebuah teknologi yang memiliki keakuratan 100% untuk melakukan analisa data dan mendapat kesimpulan berdasarkan analisa data tersebut. Namun keakuratan yang dihasilkan cukup efektif sehingga *machine learning* telah terbukti membantu di bidang kesehatan [5].

Ada banyak algoritma klasifikasi yang dapat dipakai di *machine learning*, tetapi dalam kasus ini penulis menggunakan algoritma *Logistic Regression*. *Logistic Regression* adalah analisis regresi yang tepat untuk dilakukan ketika variabel dependen adalah biner (dua

kemungkinan). *Logistic Regression* digunakan untuk menggambarkan data dan untuk menjelaskan hubungan antara satu variabel biner dependen dan satu atau lebih variabel independen nominal, ordinal, interval atau rasio tingkat. Penulis juga akan menggunakan *dataset* yang diambil dari *Kaggle* [6].

STUDI LITERATUR

Penyakit Jantung

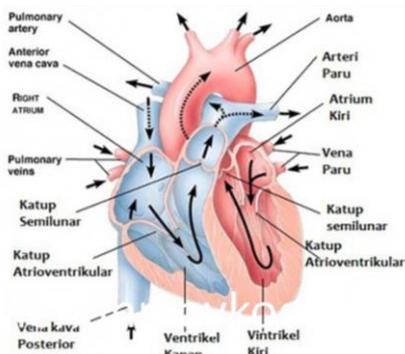
Penyakit jantung merupakan keadaan dimana jantung mengalami masalah atau gangguan. Bentuk gangguan jantung bermacam-macam, mulai dari pembuluh darah jantung, irama jantung, katup jantung, atau gangguan akibat bawaan lahir.

Anatomi jantung terdiri dari 4 ruangan, dua ruang bagian atas merupakan atrium (serambi) kiri dan kanan, sedangkan ruang bagian bawah merupakan ventrikel (bilik) kiri dan kanan. Antara ruang kanan dan kiri dipisahkan oleh otot yang berfungsi untuk mencegah tercampurnya darah yang miskin oksigen dengan yang kaya akan oksigen.

Fungsi jantung adalah untuk memompa darah yang kaya dengan oksigen ke seluruh organ tubuh, lalu darah yang miskin dengan oksigen dari organ kembali ke jantung, atrium kanan lalu diteruskan ke ventrikel kanan. Diantara atrium kanan dengan ventrikel kanan terdapat katup trikuspid yang berfungsi untuk mencegah darah kembali ke atrium kanan, lalu ketika ventrikel berkontraksi, maka darah akan dibawa melalui katup pulmonal dan arteri pulmonal ke paru paru untuk diisi dengan oksigen.

Setelah darah telah diisi oksigen oleh paru paru, maka akan dibawa ke atrium

kiri melalui vena pulmonal. Lalu atrium kiri akan berkontraksi dan darah yang kaya akan oksigen akan mengalir ke ventrikel kiri melalui katup mitral, dimana fungsi katup mitral adalah mencegah darah untuk kembali ke atrium kiri. Kemudian ventrikel kiri akan berkontraksi dan darah akan mengalir ke seluruh organ tubuh. Gambar 1 menunjukkan anatomi jantung yang terdiri dari 4 ruang serta penghubung antar ruang [7].



Gambar 1. Anatomi jantung

Machine Learning

Machine Learning merupakan sebuah metode berbasis komputer yang tidak perlu diatur dahulu oleh manusia dan dapat belajar dengan bantuan data dan akan semakin pintar seiring dengan banyaknya data yang telah diolah (belajar melalui pengalaman). Metode ini sering digunakan dalam menyelesaikan kasus klasifikasi dan *clustering* dan biasanya digunakan untuk menangani data dalam skala besar atau *big data*.

Di dalam *machine learning*, proses pengolahan datanya terbagi menjadi *data training* dan *data testing* dimana *data training* bertujuan untuk melatih metode atau algoritma yang digunakan dalam *machine learning* sedangkan *data testing* bertujuan untuk mengetahui tingkat kinerja dari algoritma yang telah dilatih tersebut ketika digunakan pada data yang

belum pernah ditemui saat melakukan proses *data mining*.

Machine learning secara umum jika dilihat dari teknik pembelajarannya terbagi menjadi *unsupervised learning* dan *supervised learning*. *Supervised learning* merupakan teknik di dalam *machine learning* dalam mengolah dataset yang telah memiliki label data untuk dipelajari sehingga label data tersebut mampu diidentifikasi oleh mesin saat melakukan proses klasifikasi maupun prediksi sedangkan *unsupervised learning* merupakan teknik di dalam *machine learning* yang memperbolehkan mesin itu sendiri dalam menarik kesimpulan atas informasi yang diperlukan dan biasanya dilakukan pada data tanpa *label* [8].

Logistik Regresi

Model logistik regresi adalah suatu model statistik yang digunakan untuk mengetahui pengaruh variabel prediktor (X) terhadap variabel respon (Y) dengan variabel responnya berupa data dikotomi yaitu bernilai 1 menyatakan bahwa variabel respon memiliki kriteria yang ditentukan dan 0 menyatakan bahwa variabel respon tidak memiliki kriteria yang ditentukan. Menurut Hosmer dan Lemeshow [9], persamaan model regresi logistik adalah:

$$\pi(x_i) = \frac{e^{\beta_0 + \beta_1 x_{1i}}}{1 + e^{\beta_0 + \beta_1 x_{1i}}} \quad (1)$$

Dengan fungsi logit $g(x_i)$ yaitu:

$$g(x_i) = \ln \left[\frac{\pi(x_i)}{1 - \pi(x_i)} \right] = \beta_0 + \beta_1 x_{1i} \quad (2)$$

Dari persamaan (1) dan (2) dapat disederhanakan menjadi:

$$\pi(x_i) = \frac{e^{g(x_i)}}{1 + e^{g(x_i)}} \quad (3)$$

Fungsi Sigmoid

Persamaan model yang digunakan untuk logistik regresi yang dapat disebut juga fungsi sigmoid adalah sebagai berikut

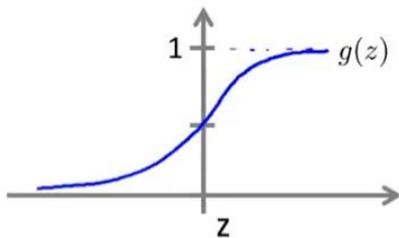
$$h_{\theta} = g(\theta^T x) \quad (4)$$

$$z = \theta^T x \quad (5)$$

$$g(z) = \frac{1}{1+e^{-z}} \quad (6)$$

$$h_{\theta}(x) = \frac{1}{1+e^{-\theta^T x}} \quad (7)$$

$h_{\theta}(x)$ akan memberikan hasil diantara 0 dan 1, contohnya $h_{\theta}(x)$ memberikan nilai 0.7 dimana memberikan hasil probabilitas 70% terhadap output dari 1 dan untuk probabilitas hasil 0 adalah 30%. Fungsi dengan kurva yang berbentuk huruf S. Untuk setiap nilai x yang dipetakan ke dalam interval 0 sampai 1 dinamakan fungsi sigmoid biner, sedangkan output yang memiliki rentang antara -1 sampai dengan 1 disebut sigmoid tan. Berikut gambar 2 adalah gambar dari persamaan sigmoid yang berbentuk huruf S.



Gambar 2. Fungsi Sigmoid

Misalnya, untuk memprediksi

- $y = 1$, maka nilai dari $h_{\theta}(x) \geq 0.5$, dimana nilai dari $z \geq 0$
- $y = 0$, maka nilai dari $h_{\theta}(x) < 0.5$, nilai dari $z < 0$,

Dari penjelasan di atas, dapat disimpulkan bahwa ketika memprediksi hasil dari sebuah persamaan, 0 ataupun 1, sama seperti memprediksi $y = 1$ ketika nilai dari $\theta^T x \geq 0$ dan sebaliknya memprediksi nilai $y = 0$, $\theta^T x < 0$.

METODOLOGI

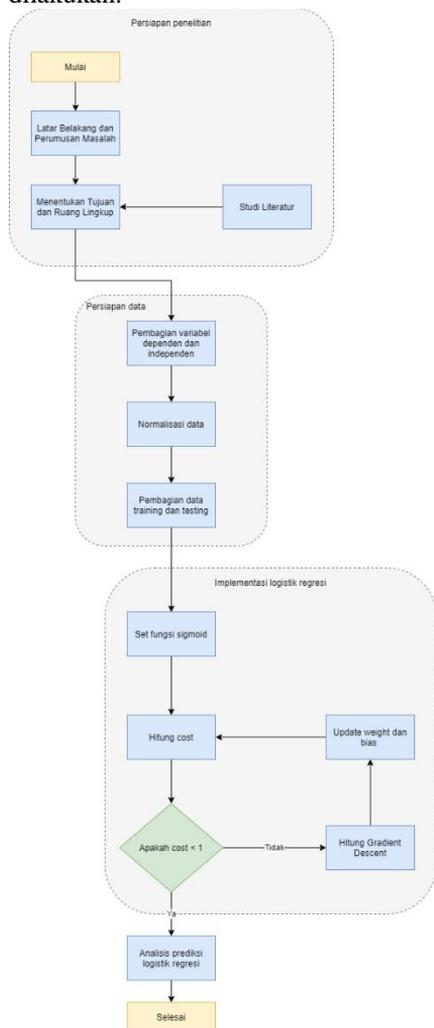
Tahapan Penelitian

Tahap awal dalam melakukan persiapan penelitian yaitu menentukan latar belakang dan tujuan penelitian serta mendefinisikan batasan dan ruang lingkup, serta studi literatur. Studi Literatur mempunyai bagian penting dalam pelaksanaan penelitian karena dapat membantu meningkatkan pemahaman akan komponen komponen yang terlibat dalam penelitian, yaitu cara kerja prediksi algoritma Logistik Regresi serta faktor faktor yang dapat mengakibatkan apakah seseorang memiliki penyakit jantung.

Tahapan kedua dalam langkah penelitian yaitu melakukan persiapan data. Dimana dalam persiapan data, data yang akan digunakan dilakukan pembagian variabel menjadi variabel dependen dan variabel independen. Dilanjutkan dengan normalisasi data agar data rentang antara 0 dan 1 untuk memudahkan perhitungan. Dan selanjutnya dilakukan pembagian data menjadi data *training* dan data *testing*

Tahapan ketiga yaitu implementasi algoritma Logistik Regresi untuk mendiagnosa apakah seseorang memiliki penyakit jantung atau tidak. Proses dari implementasi terdiri dari mencari *cost* atau bisa disebut juga sebagai *mean error* menggunakan *cost function* disebut juga *forward propagation*, lalu melihat apakah *cost* sudah sesuai dengan yang diinginkan, jika tidak, maka dapat menggunakan gradient descent atau bisa disebut juga backward propagation untuk menghitung bobot dan melakukan update terhadap bobot maupun *bias* pada variabelnya dan hitung ulang *cost*-nya menggunakan *cost function*. Jika sudah sesuai maka dapat dilanjutkan ke tahap berikutnya.

Dan langkah terakhir yaitu melakukan analisa terhadap hasil implementasi logistik regresi dan menarik kesimpulan serta memberikan saran dari hasil penelitian yang telah diperoleh. Analisa menggunakan confusion matrix untuk mengevaluasi keakuratan model logistik regresi. Gambar 3 menunjukkan flowchart proses dari penelitian yang dilakukan.



Gambar 3. Tahapan Penelitian

Metode Pengumpulan Data

Dataset didapatkan melalui repository dataset yang berada pada website Kaggle.com [6]. Kaggle merupakan

tempat kerkumpulan data scientist dimana tersedia banyak dataset yang dapat diambil dan tersedia banyak perlombaan yang berkaitan dengan Machine Learning.

Dataset yang digunakan berjudul Heart Disease UCI dimana dataset tersebut didapat dari Cleaveland Heart Disease Database dan dipublikasikan pada Kaggle. Dataset tersebut sudah digunakan oleh para peneliti machine learning saat ini, dimana dataset tersebut mengandung 76 atribusi yang memiliki pengaruh pada penyakit jantung yang diidap seseorang, tetapi berdasarkan eksperimen yang biasanya dijalankan, biasanya menggunakan 14 atribusi dan terdapat 303 data pasien seperti yang ditunjukkan pada tabel 1. Konklusi terdapatnya penyakit jantung pada kolom target dengan nilai 0 atau 1.

Tabel 1. Penjelasan Label pada Dataset

Singkatan Label	Makna Label
Age	Umur
Sex	Jenis Kelamin
CP (Chest Pain)	Rasa sakit dada
Trestbps	Tekanan darah saat istirahat (mmHg)
Chol	Kolesterol (mg/dl)
Fbs	Gula darah puasa (>120mg/dl) (1=ya;0=tidak)
Restecg	Hasil elektrographic saat istirahat
Thalac	Detak jantung maksimal
Exang	Latihan yang diinduksi angina (1=ya;0=tidak)
Oldpeak	Depresi yang diinduksi oleh latihan relatif
Slope	Kemiringan puncak ST Segmen
Ca	Jumlah pembuluh darah yang berwarna setelah diwarnai flourosopy
Thal	Tipe kerusakan pembuluh darah, 3=normal;6=cacat tetap;7=cacat sementara
Target	1 atau 0

Penelitian Dengan Logistik Regresi

Diagnosis penyakit jantung dengan logistik regresi menggunakan script

python sebagai *tool*-nya dan proses dari implementasi logistik regresi dapat dijelaskan sebagai berikut:

a. *Pembagian variabel dependen dan independen*

Pembagian variabel menjadi variabel independen yaitu kolom lain selain target dan variabel dependen yaitu target, adalah sebagai berikut, variable independen yaitu *Age, Sex, Chest Pain, Trestbps, Chol, Fbs, Restecg, Thalac, Exang, Oldpeak, Slope, Ca, Thal*, sedangkan variable dependen yaitu *Target*.

b. *Normalisasi data*

Data yang akan dimasukkan ke dalam logistik regresi dinormalisasi sehingga mempunyai nilai dengan rentang 0 dan 1. Berikut ini merupakan rumus yang digunakan untuk normalisasi data

$$X_{changed} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Dimana:

X_{min} = nilai minimum pada data set sebelum normalisasi

X_{max} = nilai maksimum pada data set sebelum normalisasi

$X_{changed}$ = nilai data setelah normalisasi

X = nilai data asli sebelum normalisasi

c. *Pembagian data training dan testing*

Data akan dibagi untuk *training* dan *testing*. Data sebesar 80 % dari *dataset* akan dipilih secara acak untuk dijadikan data *training* dan 20 % dari *dataset* akan digunakan sebagai *testing*.

d. *Menentukan fungsi*

e. *Fungsi sigmoid*

Untuk setiap nilai x yang dipetakan ke dalam interval 0

sampai 1 dinamakan fungsi sigmoid biner, sedangkan output yang memiliki rentang antara -1 sampai dengan 1 disebut sigmoid tan.

f. *Forward propagation*

Forward propagation digunakan untuk menghitung *cost function* dan y (*output*). *Cost* atau disebut juga *mean error*, mengukur perbedaan antara 2 variabel kontinu. Berikut merupakan persamaan dengan *forward propagation*.

$$\begin{aligned} J(\theta) &= -\frac{1}{m} \sum_{i=1}^m \left[y^{(i)} \log(h_{\theta}(x^{(i)})) \right. \\ &\quad \left. - (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})) \right] \end{aligned}$$

g. *Backward propagation*

Backward Propagation digunakan untuk menghitung *gradient descent*. *Gradient descent* merupakan algoritma optimisasi untuk meminimalkan beberapa fungsi secara iteratif. Penggunaan *gradient descent* digunakan untuk memperbaharui parameter bobot pada model regresi logistik. Gambar 3.3 menunjukkan proses *gradient descent* dimana terlihat *gradient* membantu mengarahkan *weight* ke titik optimal. Berikut merupakan persamaan dari *backward propagation*:

Repeat {

$$\theta_j := \theta_j - \frac{\alpha}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

}

h. *Menentukan learning rate*

Learning rate ditentukan menggunakan intuisi karena *learning rate* bersifat fleksibel, apabila

learning rate terlalu rendah maka akan memerlukan banyak waktu dan juga biaya tenaga *gpu* untuk mendapatkan *cost function* yang optimal. Sebaliknya apabila *cost function* terlalu tinggi, maka *cost function* tidak akan pernah mencapai titik optimal karena akan melewati *cost function* optimal tersebut [10].

i. Mencari *cost function* yang optimal

Dengan menggunakan *data training* dan *forward propagation* dan *backward propagation* dalam iterasi yang cukup banyak akan mendapatkan *cost function* yang optimal. Setelah mendapatkan *cost function* kita dapat menyimpulkan *weight* (beban) pada setiap variabel untuk dijadikan model pada *machine learning*.

j. *Testing model*

Setelah mendapatkan model, kita dapat melakukan *testing* dengan memasukkan data *testing* kedalam model dan mendapatkan hasilnya. Hasil tersebut dapat diolah dalam *confusion matrix* untuk mendapatkan analisa terhadap keakuratan dalam prediksi model tersebut.

HASIL DAN PEMBAHASAN

Hasil Pengumpulan Data

Berdasarkan pencarian data, diperoleh *dataset Heart Disease UCI* dimana *dataset* tersebut didapat dari *Cleveland Heart Disease Database* dan dipublikasikan pada *Kaggle*. *Dataset* tersebut sudah digunakan oleh para peneliti *Machine Learning* saat ini, dimana *dataset* tersebut mengandung 76 atribusi yang memiliki pengaruh pada penyakit jantung yang diidap seseorang, tetapi berdasarkan eksperimen yang dijalankan, menggunakan 14 atribusi dan terdapat 303 data pasien. Konklusi

terdapatnya penyakit jantung pada kolom *target* dengan nilai 0 atau 1. Gambar 4 menunjukkan informasi nilai *dataset* disediakan dengan format sebagai berikut:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
target	age	sex	cp	restecg	chol	fbs	restng	thalca	exang	oldpeak	slope	ca	thal	target
0	63	1	3	150	212	1	0	150	0	2.1	0	0	3	1
1	37	1	2	180	250	0	1	187	0	8.5	0	0	2	1
4	41	0	1	130	268	0	0	172	0	3.4	2	0	2	1
5	56	1	1	140	236	0	1	208	0	0.8	2	0	2	1
6	47	0	0	150	154	0	1	163	1	0.6	2	0	2	1
7	57	1	0	150	192	0	1	188	0	0.1	1	0	1	1
0	50	0	1	140	204	0	0	233	0	1.3	1	0	2	1
9	44	1	1	170	243	0	1	173	0	0	2	0	3	1
10	54	1	2	174	209	1	1	242	0	0.9	2	0	2	1
11	57	1	2	150	268	0	1	174	0	1.6	2	0	2	1
12	54	1	0	160	190	0	1	206	0	1.2	2	0	2	1
12	68	0	2	180	275	0	1	139	0	0.2	2	0	2	1
14	49	1	1	180	166	0	1	191	0	0.6	2	0	2	1
15	61	1	1	130	211	0	0	144	1	1.8	1	0	2	1
16	58	0	1	150	288	1	0	162	0	1	2	0	2	1
17	60	0	2	150	176	0	1	168	0	1.6	1	0	2	1
18	58	0	2	140	190	0	1	172	0	0	2	0	2	1
19	66	0	1	150	116	0	1	154	0	1.6	0	0	2	1
20	43	1	0	150	147	0	1	171	0	1.5	2	0	2	1
21	60	0	1	140	219	0	1	151	0	1.4	2	0	2	1
22	50	1	2	150	174	0	1	161	0	0.5	1	0	1	1
23	51	1	2	130	219	0	1	178	1	0.1	2	0	2	1
24	42	1	0	140	136	0	1	178	0	0	0	0	2	1
25	41	1	2	140	143	1	1	137	1	1	1	0	2	1
26	62	1	1	150	189	0	1	138	1	1.1	2	0	2	1
27	62	1	2	170	144	0	1	174	0	1.1	2	0	2	1

Gambar 4. *Dataset* Pasien Penyakit Jantung

Prediksi Dengan Logistik Regresi

Salah satu hal yang perlu ditentukan dalam prediksi menggunakan logistik regresi adalah jumlah *input*. Penelitian ini akan menggunakan 13 buah variabel *input*. Pada kolom 14, khusus kolom *target*, tidak termasuk pada variabel *input* melainkan hanya penanda apakah seseorang mengidap penyakit jantung atau tidak.

Sebelum melakukan penelitian dilakukan pembagian variabel menjadi variabel independen yaitu kolom selain *target* dan variable dependen yaitu *target*. Selanjutnya dilakukan normalisasi terlebih dahulu agar data memiliki rentang antara 0 dan 1.

	age	sex	cp	restecg	chol	fbs	restng	thalca	exang	oldpeak	slope	ca	thal
0	0.708333	1.0	1.000000	0.481132	0.244492	1.0	0.0	0.602063	0.0	0.370688	0.0	0.0	0.333333
1	0.166667	1.0	0.666667	0.339623	0.283105	0.0	0.5	0.885496	0.0	0.564516	0.0	0.0	0.666667
2	0.250000	0.0	0.333333	0.339623	0.178692	0.0	0.0	0.770962	0.0	0.225806	1.0	0.0	0.666667
3	0.962500	1.0	0.333333	0.245093	0.251142	0.0	0.5	0.816784	0.0	0.129632	1.0	0.0	0.666667
4	0.583333	0.0	0.000000	0.245093	0.520548	0.0	0.5	0.702290	1.0	0.069774	1.0	0.0	0.666667

Gambar 5. Sampel Data yang Dinormalisasi

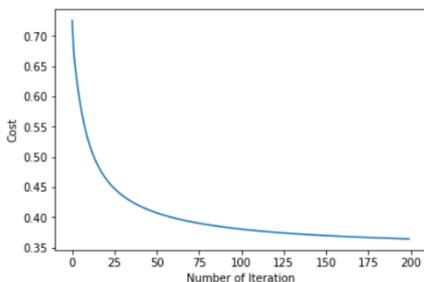
Tahap selanjutnya adalah membagi data *training* dan *testing* sebesar 80% dari *dataset* akan dipilih secara acak untuk dijadikan data *training* dan 20% dari *dataset* akan digunakan sebagai data *testing*. Data *training* yang sudah diacak akan dilakukan proses *transpose matrix*.

74	193	64	236	237	236	236	128	73	23	242	277	211	9	
pop	0.209827	0.177622	0.089497	0.178232	0.042322	0.020250	0.061497	0.239200	0.480222	0.042000	0.729747	0.082222	0.048887	0.042322
sex	0.100000	0.500000	0.200000	0.300000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
cp	0.000007	0.000007	0.000007	0.000000	0.333333	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.333333	0.000000	0.000007
trestbps	0.284151	0.460284	0.423262	0.263219	0.386028	0.377248	0.202453	0.191811	0.423262	0.268732	0.481132	0.263219	0.245233	0.528202
chol	0.100000	0.200000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000	0.100000

Gambar 6. Data Setelah Transpose

Implementasi Logistik Regresi

Setelah dilakukan iterasi sebesar 200 kali dan *learning rate* sebesar 1, maka pemilihan iterasi menggunakan intuisi dengan berdasarkan pada besaran penurunan *cost*, dimana penentuan sebesar perubahan *cost* minimal 0.005. Maka dari itu, bobot *weight* yang akan digunakan adalah pada iterasi ke-76 dikarenakan setelah iterasi ke-76, perubahan *cost* sudah tidak terlalu signifikan. Sebagai gambaran, perubahan *cost* dari iterasi ke-1 hingga iterasi ke-76 sebesar 0.33572448 sedangkan pada iterasi ke-76 hingga iterasi ke-152, perubahan *cost* hanya sebesar 0.0204791, dan terus terjadi penurunan *cost* hingga iterasi ke-200. Gambar 7 merupakan hasil pemetaan antara *cost* dengan iterasi.



Gambar 7. Grafik Pemetaan Cost dengan Iterasi

Testing Logistik Regresi

Berdasarkan bobot *output* yang didapat dari proses *training*, maka tahap selanjutnya adalah melakukan prediksi dengan algoritma logistik regresi. Data yang digunakan adalah data *testing* sebesar 20% dari total data sehingga didapatkan sebesar 62 data. Tabel 2 menunjukkan hasil bobot pada setiap variabel.

Tabel 2. Bobot *Output* Hasil *Training* Untuk Masing-Masing Variabel

Age	-0.34399865
Sex	-1.31237031
CP (Chest Pain)	1.7729785
Trestbps	-0.26222051
Chol	-0.0371197
Fbs	-0.169084
Restecg	0.38723275
Thalac	1.33400028
Exang	-1.1463211
Oldpeak	-1.04004649
Slope	1.15955381
Ca	-1.85961429
Thal	-0.67790772
Target	

Analisa Hasil Prediksi Logistik Regresi

Hasil analisa prediksi logistik regresi dapat dilakukan pada proses *training* dan proses *testing*. Keakuratan prediksi logistik regresi dapat dianalisa dari hasil prediksi *label target* seseorang mengidap penyakit jantung. Dari proses logistik regresi, digunakan metode *confusion matrix* dimana terdapat masing masing proses *training* dan *testing* seperti pada tabel 2.

Tabel 3. Hasil Proses Data *Training* & *Testing*

Proses	TP	TN	FP	FN
Training Regresi Logistik	116	86	25	15
Testing Regresi Logistik	31	21	6	1

Berdasarkan hasil dari tabel *confusion matrix* di atas, dapat dihitung *sensitivity*, *specificity*, *accuracy* dan *precision* pada masing masing proses *training* dan *testing* logistik regresi. Tabel 3 menunjukkan hasil analisa model prediksi menggunakan *confusion matrix* terhadap data *testing* dan data *training*.

Tabel 4. Hasil *Confusion Matrix* Data *Training* dan *Testing*

Proses	Spesifitas	Sensitivitas	Persisian	Training Accuracy
Training Regresi Logistik	85.14 %	88.54 %	82.26 %	83.47%
Testing Regresi Logistik	87.50 %	91.17 %	83.78 %	85.25%

Pada penelitian sebelumnya telah dilakukan perbandingan antara algoritma C.50, SVM, KNN, dan *neural network* untuk mendeteksi penyakit jantung pada *dataset* yang sama [11]. Berdasarkan hasil dari data *training*, terlihat bahwa algoritma KNN lebih unggul dibanding dengan algoritma lainnya dan algoritma logistik regresi lebih rendah tingkat akurasi, presisi dan spesifikasi dibanding algoritma yang lain.

Tabel 5. Perbandingan Algoritma pada Data Training

Algoritma	Spesifitas	Sensitivitas	Persisian	Akurasi
C.50	89.62 %	84.61 %	85.71 %	87.50 %
SVM	84.90 %	79.48 %	79.48 %	82.61 %
KNN	91.50 %	79.48 %	87.32 %	86.41 %
Neural Network	91.50 %	78.20 %	87.14 %	85.87 %
Logistik Regresi	85.14 %	88.54 %	82.26 %	83.47 %

Berdasarkan hasil dari data *testing*, terlihat bahwa algoritma C.50 lebih unggul dibanding dengan algoritma lainnya dan algoritma *neural network* lebih rendah tingkat akurasi, sensitivitas, presisi, dan spesifikasi dibanding dengan algoritma lain, sedangkan algoritma logistik regresi hanya unggul pada spesifikasi saja.

Tabel 6. Perbandingan Algoritma pada Data Testing

Algoritma	Spesifitas	Sensitivitas	Persisian	Akurasi
C.50	90.90 %	95.23 %	90.90 %	93.02 %
SVM	90.90 %	80.95 %	89.47 %	86.05 %
KNN	88.63 %	88.09 %	88.09 %	88.37 %

Neural Network	86.36 %	73.80 %	83.78 %	80.23 %
Logistik Regresi	87.50 %	91.17 %	83.78 %	85.25 %

KESIMPULAN

Dari hasil percobaan dan analisa dari prediksi penyakit jantung dengan menggunakan logistik regresi maka dapat diambil kesimpulan sebagai berikut:

- Tahapan dalam implementasi logistik regresi adalah menentukan variabel dependen dan independen, normalisasi data, pembagian data *training* dan *testing*, mendefinisikan fungsi *sigmoid*, *forward propagation* dan *backward propagation*, menentukan *learning rate* dan jumlah iterasi, menentukan *cost function* yang optimal dan melakukan *testing* pada data.
- Jumlah iterasi dan *learning rate* dalam penelitian menggunakan logistik regresi akan menentukan hasil akurasi penelitian. Dimana iterasi dan inisiasi *learning rate* menggunakan intuisi yang didasari pada perubahan *cost* pada setiap iterasi.
- Diagnosis penyakit jantung dengan menggunakan logistik regresi memiliki keunggulan yang berbeda beda terhadap metode lainnya pada model analisa *confusius matrix*.
- Semakin rendah *cost* dalam logistik regresi, semakin bagus hasil akurasi yang dihasilkan di dalam penelitian ini.

DAFTAR PUSTAKA

[1] "Informasi umum seputar penyakit jantung," Hellosehat, 2019. [Online]. Available:<https://hellosehat.com/pusat-kesehatan/serangan-jantung/informasi-umum-seputar-penyakit-jantung/>.

- [2] "Kementerian Kesehatan Republik Indonesia," 2017. [Online]. Available: <http://www.depkes.go.id/article/view/17073100005/penyakit-jantung-penyebab-kematian-tertinggi-kemenkes-ingatkan-cerdik-.html>.
- [3] J. J. Pangaribuan, "Mendiagnosis Penyakit Diabetes Melitus Dengan Menggunakan Metode Extreme Learning Machine," *Journal Information System Development (ISD)*, vol. 1, pp. 32-40, 2016.
- [4] J. J. Pangaribuan, C. Tedja and S. Wibowo, "Perbandingan Metode Algoritma C4.5 Dan Extreme Learning Machine Untuk Mendiagnosis Penyakit Jantung Koroner," *Jurnal Informatics Engineering Research and Technology (IERT)*, vol. 1, pp. 9-15, 2019.
- [5] Springer Nature Publishing, "Nature Biomedical Engineering," Nature Research, 2019. [Online]. Available: <https://www.nature.com/collections/zbkpvddmhm>.
- [6] Kaggle, "Kaggle," 2019. [Online]. Available: <https://www.kaggle.com/ronitf/heart-disease-uci>.
- [7] Fatmawati, Identifikasi Karakteristik Penyakit Jantung Koroner di Poliklinik RSU Bahteramas Provinsi Sulawesi Utara, 2017.
- [8] E. Retnoningsih and R. Pramudita, "Mengenal Machine Learning Dengan Teknik Supervised dan Unsupervised Learning Menggunakan Python," *Bina Insani ICT Journal*, vol. 7, pp. 156-165, 2020.
- [9] N. I. Mardini, L. Marlana and E. Azhar, "Regresi Logistik Pada Model Problem Based Learning Berbantu Software Cabri 3D," *Jurnal Mercumatika: Jurnal Penelitian Matematika dan Pendidikan Matematika*, vol. 4, pp. 64-70, 2019.
- [10] H. Zulkifli, "Understanding Learning Rates and How It Improves Performance in Deep Learning," *Towards Data Science*, 22 1 2018. [Online]. Available: <https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10>.
- [11] M. Abdar, S. R. N. Kalhori, T. Sutikno, M. I. I. Subroto and G. Arji, "Comparing Performance of Data Mining Algorithms in Prediction Heart Diseases," *International Journal of Electrical and Computer Engineering*, 2015.

FEATURE ADDITION TO THE COLOR HARMONIZATION APPLICATION IN TWO DIMENSIONAL IMAGE

Albert Antonio¹⁾, Irene A. Lazarusli²⁾, Samuel Lukas^{3*)}, Petrus Widjaja⁴⁾

¹⁾Informatics, Faculty of Computer Science, Universitas Pelita Harapan
Email : albertantonio88@gmail.com

²⁾Informatics, Faculty of Computer Science, Universitas Pelita Harapan
Email : irene.lazarusli@uph.edu

³⁾Informatics, Faculty of Computer Science, Universitas Pelita Harapan
Email : samuel.lukas@uph.edu

⁴⁾Mathematics, Faculty of Science and Technology, Universitas Pelita Harapan
Email : petrus.widjaja@uph.edu

**Penulis Korespondensi*

Abstract – In the era of digital technology, the demand for good visual communication increases. It deals with how to harmonize the usage of colors on that visual communication. It is called as color harmony. Color harmony is a series of combinations of colors that are balanced and aesthetically for visual perception. This paper discusses on how to add some additional features to a color harmony application so that it can be more easily used and understood for the general public. Additional features include manual harmonization operation, color palette recommendation using the K-means clustering algorithm, multithreading and GUI. This paper also measures the performance and expert judgment of the level of color harmony in the application output. The experiment data are 5 sample images with various resolution sizes of 640x480, 800x600, 960x720, 1024x768, 1280x960, 1440x1080, 1600x1200, and 1920x1440. The fastest proses time for the best harmonic scheme is 201 seconds for resolution 640x480 and the longest time is 1350 seconds for resolution 1600x1200. The rate of expert judgment in harmonious samples is 9.07 out of 12.00 and in non-harmonized samples is 6.02 out of 12.00.

Keywords: *K-means clustering, Color Harmony, multithreading, GUI*

Abstrak – Di era teknologi digital, permintaan akan komunikasi visual yang baik semakin meningkat. Ini berkaitan dengan bagaimana menyelaraskan penggunaan warna pada komunikasi visual itu. Ini disebut harmoni warna. Harmoni warna merupakan rangkaian kombinasi warna yang seimbang dan estetis untuk persepsi visual. Terdapat sebuah aplikasi yang dikembangkan oleh Yang (2018) yang dapat digunakan untuk merancang komunikasi visual yang baik berdasarkan harmoni warna. Makalah ini membahas tentang cara menambahkan beberapa fitur tambahan pada aplikasi tersebut agar lebih mudah digunakan dan dipahami oleh masyarakat umum. Fitur tambahan termasuk operasi harmonisasi manual, rekomendasi palet

Diterima <24032021>, Revisi <23062021>, Diterima untuk publikasi <26072021>.

Copyright © 2021 Published by PSDKU Sistem Informasi UPH Kampus Medan,
ISSN : 2528-5114

warna menggunakan algoritma clustering *K-means*, *multithreading* dan *GUI*. Makalah ini juga mengukur performa dan penilaian ahli dari tingkat harmoni warna pada keluaran aplikasi. Data percobaan adalah 5 citra sampel dengan berbagai ukuran resolusi 640x480, 800x600, 960x720, 1024x768, 1280x960, 1440x1080, 1600x1200, dan 1920x1440. Waktu proses tercepat untuk skema harmonisasi terbaik adalah 201 detik untuk resolusi 640x480 dan waktu terlama 1350 detik untuk resolusi 1600x1200. Tingkat penilaian ahli pada sampel harmonis adalah 9,07 dari 12,00 dan pada sampel non-harmonis adalah 6,02 dari 12,00.

Kata Kunci: *Clustering K-means, Harmonisasi Warna, multithreading, GUI*

PENDAHULUAN

Perancangan komunikasi visual yang baik melibatkan aspek keserasian warna. Aspek ini sangat penting karena memicu respons emosional para pembacanya. Respon emosional sangat mempengaruhi keadaan psikologis pembaca sehingga mereka dapat mengalami emosi yang menenangkan atau membangkitkan. [1] [2]. Suatu aplikasi digital menyelaraskan komposisi warna dalam gambar telah dikembangkan Yang [3] berdasarkan teori harmonisasi warna oleh Cohen-Or [4]. Aplikasi mencari skema harmonik terbaik dari jenis templat harmonik dan orientasi templat tersebut dengan menentukan skema harmonik terbaik yaitu mencari skor harmonik terendah. Metode harmonisasi warna ini berkisar pada template harmonik yang dibuat oleh Tokumaru et. al [5].

Aplikasi yang dikembangkan oleh Yang [3] terbukti berfungsi tetapi aplikasi menyediakan sedikit atau tidak ada interaksi user. Aplikasi tidak dapat memenuhi keinginan pengguna. Untuk mengatasi masalah tersebut, perlu dikembangkan

- 1) Penyempurnakan tampilan aplikasi harmonisasi warna agar mudah dipahami dan dioperasikan
- 2) Evaluasi penilaian subjektif pakar tentang tingkat keserasian warna atas keluaran aplikasi

- 3) Kinerja aplikasi harmonisasi warna dalam menangani input gambar dengan berbagai resolusi

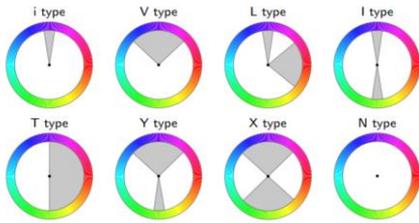
Makalah ini juga membahas tentang implementasi *GUI* dan kemampuan *multithreading* untuk memberikan interaksi kepada pengguna. Beberapa fitur telah ditambahkan bersama *GUI* seperti operasi harmonisasi manual dan rekomendasi palet warna. Fitur-fitur tersebut diimplementasikan dalam modul tambahan menggunakan Python dan beberapa pustaka.

FUNDAMENTAL TEORI

Harmonisasi Warna

Berdasarkan Granville [6], harmoni warna adalah “Kemajuan visual yang mulus melalui ruang warna yang memberikan hasil yang harmonis, bukan hanya karena warna-warna tersebut memiliki hubungan harmonis yang unik satu sama lain tetapi juga *chroma* yang sangat identik. Pada dasarnya harmoni mengandung arti keseimbangan yang dibuktikan dengan perilaku fisiologis manusia [7].

Warna yang harmonis ditentukan oleh posisi relatifnya dalam ruang warna HSV pada roda warna Gambar 1 oleh [4].



Gambar 1. 8 tipe roda *Heu Harmonic*

Proses Harmonisasi Warna

Proses harmonisasi warna terdiri dari tiga proses yang berbeda yaitu pemetaan *histogram hue*, pencarian skema harmonisa terbaik dan penerapan skema harmonis.

Pemetaan *hue histogram* menggunakan histogram lingkaran *hue* karena memiliki karakteristik yang mirip dengan color wheel dan template harmonik dalam mempertahankan sifat periodik atribut *hue* [8]. Ruang warna HSV dipilih sebagai ruang warna karena pemodelan hubungan antar warna yang mirip dengan sel fotoreseptor di mata manusia [9].

Sebagian besar gambar di televisi, layar dan model proyektor berwarna menggunakan ruang warna RGB. Oleh karena itu, harus diubah menjadi ruang warna HSV. Berdasarkan Serban [10], konversi RGB ke ruang warna HSV ditunjukkan pada persamaan (1) sd (7).

$$R' = \frac{R}{255}, G' = \frac{G}{255}, B' = \frac{B}{255} \quad (1)$$

$$m_{max} = \max(R', G', B') \quad (2)$$

$$m_{min} = \min(R', G', B') \quad (3)$$

$$\Delta = m_{max} - m_{min} \quad (4)$$

$$H = \begin{cases} \text{undefined, if } \Delta = 0 \\ \frac{G-B}{\Delta}, & \text{if } m_{max} = R \\ \frac{B-R}{\Delta} + 2, & \text{if } m_{max} = G \\ \frac{R-G}{\Delta} + 4, & \text{if } m_{max} = B \end{cases} \quad (5)$$

$$V = m_{max} \quad (6)$$

$$S = \begin{cases} 0, & \text{if } V = 0 \\ \frac{\Delta}{V}, & \text{otherwise} \end{cases} \quad (7)$$

Pencarian skema harmonis terbaik dari suatu gambar X berdasarkan Cohen-Or [4] dilakukan dengan pertama mencari type harmonic heu yang mana dari gambar 1 yang paling mendekati heu histogram gambar X. Kedua menentukan berapa nilai α dari type *hue* harmonis terbaik untuk gambar X. Ketiga melakukan perubahan nilai heu dari gambar X untuk mendapatkan model type Hue Harmonis terbaik dari gambar X itu

Penerapan skema harmonis setelah Langkah pencarian skema harmonis terbaik perlu dilakukan beberapa penyempurnaan untuk menghindari beberapa efek samping. Teknik yang digunakan dengan mengoptimalkan segmentasi biner [4] dimana proses segmentasinya menggunakan optimasi pemotongan graph [11].

K-means Color Clustering

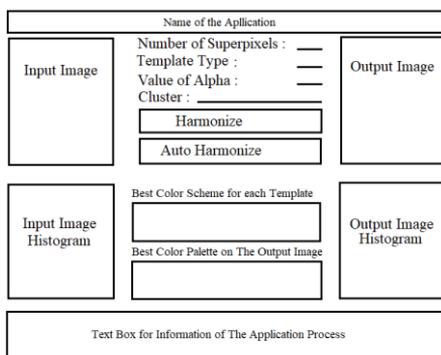
Salah satu Algoritma yang dipakai untuk mengelompokkan sejumlah data series menjadi k *cluster*. K titik sentroid ditempatkan secara acak dan kemudian data yang paling dekat dengan sentroid akan diidentifikasi sebagai bagian dari *cluster*. Setelah semua data teridentifikasi, selanjutnya titik *centroid* k akan ditempatkan di titik tengah berdasarkan lokasi *cluster*. Proses ini akan diulang sampai tidak ada lagi data yang dikelompokkan ke dalam *cluster* lain [12].

Titik *centroid* k ini akan menentukan jumlah warna pada palet warna yang akan dihasilkan. Nilai k menentukan hasil palet warna yang akan digunakan. Jika jumlah k terlalu sedikit, maka beberapa warna tidak dapat mewakili sebagian besar komposisi warna pada

gambar. Sebaliknya jika jumlah k terlalu banyak, maka ada warna yang tidak signifikan. Nilai $k \in \{3,7\}$ berfungsi dengan baik. Nilai $k = 5$ sering digunakan sebagai nilai standar [13].

PERANCANGAN PENELITIAN

Rancangan GUI aplikasi diperlihatkan pada Gambar 2. Input citra ditampilkan dan juga heu histogramnya. Kemudian user bisa melakukan harmonisasi warna manual dengan memberikan masukan ke empat *parameter* yaitu jumlah *superpixel*, *type template*, nilai *alpha* dan juga *cluster* atau hanya dengan menekan tombol *Harmonize*. Tombol *Auto Harmonize* ditekan untuk menghasilkan harmonisasi secara otomatis



Gambar 2. GUI of Color Harmonization App.

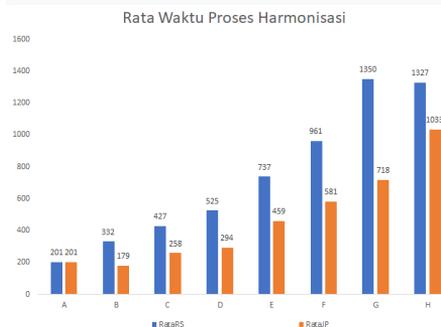
Hasil harmonisasi warnanya ditampilkan pada *Output Image* beserta *hue histogram*. Informasi mengenai *best color schema* untuk template terpilih, diperlihatkan dalam bentuk text box demikian juga *best color palette* dari *output* citra. *Text box* paling bawah memberikan informasi ke pemakai, proses apa yang sedang dilakukan oleh sistem.

DATA DAN PEMBAHASAN

Pengukuran penelitian dilakukan dengan mengukur waktu yang dibutuhkan untuk menemukan skema harmoni terbaik pada input gambar dua dimensi berdasarkan besar kecilnya resolusi gambar input.

Aplikasi diukur dengan 5 sampel gambar yang berbeda dengan 8 ukuran resolusi gambar yang bervariasi mulai dari 640x480, 800x600, 960x720, 1024x768, 1280x960, 1440x1080, 1600x1200, dan 1920x1440 pixels yang diberi label dari A sd H, Hasil rata-rata waktu untuk setiap resolusi diperlihatkan pada Gambar 3.

Grafik batang RataRS adalah waktu rata-rata mengharmonisasikan citra sesuai resolusi sedangkan grafik batang RataJP adalah waktu rata-rata yang dibutuhkan untuk mengharmonisasikan citra sesuai dengan jumlah *pixel*. Sangat logis makin banyak pixel yang diproses maka waktu yang dibutuhkan makin lama namun jika berdasarkan resolusinya maka waktu yang dibutuhkan untuk resolusi 1600x1200 relatif tidak berbeda dengan 1920x1440. Namun untuk resolusi yang lebih rendah jelas membutuhkan waktu yang lebih cepat.



Gambar 3. Rata-rata waktu vs Resolusi

Penelitian juga melibatkan *expert judgment*. Sepuluh tingkat keserasian citra diberikan ke para ahli untuk diminta penilaian atas hasil dari aplikasi. Mereka diberikan dua *set data* pengujian. Set pertama adalah data harmonisasi dari aplikasi atas *input* citra yang memang sudah harmonis sedangkan set kedua data harmonisasi dari data *input* citra yang tidak harmonis.

Input citra baik yang harmonis maupun tidak diambil berdasarkan jumlah download dan like pada platform resource grafis www.freepik.com. Citra juga diuji dengan penerapan harmonisasi warna oleh Yang [3] berdasarkan teori harmonisasi warna oleh Cohen-Or [4]. *Output input* citra harmonis tidak menghasilkan perubahan yang berarti. Namun yang input citra yang tidak harmonis mengalami perubahan nilai *hue* yang signifikan pada komposisi warna utamanya. Gambar 4 memperlihatkan 10 input citra yang sudah harmonis, sedangkan gambar 5 memperlihatkan input citra yang tidak harmonis

No	Nama gambar	Catatan	Rekomendasi harmonisasi warna	Tipe template	Orisinal template (pixel)	Sumber
1				X	154	https://www.pinterest.com/pinterest/golf-templates/
2				X	33	https://www.pinterest.com/pinterest/back-to-school-sale-templates/
3				X	44	https://www.pinterest.com/pinterest/fun-run-templates/
4				Y	210	https://www.pinterest.com/pinterest/5k-run-walk-templates/
5				X	25	https://www.pinterest.com/pinterest/kolor-run-templates/
6				X	193	https://www.pinterest.com/pinterest/kolor-run-templates/
7				X	161	https://www.pinterest.com/pinterest/trivia-night-templates/
8				X	167	https://www.pinterest.com/pinterest/fun-day-camp-templates/
9				X	227	https://www.pinterest.com/pinterest/game-night-templates/
10				Y	7	https://www.pinterest.com/pinterest/zorba-design-templates/

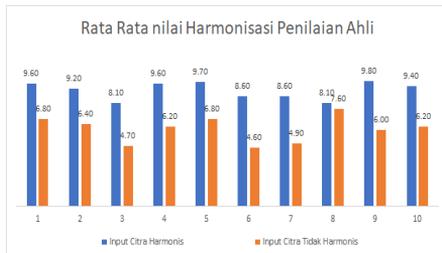
Gambar 4. *Input* citra harmonis dan *Output*

Terlihat bahwa aplikasi dapat mengenali citra yang serasi karena tidak banyak perbedaan yang signifikan antara sampel dan keluaran citra. Selain itu, tipe *template X* merupakan tipe *template* yang paling sering digunakan. Gambar 5 memperlihatkan aplikasi menghasilkan citra keluaran dengan komposisi warna perbedaan yang signifikan karena komposisi warna sampel citra tidak harmonis.

No.	Nama gambar	Catatan	Rekomendasi harmonisasi warna	Tipe template	Orisinal template (pixel)	Sumber
1				X	154	https://www.pinterest.com/pinterest/golf-templates/
2				X	33	https://www.pinterest.com/pinterest/back-to-school-sale-templates/
3				X	44	https://www.pinterest.com/pinterest/fun-run-templates/
4				Y	210	https://www.pinterest.com/pinterest/5k-run-walk-templates/
5				X	25	https://www.pinterest.com/pinterest/kolor-run-templates/
6				X	193	https://www.pinterest.com/pinterest/kolor-run-templates/
7				X	161	https://www.pinterest.com/pinterest/trivia-night-templates/
8				X	167	https://www.pinterest.com/pinterest/fun-day-camp-templates/
9				X	227	https://www.pinterest.com/pinterest/game-night-templates/
10				X	71	https://www.pinterest.com/pinterest/trivia-night-templates/

Gambar 5. *Input* Citra tidak harmonis dan *Output*

Sepuluh ahli memberikan penilaian tingkat harmonisasi citra *input* dibandingkan citra hasil harmonisasi atas ke-10 input citra harmonis dan ke-10 *input* citra tidak harmonis. Hasil penilaian diperlihatkan pada gambar 6.



Gambar 6. Rata-rata nilai harmonisasi dari para ahli.

Nilai rata-rata harmonis luaran dari aplikasi untuk *input* citra yang harmonis dan tidak harmonis adalah 9.07 dan 6.02 dari skala 12. Nilai 10 adalah nilai luaran harmonis aplikasi sama dengan nilai harmonis input citra. Ini artinya luaran aplikasi tidak berbeda dengan harmonis citra *input* yang sudah harmoni. Namun untuk citra input yang tidak harmonis meskipun luarannya rata-rata dinilai para ahli adalah 6.02 ini berarti aplikasi sudah mencoba memberikan luaran yang lebih baik dari harmonisasi citra *input*.

Untuk citra *input* yang sudah harmoni rata-rata terendah harmoni *output* adalah 8.1. Ini kembali mengindikasikan bahwa citra *input* memang harmoninya sudah cukup baik. Sebaliknya rata-rata harmoni citra dari *input* citra tak harmoni nilainya adalah 4.6. Ini mengindikasikan citra *input* memang kurang harmoni. Sehingga luarannya juga kurang harmoni.

KESIMPULAN

Berbagai perbaikan telah dilakukan untuk aplikasi harmonisasi warna agar mudah dipahami dan dioperasikan oleh masyarakat umum. Hasil survei mengindikasikan bahwa aplikasi sudah

berjalan dengan baik dan dapat meningkatkan harmonisasi citra masukan tentu sesuai dengan pemakainya. Kecepatan melakukan harmonisasi citra input dalam ratusan detik dan paling lama untuk resolusi 1600x1200 selama 1350 detik atau sekitar 22.5 menit.

DAFTAR PUSTAKA

- [1] Lasquite, Mydee, "What Makes Good Design? Basic Elements and Principles". *Easy WebContent, Inc.* September 30. <https://visme.co/blog/elements-principles-good-design/>. (2015)
- [2] Stone, N. J., & English, A. J., "Task Type, Posters, and Workspace Color on Mood, Satisfaction, and Performance". *Journal of Environmental Psychology* (1998): p. 175-185.
- [3] Yang, C. A., "Color Harmonization." *GitHub* (2018), www.github.com/tartarskunk/Color-Harmonization.
- [4] Cohen-Or, Daniel, et al., "Color Harmonization". *ACM SIGGRAPH 2006 Papers* (2006): p. 624-630.
- [5] Tokumaru, M., Murataka, N., & Imanishi, S., "Color design support system considering color harmony". *IEEE World Congress on Computational Intelligence* (2002). *IEEE International Conference on Fuzzy Systems* (2002).
- [6] Granville WC. "Color harmony: what is it?". *Color Research and Application* (1987): p. 196-201.
- [7] Itten, J. "The Elements of Color". *Van Nostrand Reinhold Company* (1961): p. 19.
- [8] Din-Chang, T., & Cheng-Tan, T. "Circular histogram thresholding for color image segmentation". *Proceedings of 3rd International Conference on Document Analysis and Recognition* (1995).
- [9] Vadivel, A., Sural, S., & Majumdar, A. K. (2005). "Human color perception in the HSV space and its application in histogram generation for image retrieval". *Color Imaging X: Processing, Hardcopy, and Applications*.
- [10] Serban, H. (n.d.). "Converting RGB to HSV". *Simon Fraser University*.

- <https://mattlockyer.github.io/iat455/documents/rgb-hsv.pdf>.
- [11] Boykov, Y., & Jolly, M.-P. “Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images”. *Proceedings of ICCV* (2001): p-105-112.
 - [12] Celebi, M. Emre. “Improving the Performance of k-Means for Color Quantization.” *Image and Vision Computing*, vol. 29, no. 4, 2011, pp. 260–271., doi:10.1016/j.imavis.2010.10.002.
 - [13] Chang, H., Fried, O., Liu, Y., DiVerdi, S. & Finkelstein, A., “Palette-based Photo Recoloring”. *ACM Transactions on Graphics (Proc. SIGGRAPH)* (2015).
 - [14] Van den Bergh, M., Boix, X., Roig, G., de Capitani, B., & Van Gool, L. (2012). “SEEDS: Superpixels Extracted via Energy-Driven Sampling”. *Lecture Notes in Computer Science*, 13–26.

APLIKASI MARKETPLACE BERBASIS WEBSITE UNTUK PENCARIAN JASA MUSISI

Alya M. Amalia¹⁾, Andree E. Widjaja^{2*)}, Hery³⁾, Calandra A. Haryani⁴⁾

¹Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: alyamutiara20@gmail.com¹⁾

²Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: andree.widjaja@uph.edu²⁾

³Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: hery.fik@uph.edu³⁾

⁴Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: calandra.haryani@uph.edu⁴⁾

**Penulis Korespondensi*

Abstract – Marketplace is an online market that can be used by sellers and buyers to do any kind of transactions. There is barely any website in Indonesia that has a marketplace specifically for musicians, especially a marketplace for seeking services in the music field. The process of seeking jobs for musicians, besides meeting in person with potential clients, musicians often use social media or search engine as a normal way to seek jobs, however this process is inefficient to musicians as well as to clients who seek for musicians as it is still taking a lot of time to only search for the jobs they desired, especially if the searching process is done in a platform that is not specialised on the music field. Therefore, the purpose of this current research is to develop an application that can help musicians to seek for jobs and help users to seek for musicians they wish to hire. The application development is carried out using HTML, PHP, and MySQL Database. The application has been successfully tested using black box testing method, and thus ready to use.

Keywords: marketplace, musician, information systems

Abstrak – Marketplace merupakan sebuah pasar yang bersifat online yang dapat dipakai oleh penjual maupun pembeli untuk melakukan berbagai jenis transaksi. Namun sayangnya, masih sedikit aplikasi berbasis website di Indonesia yang berbentuk marketplace khusus untuk mencari jasa musisi. Umumnya, proses pencarian pekerjaan untuk jasa musisi adalah selain bertemu langsung dengan klien potensial, musisi biasanya juga menggunakan media sosial dan search engine. Akan tetapi, proses tersebut dapat dikatakan masih belum efisien dikarenakan baik musisi maupun pengguna melakukan pencarian pekerjaan di platform yang tidak spesifik bergerak di dalam bidang marketplace pencarian jasa musisi. Maka dari itu, tujuan dari penelitian ini adalah untuk mengembangkan aplikasi marketplace berbasis website yang secara khusus dapat membantu serta memudahkan musisi untuk mencari pekerjaan dan membantu pengguna untuk mencari musisi yang dibutuhkan. Pengembangan aplikasi marketplace dilakukan dengan menggunakan HTML, PHP, dan Database MySQL. Hasil pengujian berdasarkan metode Blackbox testing menunjukkan bahwa aplikasi yang dikembangkan ini telah lulus uji dan siap dipakai.

Diterima <30032021>, Revisi <25052021>, Diterima untuk publikasi <26072021>.
Copyright © 2021 Published by PSDKU Sistem Informasi UPH Kampus Medan,
ISSN : 2528-5114

Kata Kunci: pengembangan aplikasi website, marketplace, jasa musisi, musik

PENDAHULUAN

Sistem Informasi dewasa ini memiliki peranan yang sangat penting untuk membantu kebutuhan manusia. Dengan adanya Sistem Informasi, pengolahan data menjadi informasi dan proses bisnis yang ada dapat dikerjakan dengan lebih efektif dan efisien. Secara umum, Sistem Informasi dapat digunakan di dalam kehidupan sehari-hari untuk menyelesaikan berbagai masalah, terutama melalui teknologi Internet. Saat ini, Internet sudah banyak sekali digunakan oleh banyak pengguna di Indonesia. Menurut Kementerian Komunikasi dan Informatika Republik Indonesia, pengguna *Internet* di Indonesia sudah mencapai 102 juta orang, hal tersebut membuat Indonesia menjadi peringkat keenam dengan pengguna *Internet* terbanyak di dunia [1].

Pada saat ini, sudah cukup banyak *marketplace online* berbasis *Internet*, mulai dari *marketplace* yang berfokuskan pada penjualan berbagai macam produk atau barang hingga jasa, seperti Tokopedia, Bukalapak, Shopee, dan lain sebagainya. Namun, berdasarkan observasi yang telah dilakukan, masih sedikit *website* di Indonesia yang berfokuskan pada *marketplace* dalam pencarian jasa di bidang musik, yaitu terutama pencarian jasa musisi. Biasanya, proses pencarian pekerjaan untuk seorang musisi dilakukan melalui audisi ke tempat-tempat yang sedang membutuhkan penampilan (*musical performance*). Selain itu, proses pencarian musisi untuk orang-orang yang sedang membutuhkan penampilan adalah dengan menanyakan kepada orang-orang yang sudah dikenal atau dengan mencari musisi-musisi yang tersedia menggunakan mesin pencari *online* (*search engine*). Misalnya, musisi

mencari pekerjaan dengan cara mencari di *Google* atau *browsing* di sosial media, contohnya di *Facebook*, *Instagram*, dan *Twitter*. Melalui berbagai sosial media tersebut, umumnya musisi berusaha mencari di berbagai *post* yang sudah diunggah oleh pengguna-pengguna lain. Berdasarkan observasi yang sudah dilakukan, memang sudah ada *marketplace* untuk mencari jasa di Indonesia. Akan tetapi, pencarian jasa di *website* yang dimaksud adalah masih bersifat umum, tidak secara khusus untuk mencari jasa musisi dan pengguna yang sedang mencari musisi untuk di-*hire*. Hal tersebut tentunya akan lebih membutuhkan banyak waktu sehingga proses pencarian jasa musisi menjadi tidak efisien.

Berangkat dari permasalahan di atas, tujuan utama dari penelitian ini adalah untuk merancang dan mengembangkan aplikasi *marketplace* berbasis *website* yang dapat secara spesifik digunakan untuk pencarian jasa musisi. Diharapkan aplikasi *marketplace* berbasis *website* yang dikembangkan ini mampu memfasilitasi musisi-musisi baru maupun lama untuk memperlihatkan bakatnya dan sekaligus juga sebagai kesempatan untuk mendapatkan pekerjaan. Selain itu, aplikasi *marketplace* ini juga dapat memfasilitasi para pengguna (klien) yang sedang membutuhkan penampilan dengan mencari musisi untuk di-*hire*.

Adapun batasan-batasan aplikasi yang dikembangkan adalah sebagai berikut:

- 1) Aplikasi dirancang dan dikembangkan berbasis *website*.
- 2) Pengguna dapat *register* dan *login*.

- 3) Pengguna dapat membagikan informasi, mengunggah foto dan video tentang musik yang mereka kuasai.
- 4) Tidak ada fitur transaksi pembayaran.
- 5) Pengguna dapat mengirim pesan kepada pengguna lain di dalam aplikasi.

METODE PENELITIAN

Dalam penelitian ini, ada dua metode penelitian yang dilakukan, yaitu pengumpulan data (observasi) untuk menganalisis sistem saat ini, dan metode *prototyping*, sebagai metode pengembangan sistem aplikasi.

Metode Observasi

Hasil dari observasi serta wawancara yang telah dilakukan oleh penulis adalah sebagai berikut:

Analisis Sistem Saat Ini

Musisi yang sedang mencari pekerjaan di bidang musik dan tidak memiliki *manager* untuk mengelola pekerjaannya biasanya mencari pekerjaan (*job*) melalui sosial media seperti *Instagram*, *Twitter* dan *Google*, dan hasil dari pencarian tersebut kebanyakan lokasinya berada di wilayah yang jauh. Musisi juga banyak menanyakan kepada orang lain bila ada acara yang sedang melakukan *hiring* sebagai cara untuk mendapatkan pekerjaan dan juga ditawarkan oleh orang yang sudah dikenal untuk bermain musik di acaranya. Musisi yang sudah memiliki label musik sendiri biasanya sudah memiliki banyak koneksi dari label tersebut dan sudah dicarikan oleh staf atau *manager* dari musisi tersebut.

Salah satu *website* yang mungkin dapat digunakan oleh musisi untuk mencari pekerjaan adalah *mussy.co* (<https://www.facebook.com/mussyco/>).

Di dalam *website* tersebut, musisi dapat melakukan pencarian untuk pekerjaan tertentu, dapat menampilkan video dari musisi. Namun kekurangan utama dari *website* tersebut adalah untuk berkomunikasi dengan sesama, pengguna harus menggunakan aplikasi lain di luar *website* tersebut. Selain itu, dalam menunjukkan hasil karyanya, musisi harus menggunakan sosial media sebagai *platform* untuk mengunggah hasil karya mereka, seperti *YouTube*, *Instagram*, *Twitter* dan *SoundCloud*. Kemudian musisi akan *share link* tersebut dan mengirimkannya agar dapat dilihat oleh banyak orang. Proses tersebut dapat dikatakan kurang efisien karena masih mengandalkan faktor eksternal.

Kendala Sistem Saat Ini

Menimbang situasi dan kondisi yang ada dalam sistem saat ini, ada dua (2) kendala yang teridentifikasi, yaitu:

- 1) Proses yang harus dilakukan oleh musisi dalam mencari *job* di bidang musik seperti mencari musisi, menghubungi klien dan mengikuti audisi. Proses ini relatif panjang dan lama sehingga membuat musisi mengalami kesulitan dalam mencari *job*. Kesulitan yang sama juga dialami dalam proses pencarian musisi bagi klien.
- 2) Pencarian pekerjaan pada sosial media seperti *Facebook*, *Instagram* dan *Twitter* atau *search engine* lainnya tidak efisien dikarenakan belum adanya pencarian yang spesifik untuk *job* dan musisi sehingga musisi dan klien harus melakukan usaha yang lebih dalam mencari *job* di sosial media.

Metode Prototyping

Metode *prototyping* adalah metode pengembangan aplikasi yang dilakukan secara berulang (*iterative*) dalam berbagai versi, yang kemudian akan disempurnakan secara terus menerus sampai aplikasi final dapat terpenuhi [2]. Secara umum, metode *prototyping* terdiri dari empat (4) tahap, yaitu:

- 1) *Analysis of Requirements*, dalam tahapan ini dilakukan analisa dan identifikasi dari kebutuhan dan persyaratan pengguna.
- 2) *Development of Prototype*, dalam tahapan ini dilakukan pengembangan dari prototipe yang sudah dibuat.
- 3) *User Testing and Feedback*, mengimplementasikan aplikasi yang sudah dibuat untuk dicoba oleh pengguna yang dapat memberikan *feedback*. Jika dari *feedback* masih ada yang harus diperbaiki, maka akan dilakukan *prototype revision or refinement*, tahapan ini akan terus berulang sampai produk diterima oleh pengguna (*users acceptance*).
- 4) *Final Product Release*, tahapan ini dilakukan jika sistem yang dikembangkan sudah diterima oleh pengguna secara final dan sudah tidak membutuhkan perubahan apapun.

Studi Pustaka

Beberapa studi pustaka dan teori yang terkait akan dielaborasi dalam sub-bab berikut ini.

E-Commerce dan Social Commerce

E-commerce adalah proses penjualan, pembelian, pengiriman atau pertukaran sebuah produk, jasa ataupun informasi yang dilakukan melalui sebuah jaringan komputer, khususnya Internet [3]. *E-commerce* didefinisikan sebagai semua transaksi yang dilakukan secara

elektronik antara suatu organisasi dengan organisasi atau pihak lainnya [4]. Sedangkan, *social commerce* adalah pengembangan lanjutan dari *e-commerce* yang didefinisikan sebagai kegiatan dan transaksi *e-commerce* melalui media sosial yang mendukung interaksi sosial dan kontribusi konten pengguna [5]. *Social commerce* dapat diartikan juga sebagai kombinasi dari kegiatan komersial dan aktivitas sosial [6].

E-Marketplace

E-Marketplace adalah sebuah *marketplace* yang bersifat *online* berupa *website* yang di mana pembeli dan penjual dapat bertemu seperti *marketplace* fisik [7]. *E-marketplace* adalah pusat pasar berbasis *website* yang dapat digunakan oleh pembeli dan penjual untuk melakukan perdagangan dan kegiatan bisnis elektronik lainnya [8][9]. Fungsi dari sebuah *e-marketplace* sama dengan pasar fisik pada umumnya, namun *e-marketplace* jauh lebih efisien karena menyediakan informasi dan berbagai layanan yang cepat dan lancar secara online [10][11].

HTML dan PHP

Hypertext Markup Language (HTML) adalah bahasa pemrograman web yang digunakan untuk mengembangkan suatu halaman situs *website*. Sementara itu, *Hypertext Preprocessor* yang disingkat sebagai PHP adalah sebuah bahasa pemrograman yang sangat banyak digunakan dalam pengembangan aplikasi berbasis *website*. PHP dapat digunakan di dalam *web server* dan dapat dipakai secara *gratis* atau *open source* [12].

Relational Database Management System (RDBMS)

Relational Database Management System (RDBMS) adalah sebuah sistem manajemen basis data yang berstruktur yang terdiri dari *tables*, *fields* dan *records* [12]. RDBMS merupakan sebuah

perangkat lunak yang dipakai untuk membuat dan mengelola sebuah *database*. Untuk mengakses database, RDBMS memakai bahasa khusus yang disebut dengan *Structured Query Language* (SQL) [12].

Penelitian Terdahulu Terkait

Beberapa penelitian terdahulu terkait pengembangan aplikasi *marketplace* yang diaplikasikan di dalam berbagai konteks telah dipublikasikan di literatur. Salah satu di antaranya adalah pengembangan aplikasi *marketplace* yang fokus pada proses pemesanan jasa konfeksi [13]. Aplikasi yang dikembangkan tersebut diklaim mampu mengatasi kesulitan yang terjadi dalam proses pemesanan jasa konfeksi seperti masalah harga dan *progress* pemantauan pengerjaan. Selain itu, penelitian lain mengembangkan aplikasi reservasi dan manajemen *marketplace* yang berfokus pada *homestay* di kota Pekanbaru [14]. Aplikasi yang dinamakan *Homie* tersebut memiliki fungsi utama untuk mempertemukan pemilik *property* persewaan dengan pelanggan penyewa tempat tinggal. Adapun pengembangan aplikasi *marketplace* lainnya dalam konteks yang sangat unik, yaitu sistem *marketplace* bonsai Indonesia [15]. Aplikasi yang dikembangkan tersebut diklaim mampu mengatasi permasalahan yang kerap terjadi dalam Perkumpulan Penggemar Bonsai Indonesia (PPBI), khususnya terkait permasalahan jual-beli bonsai. Berbeda dengan penelitian-penelitian sebelumnya, fokus dari penelitian ini adalah pengembangan aplikasi *marketplace* berbasis website dalam konteks musik, secara khusus untuk pencarian jasa musisi.

HASIL DAN PEMBAHASAN

Use Case Diagram

Use case diagram menjelaskan tentang fungsi dari sistem yang dibuat [16][17][18]. *Use case* dapat menangkap interaksi sistem dengan pengguna. *Use Case diagram* dapat dilihat pada **Appendix 1** di akhir halaman artikel (setelah referensi). Aktor-aktor yang ada di dalam *use case* adalah sebagai berikut:

- 1) **Musisi:** Musisi merupakan pengguna yang dapat mencari pekerjaan di dalam sistem.
- 2) **Klien:** Klien merupakan pengguna yang dapat mencari musisi di dalam sistem.
- 3) **Admin:** Admin merupakan pengelola data yang ada di dalam sistem.

Table Relationship Diagram dan Class Diagram

Table Relationship Diagram atau TRD menggambarkan hubungan yang ada di dalam basis data [16]. Di dalam *table relationship diagram* terdapat atribut *primary key*, yang berguna untuk mengimplementasikan relasi dari tabel dan *foreign key* yang berguna untuk merferensikan relasi antara satu tabel dengan tabel yang lain. Pada artikel ini, *Table Relationship Diagram* ditempatkan pada **Appendix 2**. Sedangkan, *class diagram* yang merupakan model statis yang menunjukkan hubungan antar kelas di dalam sistem [16] dapat dilihat di **Appendix 3** di bagian akhir halaman artikel.

Perancangan User Interface

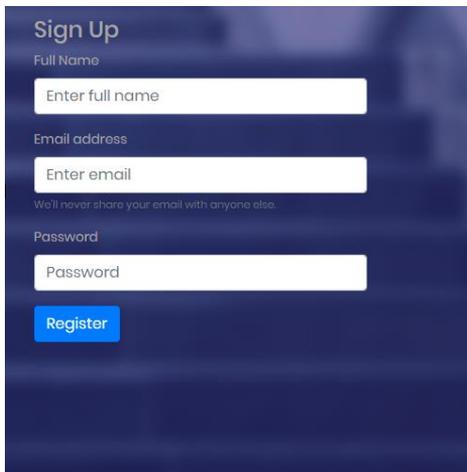
Bagian ini akan membahas secara singkat mengenai pengembangan tampilan antarmuka dari aplikasi *marketplace* berbasis *website* untuk pencarian jasa musisi. Pertama-tama untuk mendapatkan akses

penyedia pada aplikasi, musisi harus melakukan *login* terlebih dahulu. Sama seperti musisi, klien juga harus melakukan *login* terlebih dahulu. Untuk *login*, pengguna (baik musisi maupun klien) harus melakukan *register* yang dapat dilakukan di halaman utama *website*. Pengguna dapat melakukan registrasi dan *login* sebagaimana yang ditunjukkan pada Gambar 1 di bawah ini.



Gambar 1. Tampilan Halaman Utama

Pada halaman utama terdapat *textbox* yang harus diisi untuk melakukan *login* dan registrasi.



Gambar 2. Tampilan Halaman *Sign Up*

Selanjutnya, gambar 2 di atas menunjukkan tampilan halaman *sign up*. Pada tampilan awal *website* ada dua pilihan, yaitu *sign in* atau *sign up*. Jika ingin mendaftar, pengguna harus mengisi nama lengkap, alamat *email*

dan *password*. Setelah registrasi pengguna berhasil, data pengguna akan secara otomatis dimasukkan ke dalam *database*.



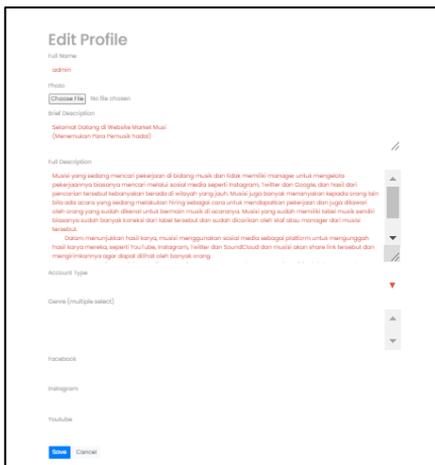
Gambar 3. Tampilan Halaman *Login*

Gambar 3 di atas adalah halaman *login* untuk masuk ke halaman selanjutnya atau halaman pengguna masing-masing. Untuk masuk ke halaman ini, pengguna baru harus melakukan registrasi atau membuat akun terlebih dahulu. Jika pengguna telah membuat akun, pengguna dapat mengisi alamat *email* yang sudah dibuat sebelumnya dan masukkan *password* yang sudah dibuat untuk *login* agar bisa masuk ke dalam halaman selanjutnya.



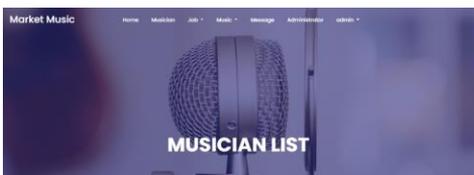
Gambar 4. Tampilan Halaman Utama Pengguna

Kemudian, gambar 4 di atas menampilkan halaman utama pada bagian akun masing-masing pengguna. Halaman ini terdiri dari beberapa menu seperti *Home*, *Music*, dan *Message*. Di beberapa menu ada beberapa bagian yang memiliki *sub-menu*, misalnya di bagian *Job* terdapat *Add Job* dan *Add List*, serta di bagian *music* terdapat *Add New* dan *List*, serta di bagian *Profile* terdiri dari bagian *edit profile*, *media* dan *logout*.



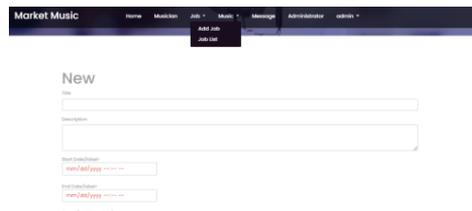
Gambar 5. Tampilan Edit Profile

Gambar 5 di atas menampilkan halaman *edit profile*. Pada bagian *edit profile* ini terdapat keterangan nama pembuatnya, sedangkan pada bagian *upload photo*, pengguna dapat mengunggah foto profil. Sementara itu, *Brief Description* dan *Full Description* dapat diisi oleh pengguna untuk menjelaskan tentang diri sendiri atau tentang musik yang dikuasai. Bagian *account type* terhubung dengan akun pengguna. Selain itu ada juga keterangan mengenai *genre* dari musisi yang memudahkan agar infomasinya lebih terarah kepada pengguna yang sesuai dengan *genre* musiknya masing-masing. Kolom Facebook, Instagram dan Youtube dapat digunakan untuk memberikan informasi tambahan mengenai informasi sosial media yang dimiliki pengguna.



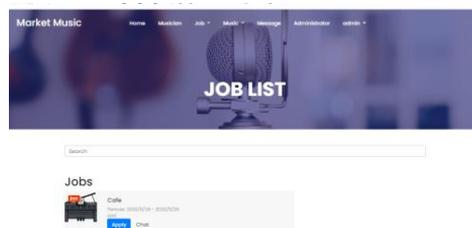
Gambar 6. Tampilan Halaman Musician List

Gambar 6 di atas menampilkan halaman *Musician List* yang berisikan mengenai daftar para musisi yang tergabung di dalam aplikasi *marketplace* ini.



Gambar 7. Tampilan Halaman Add Job

Gambar 7 di atas menampilkan halaman menu *Job*. Menu *job* memiliki dua (2) sub menu yaitu, *add job* dan *job list*. Pada menu *add job*, *user* dapat menambahkan *job* untuk musisi yang ada di dalam aplikasi. Pengguna dapat memasukan nama dan deskripsi dari pekerjaan tersebut dan mengisi tanggal yang sudah ditentukan oleh pengguna.



Gambar 8. Tampilan Halaman Job List

Gambar 8 di atas adalah halaman *Job List* yang berisikan tentang pekerjaan yang sudah ditambahkan pada halaman *Add Job* oleh pengguna. Musisi dapat mengambil pekerjaan yang dibutuhkan sesuai dengan *job* yang tersedia. Pada bagian *job* ada pilihan untuk *apply* dan *chat* yang dapat digunakan untuk berkomunikasi secara langsung dengan pengguna terkait *job* yang ditawarkan tersebut.



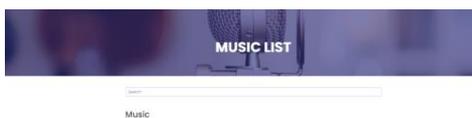
Gambar 8. Tampilan Halaman *Apply Job*

Gambar 8 di atas menampilkan halaman *apply job*. Jika musisi telah menemukan pekerjaan yang cocok, musisi dapat langsung *apply job* tersebut.



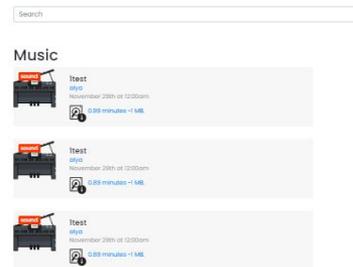
Gambar 9. Tampilan Halaman *Add Music*

Gambar 9 di atas menampilkan halaman *add music*. Pada halaman ini musisi dapat menambahkan video dari musisi tersebut dengan cara memasukkan *link* YouTube.



Gambar 10. Tampilan Halaman *My Music List*

Pengguna dapat melihat daftar musik yang sudah ditambahkan ke dalam aplikasi *marketplace* oleh pengguna. Gambar 10 menampilkan halaman *my music list*. Halaman ini menampilkan daftar musik yang sudah ditambahkan oleh pengguna.



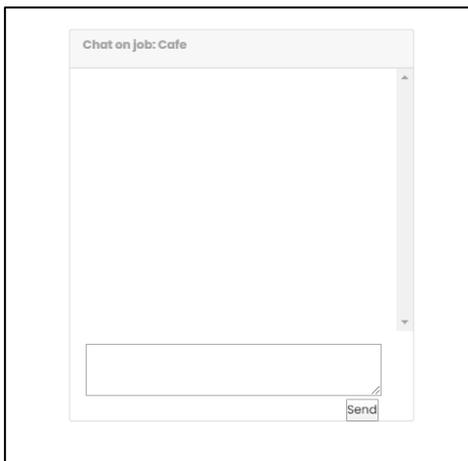
Gambar 11. Tampilan Halaman *Music List*

Gambar 11 di atas menampilkan halaman *music list* yang berisikan tentang semua musik dan video yang telah ditambahkan oleh seluruh pengguna ke dalam aplikasi *marketplace* ini.



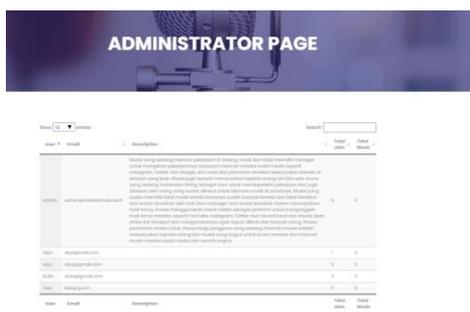
Gambar 12. Tampilan Halaman *Messages*

Gambar 12 di halaman sebelah kiri bawah menampilkan halaman *Messages* yang merupakan halaman untuk melihat dan mencari pesan yang telah dikirim oleh pengguna lain.



Gambar 13. Tampilan Halaman Chat

Gambar 13 di atas menampilkan halaman chat yang merupakan halaman untuk melakukan chatting di dalam aplikasi marketplace. Pengguna dapat menggunakan fitur ini untuk berkomunikasi dengan pengguna lain, khususnya jika ingin menanyakan tentang job lebih lanjut.



Gambar 14. Tampilan Halaman Administrasi

Gambar 14 di atas menampilkan halaman Administrator yang berisikan tentang nama user, email dan penjelasan serta informasi mengenai total pekerjaan dan total para pemusik yang sudah bergabung di dalam aplikasi marketplace ini.



Gambar 15. Tampilan Halaman Admin

Gambar 15 di atas adalah halaman Admin yang berisikan tentang bagian untuk edit profile, serta sub media yang berisikan bagian untuk upload file video dan logout untuk keluar dari akun admin.

Hasil Pengujian

Pengujian aplikasi marketplace berbasis website untuk pencarian jasa musisi ini telah dilakukan dengan menggunakan metode Black Box Testing. Pengujian bertujuan untuk mengetahui apakah aplikasi ini sudah layak dan semua fungsinya sudah berjalan dengan baik dan benar. Metode black box testing berfokus pada hasil fungsional dari sistem yang telah dirancang, yakni apakah sistem tersebut sudah sesuai dengan fungsi yang telah ditentukan sebelumnya.

Terdapat istilah-istilah yang digunakan di dalam dokumentasi pengujian untuk menunjukkan hasil dari pengujian sistem usulan yang telah dilakukan, istilah yang dimaksud adalah sebagai berikut:

- 1) **P = Passed**
 Nilai P menunjukkan bahwa kondisi dan fungsi yang diujikan menghasilkan output sesuai dengan proses perencanaan dan berfungsi dengan baik.
- 2) **F = Failed**
 Nilai F menunjukkan bahwa kondisi dan fungsi yang diujikan belum berfungsi dengan baik sesuai dengan apa yang direncanakan.

Adapun berdasarkan hasil pengujian yang sudah dilakukan, semua fungsi yang diujikan mendapatkan nilai **P**. Dengan kata lain semua fungsi yang dimiliki oleh aplikasi *marketplace* berbasis website untuk pencarian jasa musisi ini berjalan dengan sangat baik dan sesuai dengan ekspektasi. Dua (2) contoh hasil tabel pengujian dapat dilihat di **Appendix 4** di bagian akhir halaman artikel ini.

KESIMPULAN DAN SARAN

Aplikasi *marketplace* berbasis *website* untuk pencarian jasa musisi telah secara khusus dikembangkan untuk mengatasi berbagai kendala yang kerap dialami oleh para musisi dan klien pencari jasa musisi. Aplikasi *marketplace* usulan ini tentunya dapat mempermudah musisi untuk mencari *job* dan selain itu juga dapat mempermudah klien untuk mencari musisi yang cocok dengan ekspektasi atau kriteria yang diinginkan. Aplikasi ini juga telah didesain sedemikian rupa agar musisi lebih mudah untuk memperlihatkan (mendemonstrasikan) talentanya melalui video yang langsung dapat dilihat oleh klien, sehingga meningkatkan potensi musisi untuk mendapatkan *job*. Aplikasi ini juga dapat membantu musisi dalam mencari *job* tanpa harus berpergian (karena semua proses dapat dilakukan secara *online*).

Aplikasi *marketplace* berbasis *website* ini telah lulus uji coba pada tahapan *testing* berdasarkan metode pengujian *black box*. Hasil akhir dari aplikasi telah dirancang dan dikembangkan adalah sebagai berikut:

- 1) Mempermudah musisi untuk mencari pekerjaan dan mempermudah klien untuk mencari musisi.
- 2) Menghemat waktu dalam mencari *job* atau mencari musisi karena dengan adanya aplikasi ini, pengguna dapat mencari *job*

dan musisi secara lebih mudah dan cepat.

- 3) Membantu musisi dan klien untuk berkomunikasi (disediakan fitur *chat*) lebih cepat dengan menggunakan aplikasi ini.

Aplikasi yang telah dirancang dan dikembangkan ini tentunya masih belum sempurna. Aplikasi ini kemungkinan masih memiliki peluang untuk dikembangkan dengan lebih baik lagi sesuai dengan kebutuhan pengguna di masa yang akan datang. Adapun beberapa saran yang dapat diusulkan oleh penulis pada pengembangan aplikasi *marketplace* berbasis website untuk pencarian jasa musisi adalah sebagai berikut:

- 1) Mengembangkan aplikasi lebih lanjut dalam bentuk aplikasi *mobile* agar dapat diakses kapan saja dan di mana saja melalui perangkat *smartphone*.
- 2) Menambahkan fitur transaksi pembayaran agar musisi dan klien dapat melakukan *deal* dan pembayaran di dalam aplikasi. Transaksi pembayaran dapat dilakukan melalui berbagai metode, seperti bank transfer, OVO, GOPAY, Shopee dan lain-lain.
- 3) Mengembangkan fitur *chatting* lebih lanjut agar pengguna dapat mengirim dan menerima pesan secara *real time*.

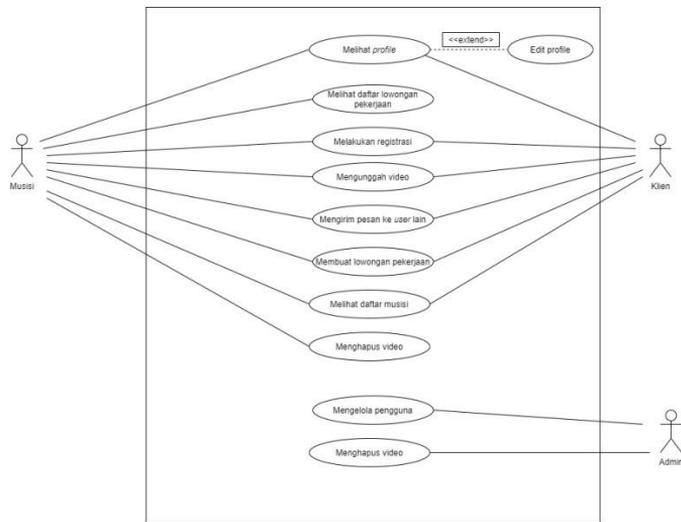
DAFTAR PUSTAKA

- [1] Kementerian Komunikasi dan Informatika, Pengguna Internet Indonesia Nomor Enam Dunia. Diakses pada: 23 Maret 2021 2017. [Online]. URL: <https://kominform.go.id/content/detail/4286/pengguna-internet->

- indonesia-nomor-enam-dunia/0/sorotan_media.
- [2] P. Isaias dan T. Issa, *High Level Models and Methodologies for Information Systems*, New York: Springer-Verlag, 2015.
- [3] R. K. Rainer Jr. dan C. G. Cegielski, *Introduction to Information Systems*, 4th Edition. Singapore: John Wiley & Sons, 2013.
- [4] D. Chaffey, *Digital Business and E-Commerce Management: Strategy, Implementation and Practice*, 6th Edition. London: Pearson Education, 2016.
- [5] T.-P. Liang dan E. Turban, *Introduction to the Special Issue Social Commerce: A Research Framework for Social Commerce*, *International Journal of Electronic Commerce*, Vol. 16, No. 2, hal. 5-13, 2011.
- [6] R. Boardman, M. Blazquez, C. E. Henninger, dan D. Ryding, *Social Commerce: Consumer Behavior in Online Environment*. California: Palgrave Macmillan, 2019.
- [7] M. Kütz, *Introduction to E-Commerce: Combining Business and Information Technology*, 1st Edition. Bookboon, 2016.
- [8] R. K. Rainer, C. G. Cegielski, I. Spletstoesser-Hogeterp, dan C. Sanchez-Rodriguez, *Introduction to Information Systems: Supporting and Transforming Business*, 3rd Edition. Hoboken, NJ: John Wiley & Sons, 2013.
- [9] I. Lee, *Trends in E-Business, E-Services, and E-Commerce: Impact of Technology on Goods, Services, and Business Transactions*. Pennsylvania: IGI Global, 2014.
- [10] E. Turban, J. Whiteside, D. King, dan J. Outland, *Introduction to Electronic Commerce and Social Commerce*. Upper Saddle River, NJ: Springer, 2017.
- [11] Z. Radovitsky, *Business Models for E-Commerce*. California: Cognella Academic Publishing, 2015.
- [12] S. Suehring and J. Valade, *PHP, MySQL, JavaScript & HTML5 All-in-One for Dummies*. Hoboken: John Wiley & Sons, Inc., 2013.
- [13] M.I. Kurniawan, B. Priyambadha, dan A.A. Soebroto, *Pengembangan Marketplace untuk Jasa Konfeksi*, *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol. 03, No. 8, hal. 7391-7398, 2019.
- [14] H.Y. Ardi, B. Priyambadha, dan A.A. Soebroto, *Pengembangan Aplikasi Reservasi dan Manajemen Marketplace Homestay Berbasis Web (Studi Kasus: Kota Pekanbaru)*, *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol. 03, No. 9, hal. 8914-8923, 2019.
- [15] J.K. Ruwanda, N. Santoso, dan A. Rachmadi, *Pengembangan Sistem Marketplace Bonsai Indonesia Berbasis Web*, *Jurnal*

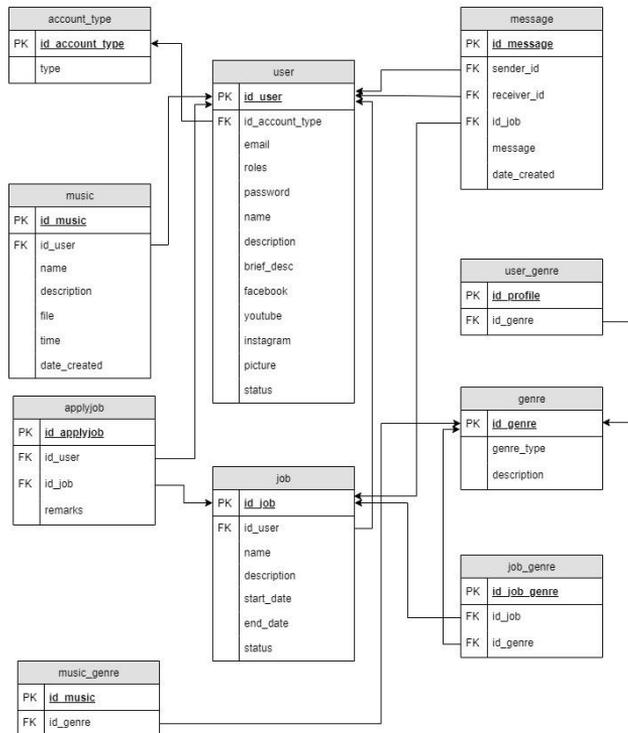
- Pengembangan Teknologi
Informasi dan Ilmu Komputer,
Vol. 04, No. 1, hal. 330-339,
2020.
- [16] A. Dennis, B. H. Wixom, dan R.
M. Roth, System Analysis and
Design, 5th edition. Hoboken,
NJ: John Wiley & Sons, 2012.
- [17] T. Clark and A. Evans,
Foundations of the Unified
Modeling Language,
Proceedings of the 2nd BCS-
FACS Conference on Northern
Formal Methods, July 1997.
- [18] J. Osis and U. Donins,
Topological UML Modeling.
Amsterdam: Elsevier, 2017.

Appendix 1



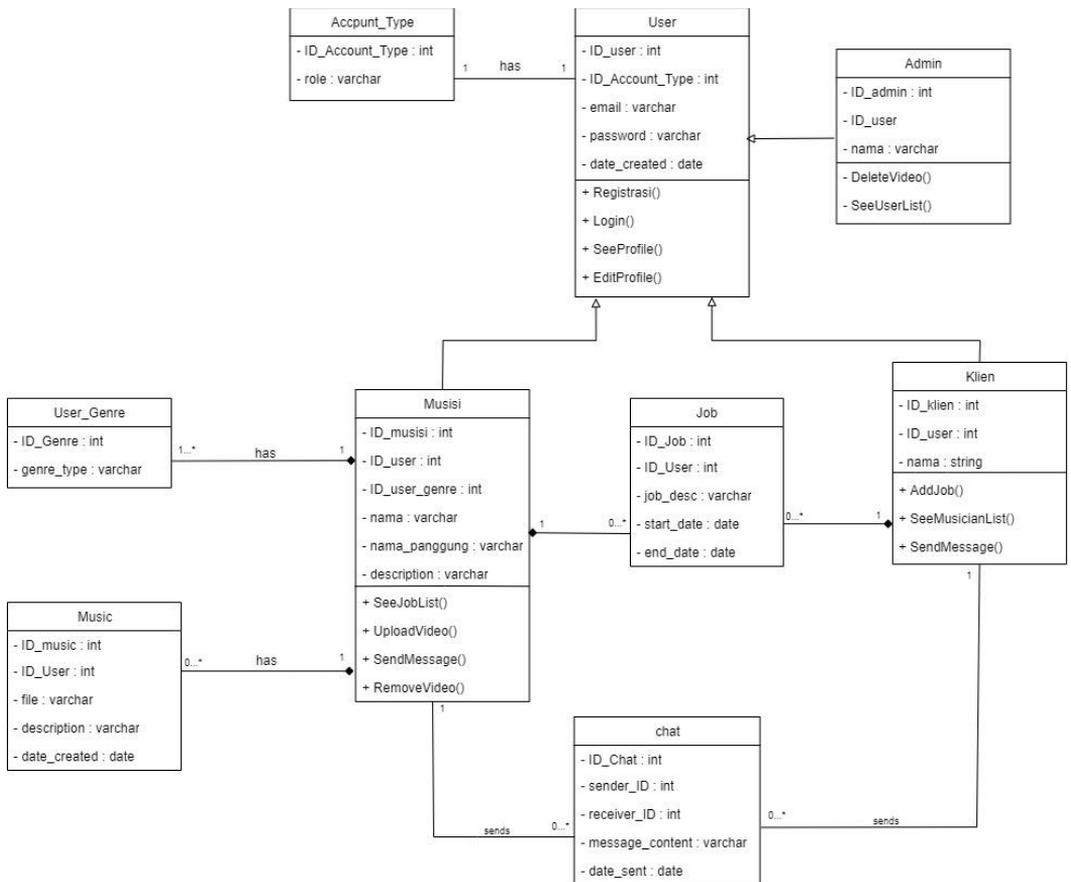
Gambar. Use Case Diagram Sistem Usulan

Appendix 2



Gambar. Table Relationship Diagram

Appendix 3



Gambar. Class Diagram

Appendix 4

Role: Pengguna

1. Melihat daftar lowongan pekerjaan

Scenario Setup 1:

1. Pengguna telah melakukan *login* ke dalam aplikasi.
2. Pengguna memilih menu *job list* atau daftar pekerjaan di dalam aplikasi.

Tabel. Test Case melihat daftar lowongan pekerjaan

<i>Test Number: 1</i>				
<i>Test Subject: Pengguna</i>				
<i>Function: Melihat daftar lowongan pekerjaan</i>				
<i>Step Description</i>	<i>Test Steps</i>	<i>Expected Results</i>	<i>Result (P/F)</i>	<i>Description Results</i>
Melihat daftar lowongan <i>job</i>	<ol style="list-style-type: none"> 1) Membuka <i>web</i> 2) Memilih menu <i>Job List</i> 3) Melihat daftar lowongan pekerjaan yang sudah dimasukan oleh pengguna lain 	Sistem menampilkan halaman <i>Job List</i> .	P	<i>Job List</i> atau daftar pekerjaan dapat ditampilkan.

Role: Pengguna

1. Menambahkan lowongan pekerjaan

Scenario Setup 4:

1. Pengguna telah melakukan *login* ke dalam aplikasi.
2. Pengguna telah memilih menu *Add Job* di dalam aplikasi.

Tabel. Test Case menambahkan lowongan pekerjaan

<i>Test Number: 4</i>				
<i>Test Subject: Pengguna</i>				
<i>Function: Menambahkan lowongan <i>job</i></i>				
<i>Step Description</i>	<i>Test Steps</i>	<i>Expected Results</i>	<i>Result (P/F)</i>	<i>Description Results</i>
Menambahkan lowongan <i>job</i>	<ol style="list-style-type: none"> 1) Memilih menu <i>Add Job</i> di dalam aplikasi 2) Mengisi deskripsi pekerjaan yang ingin dibuat 3) Menyimpan data 	Muncul notifikasi bahwa lowongan pekerjaan telah ditambahkan dan tersimpan dalam <i>database</i> .	P	Menampilkan lowongan <i>job</i> di daftar lowongan pekerjaan di dalam aplikasi.

IDENTIFIKASI DATA DRIFTING PADA APLIKASI INTERNET OF THINGS (IoT)

Alessandro Luiz Kartika¹⁾, I Made Murwantara²⁾, Pujianto Yugopuspito^{3*)}

¹⁾Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: ak70029@student.uph.edu¹⁾

²⁾Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: made.murwantara@uph.edu²⁾

³⁾Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: yugopuspito@uph.edu³⁾

**Penulis Korespondensi*

Abstract – Internet of Things has become a high demand technology which also generated massive data that streaming into their collector in a real-time manner. However, the stability of streaming data become crucial as it will influence the analytical processing at the edge. In order to reduce the mismatch of data streaming, this work utilizing concept drift to identify when the stream has begin to inconsistent in certain criteria. To this we do the classification using Stochastics Gradient Descent (SGD). To identify the data drifting, we adopted Adaptive Windowing and Page-Hinkley method. Our result shows that Adaptive Windowing (ADWIN) is more sensitive and outperform the Page-Hinkley method using our dataset.

Keywords: *Internet of Things, Concept Drift, SGD Classifier, Adaptive Windowing (ADWIN), Page-Hinkley*

Abstrak – Internet of Things telah menjadi teknologi yang sangat dibutuhkan dimana juga menghasilkan data dalam skala sangat besar yang dialirkan secara waktu nyata. Stabilitas pengiriman data ke dalam pengumpul menjadi sangat penting karena berpengaruh pada proses analitik dilokasi akhir. Untuk mengurangi kesalahan ini maka penelitian ini mempergunakan metode Concept Drift untuk melakukan identifikasi kapan ketidakteraturan terjadi dengan beberapa kriteria tertentu. Untuk melakukan klasifikasi dipergunakan SGD. Kami mempergunakan metode ADWIN dan Page-Hinkley untuk melakukan identifikasi pergeseran data. Hasil menunjukkan ADWIN lebih sensitive dibandingkan Page-Hinkley dengan mempergunakan dataset yang kami miliki.

Kata Kunci: *Internet of Things, Concept Drift, SGD Classifier, Adaptive Windowing (ADWIN), Page-Hinkley*

PENDAHULUAN

Perkembangan teknologi yang sangat pesat adalah sesuatu hal yang tidak dapat

dihindari didalam kehidupan manusia, terutama di dalam bidang teknologi informasi. Terlihat pada banyaknya

orang yang membuat aplikasi *Internet of Things* (IoT) yang diciptakan dengan tujuan untuk membantu atau memudahkan hidup manusia. *Internet of Things* itu sendiri adalah sebuah sistem dimana benda-benda di dunia fisik dapat dihubungkan ke internet oleh sensor [1].

Dengan banyaknya penggunaan data *real-time stream* pada perangkat IoT yang digunakan sebagai pengambilan keputusan dapat mengalami permasalahan ketika melakukan pengambilan dan pengiriman data sehingga diperlukannya teknik untuk melakukan identifikasi terjadinya permasalahan tersebut.

Concept drift didalam *machine learning* dan *data mining* merupakan perubahan hubungan antara data input dan output seiring dengan perubahan waktu. Hal ini terjadi akibat dari perubahan kondisi sumber data, yang kemungkinan terganggu, dan proses pengiriman data yang menyebabkan pola data berubah secara waktu nyata, dimana kejadian ini harus dimonitor. Hal ini menjadi penting karena perubahan pola dan relasi data secara waktu nyata akan mempengaruhi analisa yang dilakukan sehingga hasil yang diperoleh akan menyimpang secara berkala. Perubahan data yang mengalir ini bisa terjadi dengan perlahan, terkadang stabil dan tidak, hingga terjadi perubahan secara mendadak. Oleh karena itu *Concept Drift* menjadi penting untuk mengatasi kesalahan pembacaan dari *streaming* data yang secara *real-time*.

Identifikasi penyimpangan yang terjadi kemungkinan memiliki pola dan pergerakan data yang berbeda. Untuk itu teknik yang sama belum tentu sesuai dengan kasus berbeda, demikian juga sebaliknya. Semua itu bergantung pada hasil identifikasi yang dilakukan secara sistematis. Untuk dapat mempergunakan metode yang sesuai dengan pola dan

relasi data maka dapat dipergunakan asumsi mengenai pola data. Hal lain yang perlu dilakukan identifikasi adalah tipe perubahan yang terjadi, apakah lambat atau tiba-tiba.

Tulisan ini mempergunakan model atau algoritma seperti *Hoeffding Tree*, *Hoeffding Adaptive Tree*, *SGD Classifier*, *Adaptive Windowing* (ADWIN), dan *Page-Hinkley* memang diperuntukan untuk mengevaluasi data stream dan mengidentifikasi penyimpangan data.

METODE PENELITIAN

Perancangan penelitian ini terbagi dalam tiga tahapan yakni tahapan pengambilan data, tahapan evaluasi data menggunakan metode *Hoeffding Tree*, *Hoeffding Adaptive Tree & SGD Classifier*, tahapan identifikasi penyimpangan data menggunakan metode *Adaptive Windowing* (ADWIN) & *Page-Hinkley*

Pengambilan Data

Pada tahapan awal yang akan dilakukan adalah mengambil data *streaming* dari sensor pada IoT. *Dataset* yang digunakan pada penelitian ini merupakan data yang berkaitan keadaan atau kondisi pada suatu tempat, seperti suhu, tekanan udara, intensitas cahaya, dll. *Dataset* berjumlah 70 ribu *tuple streaming data* yang diambil dalam kurun waktu 2 bulan. Data-data ini diperoleh atau diambil dari suatu sensor yang bernama ClimaStick 2.0. Sensor ini akan menangkap data-data setiap waktu, dan data-data tersebut sudah otomatis tersimpan pada *dashboard* atau aplikasi Thinger.io. Data yang digunakan pada penelitian ini merupakan data yang bersifat *streaming*, maksudnya adalah data-data tersebut diperoleh secara terus menerus.

Evaluasi Data

Setelah mendapatkan data dari tahapan sebelumnya, maka tahapan selanjutnya

adalah mengevaluasi data tersebut. Tahapan ini bertujuan untuk mengevaluasi dan menganalisa apakah data *streaming* yang telah diperoleh memiliki *error rate* yang tinggi atau tidak. Pengujian data ini pun menggunakan dua metode yang dikhususkan untuk data *streaming* yaitu: *Hoeffding tree* yang merupakan suatu *decision tree* yang mampu menguji atau menganalisa data dalam skala yang cukup besar, biasanya pemodelan ini dikhususkan untuk mengolah data yang bersifat *streaming* [2] *Hoeffding tree* ini dikembangkan oleh Domingos dan Hulten dengan tujuan mendesain sebuah *decision tree* yang dapat mengolah dataset yang sangat besar bahkan tak terbatas, tanpa menyimpan dataset tersebut (*online*). *Hoeffding tree* memanfaatkan sampel kecil yang cukup untuk menentukan atribut pemisahan yang optimal, secara matematis *Hoeffding tree* menggunakan *Hoeffding bound* [2].

Hoeffding adaptive tree ini merupakan salah satu metode yang dikembangkan dari *Hoeffding tree*. Metode ini merupakan *decision tree* untuk data *streaming* dengan klasifikasi *adaptive naive bayes* pada cabangnya. Data *streaming* tersebut diuji tanpa mengetahui ukuran/parameter yang tetap dari *sliding window* [3]. Tujuan dari *Hoeffding adaptive tree* memonitor tingkat *error* mayoritas dan disetiap cabangnya memiliki *naive bayes decision*, yang dimana *naive bayes decision* akan digunakan jika cabang-cabangnya lebih akurat pada kasus sebelumnya [2].

Sementara SGD [4] merupakan metode klasifikasi secara linier yang akan memperkirakan *loss* dari setiap sampel disatu waktu dan dengan seiring berjalan model akan selalu diperbarui.

Accuracy [5] merupakan metrik yang menunjukkan pada ketepatan tingkat prediksi benar dengan keseluruhan data.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Dimana TP (*True Positive*) = prediksi yang benar dan kelas positif, TN (*True Negative*) = prediksi yang benar dan kelas negatif, FP (*False Positive*) = prediksi yang salah dan kelas positif, sedangkan FN (*False Negative*) = prediksi yang salah dan kelas negatif.

Precision [6] merupakan metrik yang menunjukkan rasio tingkat prediksi benar positif dibandingkan dengan hasil yang diprediksi positif.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Recall [6] merupakan metrik yang menunjukkan rasio tingkat prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

Kappa [5] merupakan metrik menunjukkan kecocokan (*degree of agreement*) untuk data kualitatif atau menunjukkan kelas-kelas (kelompok data) yang berbeda. Metrik *kappa* biasanya digunakan untuk mengatasi masalah *multi-class* dan *imbalanced class*.

$$Kappa = \frac{p_o - p_e}{1 - p_e} \quad (4)$$

Dimana p_o (*observed agreement*) merupakan *accuracy* prediksi dari suatu metode/model sedangkan p_e (*expected agreement*) merupakan probabilitas kesempatan “*agreement*” dari kedua pengklasifikasian.

Identifikasi *Concept Drift*

Pada tahapan ini akan dilakukan pengidentifikasian penyimpangan (*Concept drift*) pada *dataset* tersebut. Setelah pada tahapan sebelumnya *dataset* tersebut diuji menggunakan kedua metode sebelumnya, dan *dataset* tersebut harus memiliki *error rate* yang rendah tidak lebih dari 0.5. Selanjutnya, pada *dataset*

tersebut akan dilakukan pengidentifikasi-an penyimpangan data (*Concept drift*) dengan menggunakan dua metode yaitu:

Adaptive Windowing (ADWIN) [7] melakukan identifikasi penyimpangan data (*Concept Drift*) menggunakan *adaptive sliding window*. Page-Hinkley (PH) [8] mempergunakan teknik *sequential analysis* yang dapat digunakan sebagai pendeteksi penyimpangan data (*Concept Drift*). Metode ini menghitung nilai *observed* dan rata-ratanya sampai suatu waktu yang ditentukan

HASIL DAN PEMBAHASAN

Bagian ini berisi penjelasan terkait hasil dan analisa dari perancangan penelitian yang telah dibuat pada bagian sebelumnya. Semua hasil dari rancangan penelitian yang telah didapat akan dibahas pada bab ini. Tak hanya itu saja pada bagian ini juga akan memaparkan analisis berdasarkan hasil dari rancangan penelitian

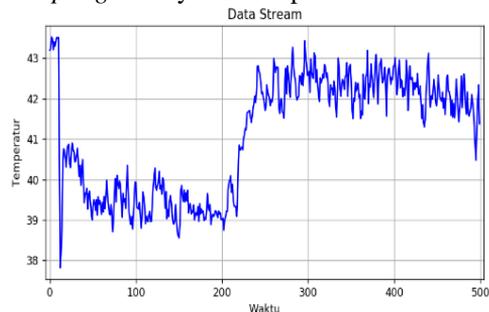
Hasil

Pada bagian ini akan dibagi menjadi beberapa bagian tahapan seperti pada perancangan penelitian pada bab sebelumnya. Mulai dari hasil pada tahapan pengambilan data stream pada perangkat IoT sampai tahapan identifikasi penyimpangan data (*concept drift*).

1). Pengambilan Data

Hasil pada bagian ini merupakan pengambilan data stream pada perangkat IoT sebanyak 70 ribu tuple yang diambil dalam kurun waktu kurang lebih 2 bulan. Data stream yang telah diambil merupakan data yang merekam perubahan suhu atau *temperature* pada suatu ruangan. Gambar 1. Memperlihatkan-

kan gambaran dari data *stream* dengan *sampling* sebanyak 500 tuple data.

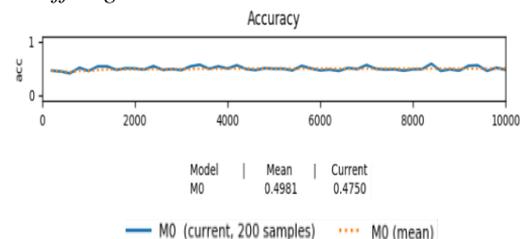


Gambar 1. Data *Stream* pada Perangkat IoT

2). Evaluasi Data

Hasil pada bagian ini memperlihatkan nilai *error rate* dari data *stream* yang telah diambil. Tak hanya itu hasil dari bagian ini memperlihatkan evaluasi performa kinerja dan keefetifkan kedua metode dalam hal *accuracy*, *precison*, *recall*, dan *kappa*. Lalu hasil dari perbandingan kedua metode tersebut akan ditunjukkan.

Ada beberapa *evaluator*/parameter yang digunakan untuk memperlihatkan hasil dari metode *hoeffding tree* seperti 10 ribu *sampling* data yang akan diuji, serta 100 sampel yang digunakan *train* model sebelum memulai mengevaluasi. Terlihat pada Gambar 2 hasil dari evaluasi data menggunakan metode *Hoeffding Tree*.

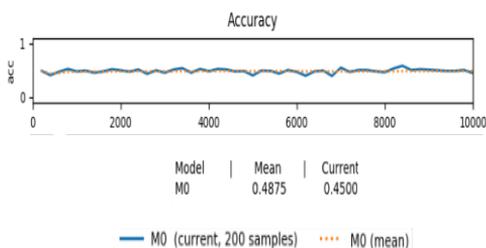


Gambar 2. Hasil *Accuracy* dari Evaluasi Data Metode *Hoeffding Tree*

Dapat terlihat bahwa pada Gambar 2 yang menunjukkan bahwa nilai rata-rata pada metrik *accuracy* dari 10 ribu sampel yang telah dievaluasi adalah sebesar 0,49 sedangkan nilai rata-rata dari 200 sampel terakhir (*current*) adalah sebesar 0.47.

b). *Hoeffding Adaptive Tree*

Bagian ini akan menunjukkan hasil evaluasi data menggunakan metode *hoeffding adaptive tree*. Beberapa *evaluator*/parameter yang digunakan untuk mendapatkan hasil dari metode *hoeffding adaptive tree*, seperti seperti 10 ribu *sampling* data yang akan dievaluasi, serta 100 sample yang digunakan train model sebelum memulai proses evaluasi. Ini menunjukkan hasil dari evaluasi data menggunakan metode *hoeffding adaptive tree*.



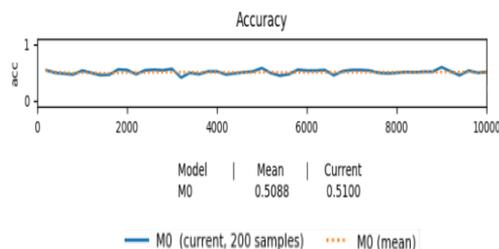
Gambar 3 *Accuracy* dari Evaluasi Data Metode *Hoeffding Adaptive Tree*

Pada Gambar 3, menunjukkan bahwa nilai rata-rata pada metrik *accuracy* dari 10 ribu sampel data *stream* yang telah dievaluasi adalah sebesar 0,48. Sedangkan nilai rata-rata dari 200 sampel terakhir (*current*) yang telah dievaluasi adalah sebesar 0,45.

c). *SGD Classifier*

Pada bagian ini akan menunjukkan hasil evaluasi data *stream* dengan menggunakan metode *sgd classifier*. Beberapa *evaluator*/parameter yang digunakan untuk mendapatkan hasil evaluasi dari metode *sgd classifier*, seperti maksimum sampel data *stream*

yang digunakan sebesar 10 ribu tuple data serta 100 sample yang digunakan untuk *train* model sebelum memulai proses evaluasi. Dibawah ini menunjukkan hasil evaluasi data dari metode *SGD classifier*.

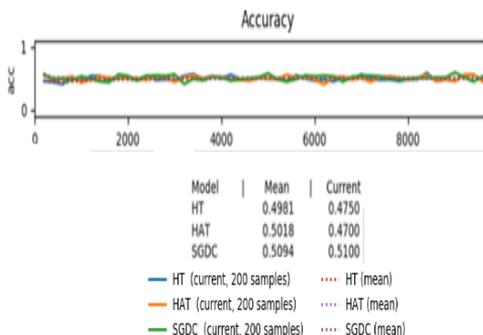


Gambar 4 *Accuracy* dari Evaluasi Data Metode *SGD Classifier*

Pada Gambar 4, menunjukkan bahwa nilai rata-rata pada metrik *accuracy* dari 10 ribu sampel data *stream* yang telah dievaluasi adalah sebesar 0,50. Sedangkan nilai rata-rata dari 200 sampel terakhir (*current*) yang telah dievaluasi adalah sebesar 0,51.

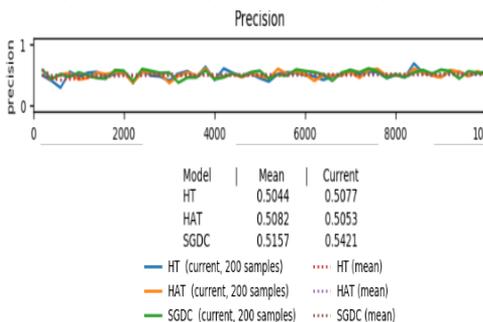
d). Perbandingan Ketiga Metode

Bagian ini akan memperlihatkan hasil evaluasi data *stream* perbandingan dari ketiga metode yaitu *hoeffding tree*, *hoeffding adaptive tree*, dan *SGD classifier*. Beberapa *evaluator*/parameter yang digunakan untuk mendapat hasil perbandingan evaluasi data dari ketiga metode, seperti maksimum sampel data yang digunakan sebesar 10 ribu tuple data serta 100 sample yang digunakan untuk *train* model sebelum memulai evaluasi.



Gambar 5 Hasil Accuracy dari Evaluasi Data Ketiga Metode

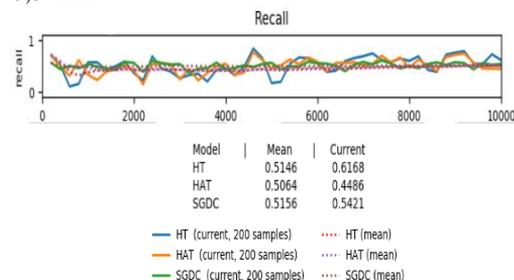
Pada Gambar 5, menunjukkan bahwa nilai rata-rata pada metrik *accuracy* dari ketiga metode dengan 10 ribu sampel data *stream* yang telah dievaluasi, yang masing-masing metode memiliki adalah sebesar HT = 0,4981, HAT = 0,5018, dan SGDC = 0,5094. Sedangkan, nilai rata-rata dari 200 sampel terakhir (*current*) yang telah dievaluasi dari ketiga metode masing masing adalah sebesar HT = 0,475, HAT = 0,47, dan SGDC = 0,51



Gambar 6 Hasil Precision dari Evaluasi Data Ketiga Metode

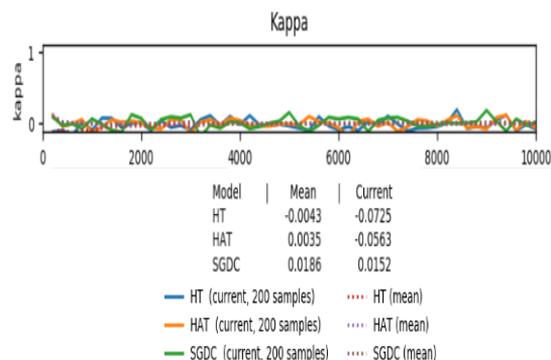
Pada Gambar 6, menunjukkan bahwa nilai rata-rata pada metrik *precision* dari ketiga metode dengan 10 ribu sampel data *stream* yang telah dievaluasi, yang masing-masing metode memiliki adalah sebesar HT = 0,5044, HAT = 0,5082, dan SGDC = 0,5157. Sedangkan, nilai rata-rata dari 200 sampel terakhir (*current*) yang telah dievaluasi dari ketiga metode masing-masing adalah sebesar HT =

0,5077, HAT = 0,5053, dan SGDC = 0,5421.



Gambar 7 Hasil Recall dari Evaluasi Data Ketiga Metode

Pada Gambar 7, menunjukkan bahwa nilai rata-rata pada metrik *recall* dari ketiga metode dengan 10 ribu sampel data *stream* yang telah dievaluasi, yang masing-masing metode memiliki adalah sebesar HT = 0,5146, HAT = 0,5064, dan SGDC = 0,5156. Sedangkan, nilai rata-rata dari 200 sampel terakhir (*current*) yang telah dievaluasi dari ketiga metode masing masing adalah sebesar HT = 0,6168, HAT = 0,4486, dan SGDC = 0,5421.



Gambar 8 Hasil Kappa dari Evaluasi Data Ketiga Metode

Pada Gambar 8, menunjukkan nilai rata-rata pada metrik *kappa* dari ketiga metode dengan 10 ribu sampel data *stream* yang telah dievaluasi, yang masing-masing metode memiliki adalah sebesar HT = -0,0043, HAT = 0,0035, dan SGDC = 0,186. Sedangkan, nilai

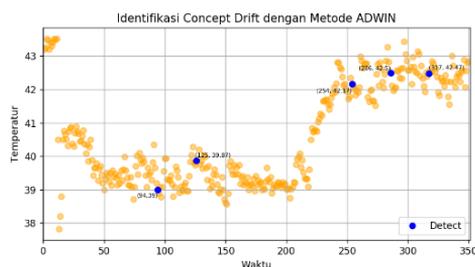
rata-rata dari 200 sampel terakhir (*current*) yang telah dievaluasi dari ketiga metode masing masing adalah sebesar $HT = -0,0725$, $HAT = -0,0563$, dan $SGDC = 0,0152$.

3). Identifikasi Data

Pada bagian ini akan menunjukkan hasil dari identifikasi penyimpangan data (*concept drift*) dari kedua metode yaitu *Adaptive Windowing* (ADWIN) dan *Page-Hinkley*.

a). ADWIN

Pada bagian ini akan menunjukkan hasil identifikasi penyimpangan data (*concept drift*) menggunakan metode (ADWIN). Sampel data yang digunakan sebanyak 350 tuple data, serta diasumsikan bahwa indeks data merupakan waktu (menit) ketika data *stream* tersebut terekam.



Gambar 9 Hasil Identifikasi *Concept Drift* dengan Metode ADWIN

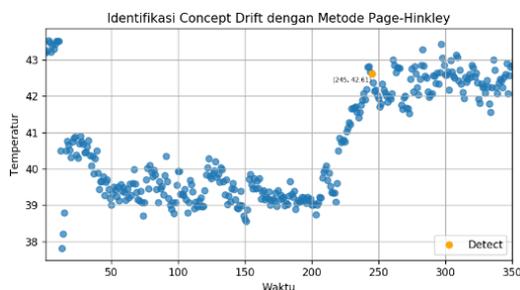
Mtd	Pre-train size	Accura	Precision	Recall
HT	100	0,498	0,504	0,514
	200	0,498	0,504	0,512
	300	0,499	0,505	0,512
HAT	100	0,496	0,502	0,518
	200	0,494	0,500	0,465
	300	0,496	0,503	0,531
SGD	100	0,509	0,515	0,515
	200	0,509	0,515	0,515
	300	0,509	0,516	0,516

Dapat terlihat pada Gambar 9, bahwa terjadi penyimpangan data (*concept drift*) yang terdeteksi pada indeks 94, 125, 254,

286, dan 317 ditandai dengan titik berwarna biru. Selain itu, sumbu x pada Gambar 9 tersebut merupakan *range* dari waktu, sedangkan sumbu y merupakan *range* dari *temperature*.

b). *Page-Hinkley*

Bagian ini akan menunjukkan hasil dari identifikasi penyimpangan data (*concept drift*) dengan metode *page-hinkley*. Sampel data yang digunakan sebanyak 350 tuple data, serta diasumsikan bahwa indeks data merupakan waktu (menit) ketika data *stream* tersebut terekam



Gambar 10 Hasil Identifikasi *Concept Drift* dengan metode *Page-Hinkley*

Dapat terlihat pada Gambar 10, bahwa penyimpangan data (*concept drift*) terdeteksi pada indeks 243 dan 325 ditandai dengan titik berwarna oranye. Selain itu, sumbu x pada Gambar 10 tersebut merupakan range dari waktu, sedangkan sumbu y merupakan *range* dari *temperature*.

Analisa

Tabel 1. Hasil Evaluasi HT, HAT dan SGD

Dapat terlihat pada Tabel 1, bahwa hasil evaluasi dari *hoeffding tree* dan *SGD Classifier* cenderung mengalami peningkatan seiring dengan penambahan parameter (*pre-train*) yang digunakan untuk melatih (*train*) model sebelum memulai evaluasi. Lalu total waktu (*train + test*) dan memori yang digunakan dari *hoeffding tree* dan *SGD Classifier*

cenderung stabil. Pada metode *hoeffding tree* hal tersebut terjadi karena

Indeks	Metode		Indeks	Metode	
	AW	PH		AW	PH
0 - 2000	95	246	68000 -70000	68511	68356
	127				
	255				
	287				
	318				
	575				
	607				
	639				
	735	775		68607	
	799			68639	
	831			68991	
	991			69023	
	1119			69087	
	1151			68680	
	1215				
	1279				
	1311				
	1343				
	1407	957		69599	
	1439			69887	
	1663			69785	
	1695				
	1855				
	1887				
	1919				
	1951				

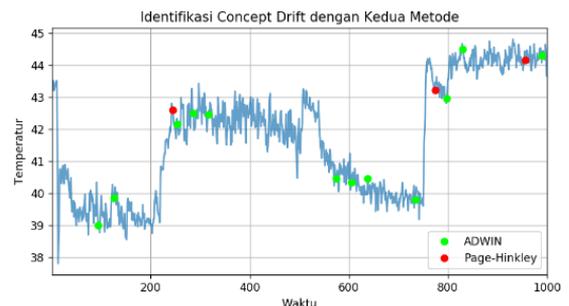
disebabkan *hoeffding tree* merupakan metode *incremental* yang dimana data atau sampel (*pre-train*) digunakan untuk memperluas pengetahuan model, oleh karena itu hasil metrik-metrik dari metode *hoeffding tree* juga cenderung meningkat. Begitu juga dengan *SGD Classifier* yang pada penelitian ini digunakan untuk mendemonstrasikan *compability* dengan metode *incremental*, oleh karena itu hasil metrik-metrik dari metode *SGD Classifier* juga cenderung meningkat. Dari hasil evaluasi dari metode *SGD Classifier* menunjukkan hasil yang paling tinggi disetiap metrik dibandingkan dengan kedua metode yang lain. Sedangkan pada metode *hoeffding*

adaptive tree, hasil evaluasi disetiap metrik menunjukkan ketidakstabilan seiring dengan peningkatan pada jumlah sampel (*pre-train*), serta total waktu (*train + test*) dan memori digunakan cenderung meningkat. Hal ini disebabkan karena metode *hoeffding adaptive tree* tidak memiliki ukuran yang tepat untuk menentukan *sliding window* dan ukuran tersebut sulit ditebak atau tergantung dari tingkat perubahan distribusi dari dataset tersebut.

Tabel 2, menunjukkan hasil identifikasi *concept drift* dari kedua metode *ADWIN* dan *page-hinkley* terhadap data *stream* sebanyak 70 ribu tuple data. Selain itu dapat terlihat bahwa metode *ADWIN* lebih banyakk mengidentifikasi *concept drift* dibandingkan dengan metode *Page-Hinkley*.

Tabel 2 Hasil Identifikasi *Concept Drift* dari Kedua Metode

Sementara pada gambar 11 menunjukkan hasil identifikasi *concept drift* dari kedua metode *ADWIN* dan *Page-Hinkley* pada 1000 tuple data *stream*.



Gambar 11 Identifikasi *Concept Drift* dari Kedua Metode

Terlihat pada Gambar 11, bahwa kedua metode memiliki perbedaan hasil dalam mengidentifikasi *concept drift*. Metode *ADWIN* dapat mengidentifikasi 12 penyimpangan data sedangkan metode *Page-Hinkley* dapat mengidentifikasi 3

penyimpangan data. Selain itu metode ADWIN cenderung mengidentifikasi penyimpangan data ketika terjadi kenaikan temperatur. Berbeda dengan metode ADWIN, metode *Page-Hinkley* cenderung mengidentifikasi penyimpangan data ketika terjadi penurunan temperatur.

KESIMPULAN

Penelitian ini dapat disimpulkan bahwa pada penelitian ini data *stream* yang digunakan mengalami perubahan data berdasarkan waktu secara bertahap sehingga membentuk pola *gradual*. Selain itu, berdasarkan hasil evaluasi data pada penelitian ini yang menggunakan ketiga metode yakni *Hoeffding Tree*, *Hoeffding Adaptive Tree*, dan *SGD Classifier* dan ketiga metode tersebut memiliki perbandingan antara nilai prediksi yang benar dengan nilai prediksi yang salah hampir sama. Metode *SGD Classifier* memiliki hasil nilai rata-rata yang paling tinggi pada setiap metrik yakni *accuracy*, *precision*, *recall*, dan *kappa*. Selain itu metode *SGD Classifier* paling sedikit dalam menggunakan memori. Pada hasil identifikasi penyimpangan data/*concept drift* dari kedua metode yakni ADWIN dan *Page-Hinkley*, metode ADWIN yang mengidentifikasi penyimpangan data/*concept drift* yang paling banyak dibandingkan dengan metode *Page-Hinkley*, tetapi metode ADWIN cenderung mengidentifikasi penyimpangan data/*concept drift* ketika terjadi peningkatan temperatur. Sedangkan, metode *Page-Hinkley* mengidentifikasi penyimpangan data ketika terjadi penurunan temperatur. Berdasarkan penelitian yang telah dilakukan masih terdapat kekurangan yang dapat dikembangkan kemudian hari seperti pada tahapan evaluasi data dapat menggunakan metode-metode terbaru yang dikhususkan untuk data stream sehingga hasil yang didapatkan lebih

baik. Selain itu pada tahapan pengidentifikasian penyimpangan data dapat menggunakan metode-metode terbaru atau metode yang lebih akurat dalam mengidentifikasi penyimpangan data.

DAFTAR PUSTAKA

- [1] K. Asthon, "That ' Internet of Things ' Thing," *RFID J.*, p. 4986, 2010.
- [2] P. K. Srimani and M. M. Patil, "Performance analysis of Hoeffding trees in data streams by using massive online analysis framework," *Int. J. Data Mining, Model. Manag.*, vol. 7, no. 4, pp. 293–313, 2015.
- [3] A. Bifet and R. Gavaldà, "Adaptive Parameter-free Learning from Evolving Data Streams," *Adv. Intell. Data Anal. VIII*, no. September, pp. 249–260, 2017.
- [4] H. Robbins and S. Monro, "A Stochastic Approximation Method," *Ann. Math. Stat.*, vol. 22, no. 3, pp. 400–407, 1951.
- [5] Y. Liu, Y. Zhou, S. Wen, and C. Tang, "A Strategy on Selecting Performance Metrics for Classifier Evaluation," *Int. J. Mob. Comput. Multimed. Commun.*, vol. 6, no. 4, pp. 20–35, 2014.
- [6] D. M. W. Powers, "Evaluation : From Precision , Recall and F-Factor to ROC , Informedness , Markedness & Correlation," no. December, 2007.
- [7] A. Bifet and R. Gavaldà, "Learning from time-changing data with adaptive windowing," *Proc. 7th SIAM Int. Conf. Data Min.*, no. April, pp. 443–448, 2007.
- [8] E. S. Page, "Continuous Inspection Schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, Oct. 1954.

RANCANGAN CNC PLOTTER UNTUK MENULIS DAN MENGGAMBAR

Christhoper Tanato¹⁾, Alfa Satya Putra^{2*)}

¹Sistem Komputer, Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: christopher_tanato@yahoo.com¹⁾

²Sistem Komputer, Fakultas Ilmu Komputer, Universitas Pelita Harapan
E-mail: alfa.putra@uph.edu²⁾

**Penulis Korespondensi*

Abstract – The growth of technology in writing and drawing using machineries has experienced a significant growth, one such technology is a CNC Plotter machine. This research focuses on designing a CNC Plotter machine that can be programmed to write and draw with movement similar to human hand movement. CNC Plotter is designed using Arduino as microcontroller, stepper motors, and A4988 module as motor driver. G-code is used to control the movement of CNC Plotter, and can be created from importing an image that has been threshold filtered and using trace bitmap function to obtain the outer line of the image. G-code is created using InkScape and is processed using Processing. From experiment results, it can be concluded that the designed CNC Plotter machine works with very good accuracy. Stepper motor works very well. Images and texts created using CNC Plotter has accuracy rate between 85% to 90%.

Keywords: CNC Plotter, Arduino, Stepper motor, G-code

Abstrak – Perkembangan teknologi untuk menulis dan menggambar menggunakan mesin terus mengalami kemajuan yang signifikan, salah satunya adalah teknologi mesin CNC Plotter. Penelitian ini bertujuan untuk merancang sebuah mesin CNC Plotter yang dapat diprogram untuk menulis dan menggambar dengan gerakan yang menyerupai pergerakan tangan manusia. CNC Plotter dirancang menggunakan Arduino sebagai mikrokontroler, stepper motor, dan modul A4988 sebagai motor driver. G-code digunakan untuk mengatur pergerakan dari CNC Plotter, dan dibuat dari import gambar yang telah difilter threshold dan menggunakan fungsi trace bitmap untuk mendapatkan outer-line dari gambar. G-code dibuat menggunakan aplikasi InkScape dan dibaca menggunakan program Processing. Dari hasil pengujian yang dilakukan, dapat disimpulkan bahwa rancangan dari CNC Plotter yang dibuat berjalan dengan akurasi cukup baik. Stepper motor dapat bekerja dengan baik. Gambar dan tulisan yang dibentuk oleh mesin CNC Plotter memiliki tingkat akurasi antara 85% hingga 90%.

Kata Kunci: CNC Plotter, Arduino, Stepper motor, G-code

PENDAHULUAN

Tradisi menulis sudah dimulai sejak 35.000 tahun lalu sebelum masehi,

Diterima <10062021>, Revisi <12072021>, Diterima untuk publikasi <26072021>.
Copyright © 2021 Published by PSDKU Sistem Informasi UPH Kampus Medan,
ISSN : 2528-5114

sedangkan tradisi menggambar dimulai sejak 30.000 tahun lalu sebelum masehi[1][2]. Tulisan pertama kali ditemukan berupa penjelasan kejadian-kejadian yang terjadi pada zaman dahulu, sedangkan menggambar ditemukan dengan gambar binatang yang ditemui manusia pada zaman dahulu[3]. Salah satu perkembangan ilmu menulis adalah menulis indah, yaitu tulisan yang memiliki makna yang sama namun dengan bentuk tulisan yang dibuat sedikit berbeda dengan tulisan biasa.

Perkembangan dalam teknologi alat untuk menulis dan menggambar juga mengalami kemajuan yang cepat. Saat ini terdapat berbagai mesin yang dapat menghasilkan tulisan dan gambar, salah satunya adalah mesin *printer*. Namun mesin *printer* tidak dapat melakukan gerakan tangan untuk menulis dan menggambar. Makalah ini akan membahas penelitian menggunakan teknologi alat *CNC Plotter* untuk membuat tulisan dan gambar. Penelitian ini bertujuan untuk membuat prototipe mesin *CNC Plotter* yang dapat menulis dan menggambar dengan gerakan yang menyerupai gerakan manusia. Batasan-batasan yang diterapkan pada penelitian ini antara lain adalah mikrokontroler yang digunakan untuk menjalankan mesin CNC adalah Arduino Uno, mesin CNC hanya digunakan pada bidang datar 2 dimensi, aktuator yang digunakan adalah *Stepper Motor* dan servo, mesin CNC hanya menggunakan 1 pulpen untuk menggambar dan menulis, mesin CNC hanya dapat menulis sesuai font yang terdapat pada *software* untuk membuat tulisan, dan mesin CNC hanya dapat membaca file *g-code*.

METODE PENELITIAN

Metode yang dilakukan dalam penelitian ini adalah studi literatur untuk pembelajaran bahasa pemrograman C, perhitungan matematika dan logika, proses kendali dengan sistem control, dan teori struktur data. Lalu dilanjutkan

dengan studi eksperimen untuk kinerja *Stepper Motor* dan mikrokontroler Arduino dan pemahaman komunikasi Arduino dengan *Stepper Motor*.

LANDASAN TEORI

Pada bagian ini akan dijelaskan beberapa dasar teori yang digunakan pada penelitian ini, antara lain teori tentang *Computer Numerical Control (CNC)*, *CNC Plotter*, Arduino UNO, InkScape, G-Code, Stepper dan Servo Motor.

Computer Numerical Control (CNC)

Computer Numerical Control (CNC) adalah mesin untuk mengelolah bahan mentah berdasarkan desain yang telah dibuat pada *software* dan dikirimkan ke *microcontroller* yang terdapat pada mesin CNC. Konsep *Computer Numerical Control* merupakan pengembangan dari konsep *Numerical Control (NC)* yang bekerja berdasarkan kumpulan instruksi kode berupa angka, huruf, dan simbol yang dapat dimengerti oleh *Machine Control Unit (MCU)*[4][5].

CNC Plotter

CNC Plotter merupakan salah satu jenis *printing* yang menggunakan pulpen untuk menggambar pada sebuah permukaan rata. *CNC Plotter* umum digunakan untuk melakukan *printing* baik dalam dua maupun tiga dimensi, dan memiliki kelebihan dimana komponen yang digunakan relatif murah dan mudah untuk didapatkan[4].

Arduino UNO

Arduino UNO adalah papan *microcontroller* yang menggunakan chip ATmega328P, memiliki 14 pin digital input/output dimana 6 pin dapat digunakan sebagai PWM (*Pulse Width Modulation*) output, 6 pin analog input, 16 MHz *crystal oscillator*, konektor USB tipe B *female*, power jack DC, ICSP (*In Circuit Serial Programming*) header, dan tombol *reset*. Arduino UNO dapat

deprogram menggunakan kabel USB dan software Arduino IDE[6].

InkScape

InkScape adalah perangkat lunak *editor* grafis vektor yang bersifat *open-source*. InkScape menggunakan *Scalable Vector Graphics* (SVG) sebagai format standar penyimpanan file. InkScape juga dapat menggunakan ekstensi untuk format *file g-code* yang digunakan pada mesin CNC. Pada penelitian ini, versi InkScape yang digunakan adalah versi 0.48 dengan ekstensi untuk mengolah *file g-code*.

G-Code

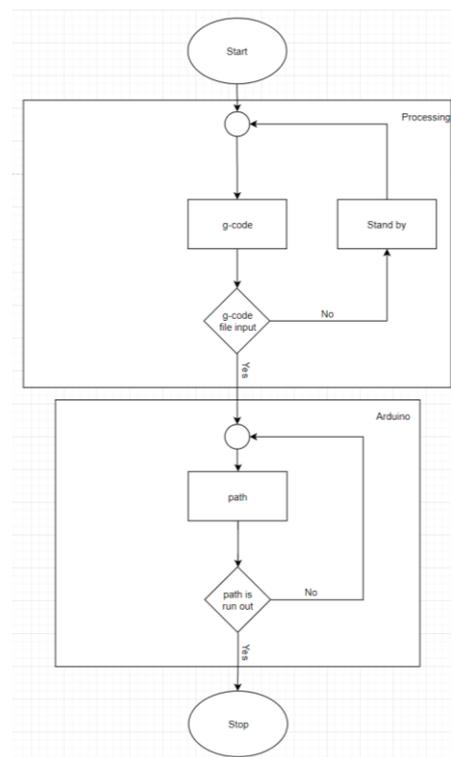
G-code adalah fungsi yang digunakan dalam bahasa pemrograman *Numerical Control* (NC) yang mengandung informasi posisi alat penggambar pada mesin. Perintah yang dilakukan pada G-code meliputi mengubah *pallette*, mengatur pergerakan alat, dan mengatur informasi alat. *Software Processing* digunakan untuk memproses file G-code, dan software CAMotic digunakan untuk melakukan simulasi file G-code pada penelitian ini[7].

Motor

Pada penelitian ini jenis motor yang digunakan adalah *Stepper Motor* dan *Servo Motor*. *Stepper Motor* adalah motor DC yang tidak mempunyai komutator, umumnya hanya mempunyai kumparan pada bagian *stator* sedangkan bagian *rotor* merupakan magnet permanen. *Stepper motor* dapat diperintah untuk bergerak ke posisi dan berputar ke arah yang diinginkan[8]. *Servo Motor* adalah perangkat atau aktuator putar yang dirancang dengan sistem kontrol umpan balik *loop* tertutup, dapat diatur untuk menentukan dan memastikan posisi sudut dari poros *output* motor[9].

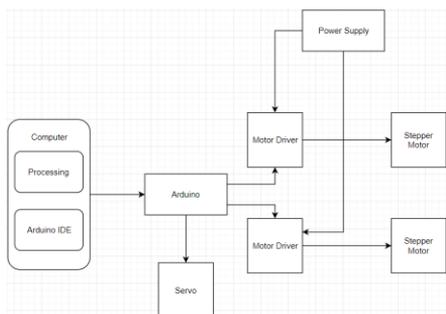
PERANCANGAN SISTEM

Sistem CNC *Plotter* diharapkan dapat bekerja seperti diagram *flow chart* pada Gambar 1. *Flow chart* dibagi menjadi dua bagian, yaitu bagian *Processing* dan bagian Arduino. Sistem dimulai dari bagian *Processing* dengan membaca file *g-code* yang telah dimasukkan oleh pengguna. Isi *g-code* diubah menjadi *string* yang dapat dibaca oleh Arduino. *String* kemudian dikirimkan ke Arduino untuk diproses hingga *string* selesai dibaca.



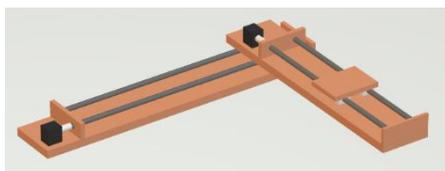
Gambar 1. Flow Chart Sistem

Sistem CNC *Plotter* memiliki diagram blok seperti pada Gambar 2. Komputer memberikan instruksi kepada Arduino untuk menjalankan *Servo* dan *Stepper Motor*. *Stepper Motor* memerlukan daya 12-36V, sehingga diperlukan sebuah *power supply*.



Gambar 2. Diagram Blok Sistem

Pada perancangan sistem CNC *Plotter* ini akan dibuat 3 model mesin yang memiliki penggerak yang berbeda, yaitu *threadroad*, *timing belt*, dan *gear&rack gear*. Gambar 3 adalah desain 3D dari CNC *Plotter* yang akan dibuat.



Gambar 3. Desain 3D CNC *Plotter*

HASIL DAN PEMBAHASAN

Pengujian dibagi menjadi dua bagian, yaitu pengujian pergerakan stepper motor dan pengujian menulis dan menggambar.

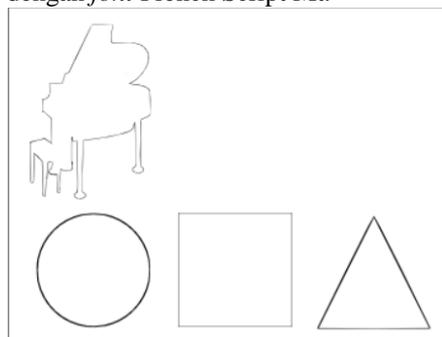
Pada pengujian *stepper motor* dilakukan uji putaran step *Stepper Motor* dengan menggunakan *motor driver A4988* dan kecepatan putaran motor. Dari pengujian yang dilakukan, didapatkan hasil rumus untuk menghitung skala 1 koordinat untuk menjadi 1 cm, yaitu rumus (1):

$$Step' = \frac{Panjang'}{\frac{Jarak}{Step^\circ} \times Panjang^\circ} \quad (1)$$

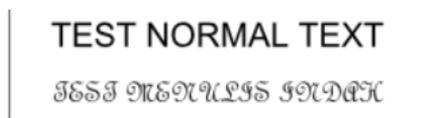
Dimana *Step'* adalah skala perubahan 1 satuan koordinat untuk step dalam perbandingan ukuran *real* dan *g-code*, *Panjang'* adalah panjang ukuran kertas, *Jarak* adalah jarak yang ditempuh dalam *Step°*, *Step°* adalah jumlah *step* yang digunakan sebagai standar, dan *Panjang°*

adalah ukuran kertas dalam skala XY pada *g-code*.

Pengujian menulis dan menggambar dilakukan dengan membuat *g-code* untuk tulisan dan gambar menggunakan *InkScape*. Gambar 4 merupakan gambar yang akan dibentuk berupa piano, lingkaran, segiempat dan segitiga. Gambar 5 adalah tulisan yang akan dibentuk berupa tulisan “TEST NORMAL TEXT” dengan font *Calibri*, dan tulisan “TEST MENULIS INDAH” dengan *font French Script Mt*.

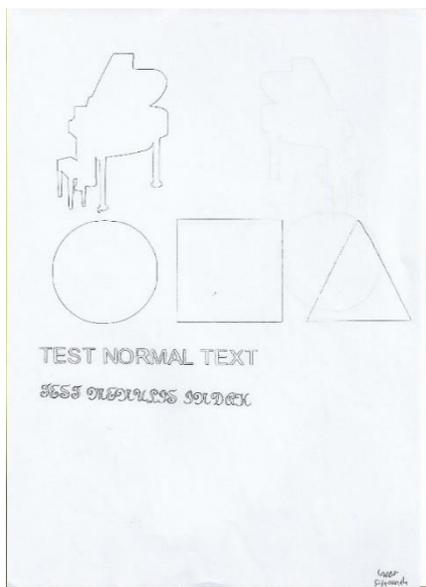


Gambar 4. Pengujian Gambar



Gambar 5. Pengujian Tulisan

Setelah *g-code* untuk gambar dan tulisan dibuat, dilakukan pengujian pada CNC *Plotter* untuk membuat gambar dan tulisan yang diberikan. Hasil dari mesin CNC *Plotter* untuk menulis dan menggambar dapat dilihat pada gambar 6, dan analisis dari hasil menulis dan menggambar dapat dilihat pada tabel 1. Berdasarkan hasil yang diperoleh, dapat disimpulkan bahwa tingkat akurasi dari mesin CNC *Plotter* untuk menulis dan menggambar berkisar antara 85% hingga 90%.



Gambar 6. Hasil CNC Plotter

Tabel 1. Perbandingan Desain dan Hasil

Desain	Hasil
Dapat bergerak secara vertikal dan horizontal seperti tangan	Dapat bergerak secara vertikal dan horizontal seperti tangan
Dapat bergerak membentuk lengkungan	Gergerak membentuk lengkungan tidak sempurna
Dapat menghasilkan tulisan dengan gerakan seperti tulisan tangan	Dapat menghasilkan tulisan dengan pergerakan yang sedikit berbeda dengan tangan
Dapat menghasilkan gambar dengan gerakan seperti menggambar tangan	Dapat menghasilkan gambar dengan pergerakan yang sedikit berbeda dengan menggambar tangan

KESIMPULAN DAN SARAN

Dari hasil perancangan dan pengujian yang telah dilakukan maka dapat disimpulkan bahwa rancangan mesin CNC Plotter dapat melakukan pergerakan yang menyerupai gerakan tangan manusia untuk menulis dan menggambar dengan tingkat akurasi antara 85% hingga 90%. Rancangan mesin CNC Plotter mengalami sedikit kesulitan dalam menulis dan

menggambar lengkungan dengan baik, dikarenakan perhitungan untuk menggerakkan sumbu X dan Y dengan stepper motor masih belum sempurna. Untuk pengembangan kedepan, beberapa saran yang dapat diterapkan untuk meningkatkan akurasi dari mesin CNC Plotter antara lain adalah penggunaan stepper motor yang memiliki jarak tempuh 1 step lebih kecil, menggunakan dua Stepper Motor pada sumbu Y, dan menggunakan bahan dasar dan kerangka yang lebih kokoh dalam pembuatan mesin CNC plotter.

DAFTAR PUSTAKA

[1] Schmandt-Besserat, D. (1992). Before Writing. (2 vols). Austin, Texas: University of Texas Press.

[2] Bruce Cole, Adelheid M. Gealt (1991). Art of the Western World: From Ancient Greece to Post Modernism. Simon and Schuster.

[3] Baines, J. (2007). Visual and Written culture in Ancient Egypt. Oxford: Oxford University Press, Cambridge: Cambridge University Press.

[4] Prof. Neha Chourasia. 2018. Implementation of Low Cost CNC Plotter Using Arduino. International Journal for Research in Applied Science & Engineering Technology Volume 6 Issue IV, Halaman 20-28. <https://www.ijraset.com/files/serve.php?FID=15734>.

[5] Sutarman. 2018. Computer Numerical Control (CNC) Milling and Turning for Machining Process in Xintai Indonesia. Quest Journals Journal of Research in Mechanical Engineering Volume 3 ~ Issue 5. Halaman 1-7. <http://www.questjournals.org/jrme/papers/vol3-issue5/A350107.pdf>

- [6] Arduino. 2021. What is Arduino?
<https://www.arduino.cc/en/Guide/Introduction>
- [7] Stp Administrator.
<http://technopark.surakarta.go.id/id/media-publik/sains-dan-edukasi/543-mengenal-program-g-code-pada-mesin-cnc>.
- [8] Syahrul. 2011. Stepper Motor: Teknologi, Metoda dan Rangkaian Kontrol. Jurnal Majalah Ilmiah Unikom Vol.16, No.2. Halaman 187-202.
https://jurnal.unikom.ac.id/_s/data/jurnal/v06-n02/vol-6-artikel-7.pdf/pdf/vol-6-artikel-7.pdf.
- [9] Trikueni Dermanto. 2014. Pengertian dan Prinsip Kerja Motor Servo. <http://trikueni-desain-sistem.blogspot.com/2014/03/Pengertian-Motor-Servo.html>

IMPLEMENTASI METODE ANALYTIC HIERARCHY PROCESS UNTUK PEMILIHAN SAHAM TERBAIK BERBASIS WEBSITE

Rudolfo Rizki Damanik^{1*}, Leonardy Khanady²⁾

¹Fakultas Ilmu Komputer, Universitas Pelita Harapan Kampus Medan
E-mail: rudolfo.damanik@uph.edu¹⁾

²Fakultas Ilmu Komputer, Universitas Pelita Harapan Kampus Medan
E-mail: lk3091@student.uph.edu²⁾

**Penulis Korespondensi*

Abstract – The number of stock investors is increasing from the past until now. Likewise with the number of companies listed on the Indonesia Stock Exchange (IDX). Which makes investors confused about choosing the best stock from hundreds of stocks traded on the IDX. The method used for selecting the best stock is using the Analytic Hierarchy Process (AHP) method. Investors will enter an assessment of the comparison between criteria and between alternatives for each criterion. The criteria used for stock selection are Price to Earnings Ratio (PER), Price to Book Value (PBV), Debt to Equity Ratio (DER), Return on Equity (ROE), Return on Assets (ROA). The researcher designed a web-based decision support system which helps investors in determining the best stocks according to their own assessment with the AHP method approach. From the results obtained that the AHP method approach gives the best results in the form of stocks.

Keywords: Decision Support System, Analytical Hierarchy Process, Stock, Website

Abstrak – Jumlah investor saham semakin meningkat dari dulu hingga saat ini. Begitu juga dengan jumlah perusahaan yang tercatat di Bursa Efek Indonesia (BEI). Yang dimana membuat investor bingung untuk memilih saham yang terbaik dari ratusan saham yang diperdagangkan di BEI. Metode yang digunakan untuk pemilihan saham terbaik yaitu menggunakan metode Analytic Hierarchy Process (AHP). Investor akan memasukkan penilaian terhadap perbandingan antar kriteria dan antar alternatif tiap kriteria. Kriteria-kriteria yang digunakan untuk pemilihan saham adalah Price to Earnings Ratio (PER), Price to Book Value (PBV), Debt to Equity Ratio (DER), Return on Equity (ROE), Return on Assets (ROA). Pada penelitian ini, peneliti merancang suatu sistem pendukung keputusan berbasis web yang dimana membantu investor dalam menentukan saham terbaik yang sesuai dengan penilaiannya sendiri dengan pendekatan metode AHP. Dari hasil yang didapatkan bahwa pendekatan metode AHP memberikan hasil berupa saham terbaik.

Kata Kunci: Sistem Pendukung Keputusan, Analytical Hierarchy Process, Saham, Situs Web

PENDAHULUAN

Menabung saham mulai digemari pada zaman digital ini khususnya untuk investor. Selain kemudahan dan likuiditas yang relatif tinggi, investasi pada saham biasanya memberikan keuntungan yang lebih tinggi dari tingkat inflasi. Saham itu sendiri merupakan surat – surat berharga yang menunjukkan bagian kepemilikan atas suatu perusahaan. Di Indonesia penjualan dan pembelian saham dilakukan di Bursa Efek Indonesia (BEI) melalui pialang (*broker*) yang dipilih sendiri dan diawasi oleh Otoritas Jasa Keuangan (OJK).

Memilih saham terbaik merupakan hal yang paling sulit dikarenakan perlunya pengalaman dan analisis yang mendalam. Saham merupakan investasi yang beresiko tinggi karena fluktuasi harga dan likuidasi dari saham itu sendiri. Sehingga hal ini perlu dicermati oleh investor dalam pemilihan saham terbaik. Pergerakan harga saham biasanya dipengaruhi oleh beberapa faktor seperti laporan keuangan perusahaan, kinerja perusahaan, isu / berita dsb. Maksud dari saham terbaik itu sendiri adalah saham yang memiliki kriteria yang paling sesuai dengan harapan investor.

Dengan bantuan Sistem Pendukung Keputusan (SPK) ini dapat memudahkan *investor* dalam memilih saham yang terbaik. SPK adalah sistem informasi yang membantu suatu bisnis ataupun organisasi dalam pengambilan suatu keputusan. Dengan menggunakan sistem ini, Investor dapat mempercepat dan memudahkan analisis untuk menentukan alternatif terbaik atau saham terbaik.

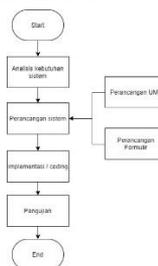
Penelitian terhadap sistem pendukung keputusan sudah dilakukan oleh beberapa peneliti yaitu yang menggunakan metode AHP untuk memprioritaskan perbaikan jalan dengan menggunakan multi-kriteria dalam perumusan alternatif. Pada penelitian tersebut menghasilkan peringkat dalam skala prioritas perbaikan jalan berdasarkan persepsi dari

responden[1]. Paksi menerapkan metode *Fuzzy Technique for Order Preference by Similarity to Ideal Solution* (FTOPSIS) pada pemilihan saham terbaik berdasarkan analisis rasio keuangan. Metode menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan terjauh dari solusi ideal negatif untuk menentukan kedekatan relatif dari suatu alternatif dengan solusi optimal[2].

Analytical Hierarchy Process (AHP) merupakan metode untuk menyelesaikan suatu permasalahan yang memiliki struktur hierarki yang rumit dan membantu pengambilan suatu keputusan [3]. Metode ini digunakan untuk menentukan saham terbaik karena metode AHP dapat menganalisis permasalahan yang kompleks dengan kriteria seperti *Price to Earnings Ratio* (PER), *Price to Book Value* (PBV), *Debt to Equity Ratio* (DER), *Return on Equity* (ROE), *Return on Assets* (ROA).

METODE PENELITIAN

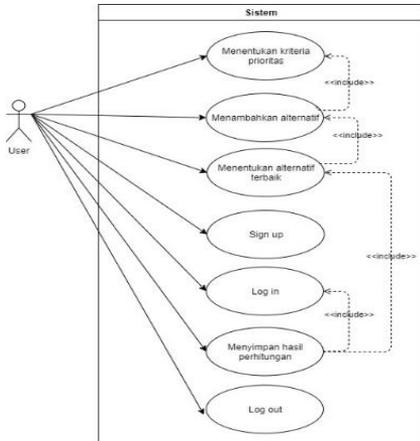
Pada penelitian ini perancangan Sistem Pendukung Keputusan menggunakan metode *waterfall*, Metode *Waterfall* menjadi pilihan dalam pengembangan aplikasi ini karena dengan metode ini, pengerjaan aplikasi akan lebih terstruktur dan berurut sehingga aplikasi yang dihasilkan dapat optimal dan meminimalisir kekurangan[4].



Gambar 1. Pengembangan Aplikasi Metode *Waterfall*

Perancangan Sistem *Use Case Diagram*

Use case diagram biasanya digunakan dalam perancangan sistem untuk memperoleh kebutuhan fungsional dari suatu sistem dimana terdiri dari interaksi antara beberapa pengguna (*actor*) dengan sistem secara keseluruhan[5].

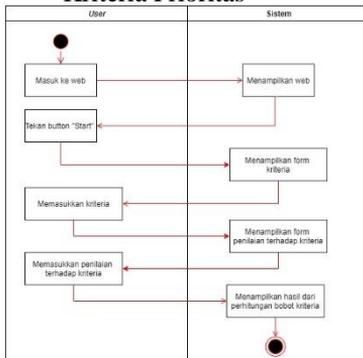


Gambar 2. Use Case Diagram Sistem Pendukung Keputusan Menggunakan Metode AHP.

Activity Diagram

Activity Diagram merupakan diagram yang menggambarkan tahapan – tahapan aktifitas dalam setiap use case terdiri dari

a. Activity Diagram Menentukan Kriteria Prioritas

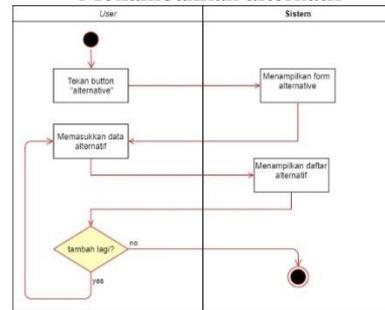


Gambar 3. Activity Diagram Menentukan Kriteria Prioritas

Pada diagram ini menggambarkan cara untuk menentukan nilai prioritas pada kriteria yang dimulai dengan menekan

button start pada home page. Maka sistem akan menampilkan form criteria. Setelah itu user memasukkan kriteria dan melanjutkan ke form penilaian perbandingan berpasangan kriteria. Setelah memasukkan nilai tersebut maka sistem akan menghitung nilai prioritas dan rasio konsisten. Jika rasio konsisten tidak dicapai maka user harus mengulangi perhitungan dengan memasukkan kembali nilai pada form. Setelah melewati rasio konsisten maka sistem akan menampilkan hasil perhitungan pada halaman hasil.

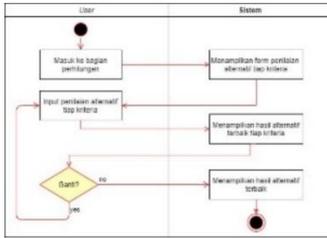
b. Activity Diagram Menambahkan alternatif



Gambar 4. Activity Diagram Menambahkan Alternatif

Pada diagram ini user dapat melakukan penambahan, mengganti, ataupun menghapus alternatif pada sistem. Dengan cara menekan button alternatif pada menu navigasi. Setelah itu sistem akan menampilkan form alternatif dan user tinggal memasukkan alternatif.

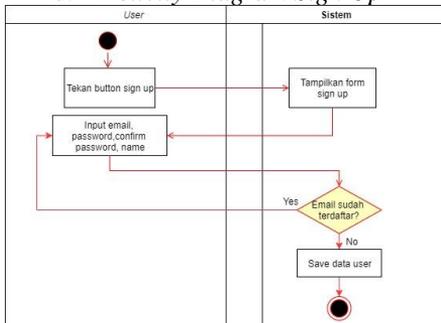
c. *Activity Diagram* menentukan alternatif terbaik



Gambar 5. Activity Diagram menentukan alternatif terbaik

Setelah menambahkan kriteria dan alternatif maka *user* akan menghitung nilai prioritas pada perbandingan berpasangan alternatif pada setiap kriteria. Dengan cara memilih tombol *calculate*, setelah itu pilih kriteria dan masukkan nilai perbandingan alternatifnya. Jika hasil perhitungan melebihi rasio konsistensi maka perhitungan diulangi lagi. Setelah semua perhitungan antara alternatif tiap kriteria selesai maka sistem akan menuju menampilkan hasil akhir yaitu alternatif terbaik.

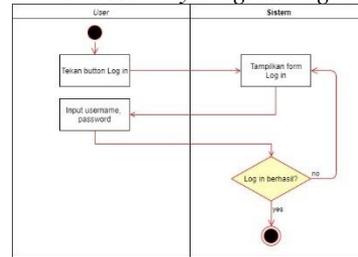
d. *Activity Diagram Sign Up*



Gambar 6. Activity Diagram Sign Up

Pada diagram ini menggambarkan bagaimana *user* untuk mendaftar ke dalam sistem dalam *database* sebagai user yang dikenal. *User* memasukkan nama, *email*, *password*, ke dalam *form Sign up*. jika email sudah terdaftar dalam sistem maka *user* menggunakan email yang lain untuk registrasi.

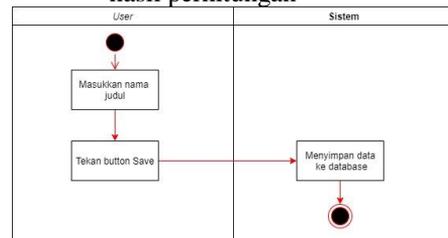
e. *Activity diagram log in*



Gambar 7. Activity Diagram Log In

Pada diagram ini menjelaskan bagaimana user bisa *login* ke dalam sistem dengan cara menekan tombol *log in*. Setelah itu masukkan *email* dan *password* yang sama saat registrasi. Jika data sudah diverifikasi oleh sistem maka sistem akan mengarahkan halaman sebelumnya saat user menekan *button log in*.

f. *Activity diagram* menyimpan hasil perhitungan

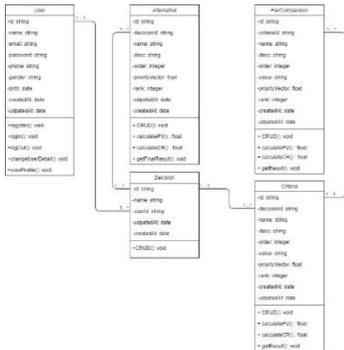


Gambar 8. Activity Diagram Menyimpan Hasil Perhitungan

Setelah mendapatkan alternatif terbaik / saham terbaik maka *user* dapat menyimpan data tersebut ke dalam *database*. Masukkan nama judul dari perhitungan ini pada *form*. Setelah itu tekan result pada navigasi dan tekan tombol *save* (Jika belum *login* tombol *save* tidak ada).

Class Diagram

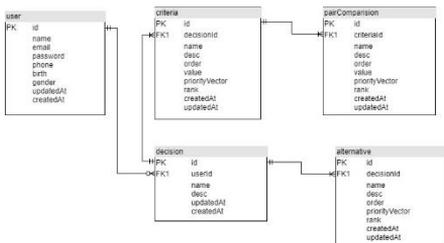
Class diagram menjelaskan struktur dari suatu sistem yang digambarkan dengan kelas – kelas dan hubungannya satu sama lain. Terdiri dari nama kelas, atribut, dan operasi/metode [6].



Gambar 9. *Class Diagram* Sistem Pendukung Keputusan Pemilihan Saham Terbaik.

Entity Relationship Diagram

Dalam sistem ini terdapat hubungan antar tabel yang saling berkaitan. Ditandai dengan adanya *Primary Key* (PK) dan *Foreign Key* (FK) pada tabel dapat digambarkan sebagai berikut :



Gambar 10. ERD Sistem Pendukung Keputusan Pemilihan Saham Terbaik

Rancangan User Interface

Merupakan tampilan visual sebuah produk yang menjembatani sistem dengan pengguna (*user*). Tampilan UI dapat berupa bentuk, warna, dan tulisan yang didesain semenarik mungkin. Secara sederhana, UI adalah bagaimana tampilan sebuah produk dilihat oleh pengguna.

a. Rancangan UI *Login Page*

Pada halaman ini terdiri dari *textfield* untuk memasukkan email dan password ke dalam sistem. Terdapat link ke *register page* pada bagian bawah *form*.



Gambar 11. Rancangan UI *Login*

b. Rancangan UI *Home Page*

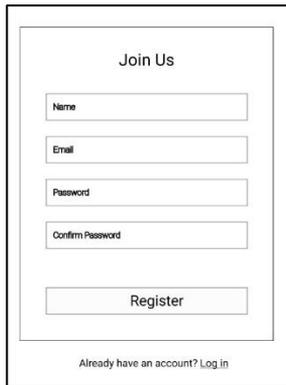
Halaman ini merupakan tampilan pertama saat memasuki *web*. Terdiri dari logo, *button login*, *button sign up* dan *start*. Jika *user* telah login maka *button login* dan *sign up* diganti menjadi *menu dropdown* yang terdiri dari tautan *profile*, *switch user*, *AHP*, dan *log out*.



Gambar 12. Rancangan UI *Home Page*

c. Rancangan UI *Register Page*

Pada halaman ini pengguna dapat mendaftar ke dalam sistem sebagai *user* yang dikenali. Terdiri dari *textfield name*, *email*, *password*, dan *confirm password*. Terdapat tautan ke halaman login pada bagian bawah formulir.



Gambar 13. Rancangan UI Register Page

d. Rancangan Interface Profile Page

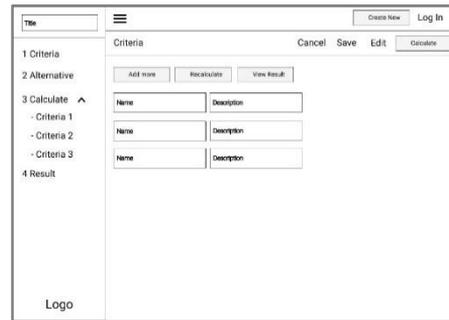
Pada halaman ini user dapat mengganti profil. Terdiri dari *textfield* name, email, phone number, gender, birth. Untuk mengantinya, user menekan button edit yang membuat semua *textfield* menjadi bisa diganti (awalnya dalam kondisi *disable*). Setelah itu user menekan *button save* yang membuat *textfield* menjadi kondisi *disable*.



Gambar 14. Rancangan UI Interface Profile Page

e. Rancangan UI Criteria Page

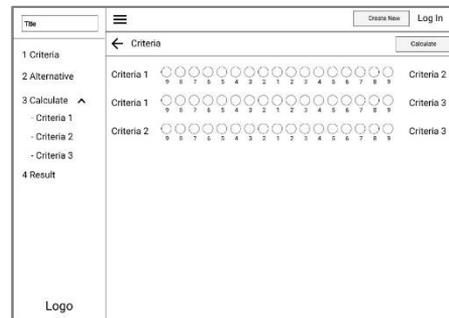
Halaman ini merupakan *form* untuk memasukkan nama kriteria dan keterangannya. Untuk ke halaman ini user dapat menekan tautan yang ada di sebelah kiri(1 kriteria).



Gambar 15. Rancangan UI Criteria Page

f. Rancangan UI Criteria Calculate Page

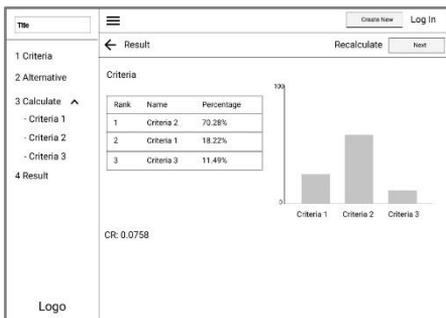
Pada halaman setelah user akan menghitung nilai *eigen* dengan memasukkan nilai *radio button* seperti pada gambar dibawah. Tanda panah untuk kembali ke halaman kriteria dan *button calculate* untuk ke halaman *result*.



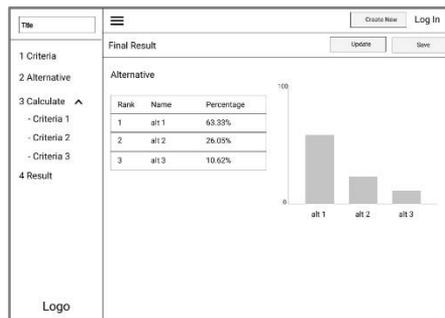
Gambar 16. Rancangan UI Criteria Calculate Page

g. Rancangan UI Criteria Result Page

Pada halaman ini sistem akan menghitung nilai *eigen* dan menampilkan hasil dalam bentuk tabel dan grafik. Jika user ingin mengulangi perhitungan maka tekan *button recalculate*. Tekan *button next* untuk ke halaman alternatif.

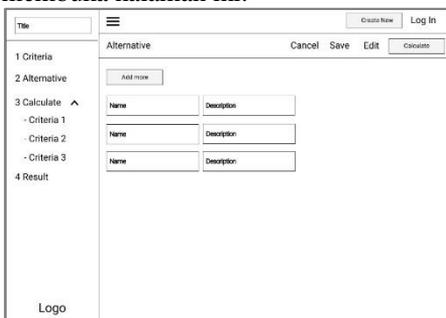


Gambar 17. Rancangan UI *Criteria Result Page*



Gambar 19. Rancangan UI *Final Result Page*

h. Rancangan UI *Alternative Page*
 Pada halaman ini user memasukkan alternatif / saham yang ingin dicari. Dimana fungsi dan inputannya sama dengan kriteria dan pilih tautan 2 alternatif pada sebelah kiri navigasi untuk membuka halaman ini.



Gambar 18. Rancangan UI *Alternative Page*

i. Rancangan UI *Final Result Page*
 Pada halaman ini *user* dapat melihat hasil akhir dari perhitungan dimana mendapatkan alternatif terbaik / saham terbaik. Dimana ditampilkan dalam bentuk tabel dan grafik. Jika sudah *login* maka *button save* akan muncul. Jika sudah pernah di *save* maka *button save* diganti menjadi *button update*. Edit *textfield title* yang merupakan judul dari perhitungan ini.

HASIL DAN PEMBAHASAN Perhitungan Metode AHP

Perhitungan metode AHP secara manual yaitu dengan menghitung bobot prioritas kriteria dan alternatif. Kriteria merupakan komponen utama untuk menentukan hasil perhitungan. Sedangkan alternatif adalah komponen yang akan dibandingkan. Berikut merupakan perhitungan metode AHP secara manual:

1. Kriteria yang akan digunakan yaitu:
 C1 = PER; C2 = PBV; C3 = DER
 C4 = ROE; C5 = ROA
2. Alternatif yang akan digunakan yaitu
 - a. A1 = CPIN (PT Charoen Pokphand Indonesia Tbk)
 - b. A2 = JFPA (PT Japfa Comfeed Indonesia Tbk)
 - c. A3 = MAIN (PT Malindo Feedmill Tbk)
3. Masukkan nilai pada matrik perbandingan berpasangan pada kriteria seperti pada tabel dibawah dengan cara:
 - a. Memasukkan nilai 1 jika kriteria yang dibandingkan sama atau memiliki kepentingan yang sama.
 - b. Masukkan nilai antar 2 sampai 9 (sesuai dengan bobot kepentingan) pada tabel dengan membandingkan baris terhadap kolom. Sebaliknya jika kriteria yang dibandingkan sama namun berbeda posisinya maka nilainya

yaitu 1/ (nilai bobot sebelumnya).
 Contohnya jika C1 terhadap C2
 yaitu 5, maka C2 terhadap C1
 yaitu 1/5.

Tabel 1 Matrik Perbandingan Berpasangan Kriteria

Kriteria	C1	C2	C3	C4	C5
C1	1	1	1/3 = 0.3333	1/5 = =0.2	1/2 = =0.5
C2	1	1	1/3 = 0.3333	1/5 = =0.2	1/2 = =0.5
C3	3	3	1	1/5 = =0.2	3
C4	5	5	5	1	3
C5	2	2	1/3 = 0.3333	1/3 = 0.3333	1

4. Jumlahkan nilai pada setiap kolom pada tabel 1 seperti dibawah ini:

C1: 1+1+3+5+2 = 12
 C2: 1+1+3+5+2 = 12
 C3: 0.3333+0.3333+1+5+0.3333 = 7
 C4: 0.2+.02+0.2+1+0.3333 = 1.9333
 C5: 0.5+0.5+3+3+1 = 8

5. Normalisasi setiap elemen pada tabel 1 dengan cara membagikan setiap elemen dengan jumlah nilai total pada setiap kolom (bagian 4) yaitu

Normalisasi kolom C1:

C1 = 1/12 = 0.0833
 C2 = 1/12 = 0.0833
 C3 = 3/12 = 0.25
 C4 = 5/12 = 0.4167
 C5 = 2/12 = 0.1667

Normalisasi kolom C2:

C1 = 1/12 = 0.0833
 C2 = 1/12 = 0.0833
 C3 = 3/12 = 0.25
 C4 = 5/12 = 0.4167
 C5 = 2/12 = 0.1667

Normalisasi kolom C3:

C1 = 0.3333/7 = 0.0476
 C2 = 0.3333/7 = 0.0476
 C3 = 1/7 = 0.1429
 C4 = 5/7 = 0.7143
 C5 = 0.3333/7 = 0.0476

Normalisasi kolom C4:

C1 = 0.2/1.9333 = 0.1034
 C2 = 0.2/1.9333 = 0.1034
 C3 = 0.2/1.9333 = 0.1034
 C4 = 1/1.9333 = 0.5172
 C5 = 0.3333/1.9333 = 0.1724

Normalisasi kolom C5:

C1 = 0.5/8 = 0.0625
 C2 = 0.5/8 = 0.0625
 C3 = 3/8 = 0.3750
 C4 = 3/8 = 0.3750
 C5 = 1/8 = 0.125

Tabel 2 Hasil Matrik Normalisasi Kriteria

Kriteria	C1	C2	C3	C4	C5
C1	0.083	0.083 3	0.047 6	0.103 4	0.062 5
C2	0.083	0.083 3	0.047 6	0.103 4	0.062 5
C3	0.25	0.25	0.142 9	0.103 4	0.375
C4	0.416	0.416 7	0.714 3	0.517 2	0.375
C5	0.166 7	0.166 7	0.047 6	0.172 4	0.125

6. Setelah normalisasi, hitunglah priority vector dengan cara membagi jumlah semua elemen pada tiap baris dengan jumlah kriteria.

PV C1 : ((0.0833+0.0833+0.0476 + 0.1034+ 0.0625))/5=0.0760

PV C2 : ((0.0833+0.0833+0.0476 + 0.1034+ 0.0625))/5=0.0760

PV C3 : ((0.25+0.25+0.1429 + 0.1034 + 0.375))/5=0.2243

PV C4 : ((0.4167+0.4167+0.7143 + 0.5172 + 0.375))/5=0.4880

PV C5 : ((0.1667+0.1667+0.0476 + 0.1724 + 0.125))/5=0.1357

7. Dari hasil perhitungan priority vector, C4 merupakan bobot tertinggi dalam menentukan saham terbaik, selanjutnya C3, C5, C1 dan C2.

Tabel 3. Priority Vector Kriteria

Kriteria	Priority Vector	Persentase	Rank
C4	0.4880	48.8%	1
C3	0.2243	22.43%	2

C5	0.1357	13.57%	3
C1	0.0760	7.6%	4
C2	0.0760	7.6%	5

Tabel 4. *Random Index*

n	2	3	4	5	6	7
RI_n	0	0,58	0,90	1,12	1,24	1,32

8. Setelah mendapatkan *priority vector* pada setiap kriteria, perlu dicek apakah bobot yang dihitung konsisten atau tidak. Dengan cara yaitu :

a. Hitung lamda maksimum dengan persamaan dibawah ini :

$$\lambda \max = (A)(W^T)$$

b. Hitung *consistency index* (CI), dimana n adalah jumlah kriteria
 $CI = (\lambda \max - n)/(n - 1) = (5.4237 - 5)/(5 - 1) = 0.1059$

c. Tetapkan nilai RI yang diambil dari tabel 4 dengan n = 5, maka: RI = 1.12

d. Hitung rasio konsistensi (CR) dengan persamaan:
 $CR = CI/RI = 0.1059/1.12 = 0.0946$
 karena jika dari CR < 0.1 maka ketidak konsistenan masih diterima dan jika lebih besar dari 0.1 maka ulangi dari Langkah 3

9. Selanjutnya hitunglah *priority vector* dari alternatif pada setiap kriteria. Dengan langkah yang sama pada perhitungan *priority vector* pada kriteria yaitu:

a. Perbandingan antar alternatif pada kriteria C1

Tabel 5. Matrik Perbandingan Berpasangan Alternatif pada Kriteria C1

	A1	A2	A3
A1	1	1/5 = 0.2	1/2 = 0.5
A2	5	1	4
A3	2	1/4 = 0.25	1
Jumlah	8	1.45	5.5

Tabel 6. Hasil Matriks Normalisasi Alternatif pada Kriteria C5

	A1	A2	A3
A1	0.7975	0.8235	0.7500

A2	0.1139	0.1176	0.1667
A3	0.0886	0.0588	0.0833

Tabel 7. *Priority Vector Alternatif* pada Kriteria C5

Alternatif	Priority Vector	Persentase	Peringkat
A1	0.7903	79.03%	1
A2	0.1327	13.27%	2
A3	0.0769	7.69%	3

$$\lambda \max = \begin{bmatrix} 1 & 7 & 9 \\ 0.1429 & 1 & 2 \\ 0.1111 & 0.5 & 1 \end{bmatrix} * \begin{bmatrix} 0.7903 \\ 0.1327 \\ 0.0769 \end{bmatrix} = 3.0425$$

$$CI = \frac{3.0425 - 3}{3 - 1} = 0.0212$$

$$CR = \frac{0.0212}{0.58} = 0.0366$$

10. Setelah mendapatkan *priority vector* setiap kriteria dan alternatif tiap kriteria. Hitunglah *priority vector alternatif* untuk mendapatkan alternatif yang terbaik dengan cara mengalikan PV alternatif tiap kriteria dengan PV kriteria yaitu :

$$\begin{bmatrix} 0.1179 & 0.0637 & 0.8004 & 0.7235 & 0.7903 \\ 0.6806 & 0.2674 & 0.0753 & 0.1932 & 0.1327 \\ 0.2014 & 0.6689 & 0.1244 & 0.0833 & 0.0769 \end{bmatrix} * \begin{bmatrix} 0.0760 \\ 0.0760 \\ 0.2243 \\ 0.4880 \\ 0.1357 \end{bmatrix} = \begin{bmatrix} 0.6536 \\ 0.2013 \\ 0.1452 \end{bmatrix}$$

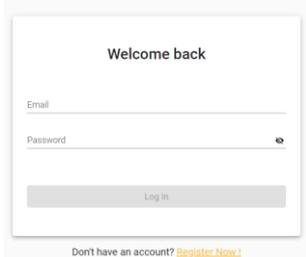
Tabel 8. *Priority Vector Alternatif*

Alternatif	Priority Vector	Persentase	Peringkat
A1	0.6536	65.36%	1
A2	0.2013	20.13%	2
A3	0.1452	14.52%	3

Dari hasil perhitungan diatas, A1 memiliki nilai yang tertinggi. Sehingga A1 (CPIN) merupakan saham terbaik dari ketiga alternatif yang dibandingkan.

Tampilan website pemilihan saham Interface Login Page

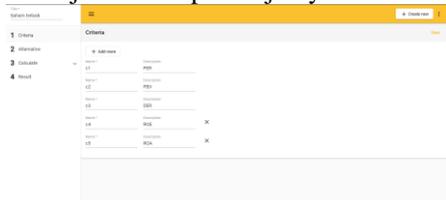
Pada halaman ini untuk pengguna yang sudah melakukan pendaftaran dan pengguna harus login terlebih dahulu agar dapat menyimpan hasil perhitungan.



Gambar 20. Interface Login Page

Interface Criteria Page

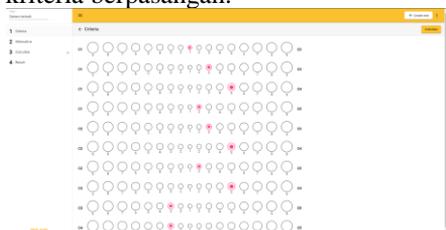
Halaman ini yang digunakan untuk memasukkan, mengubah dan menghapus kriteria. Setelah menyimpan kriteria akan muncul *button calculate* untuk melanjutkan tahap selanjutnya.



Gambar 21. Interface Criteria Page

Interface Criteria Calculate Page

Halaman ini pengguna bisa memasukkan nilai kepentingan terhadap perbandingan kriteria berpasangan.



Gambar 22. Interface Criteria Calculate Page

Interface Criteria Result Page

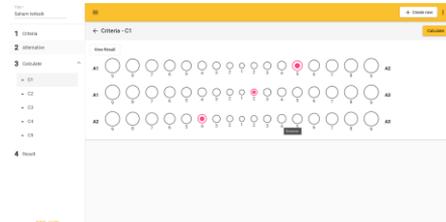
Jika perhitungan berhasil dan konsisten maka akan lanjut di halaman ini. Pada halaman ini pengguna dapat melihat tabel dan grafik dari hasil perhitungan sebelumnya dan nilai dari *consistency ratio*



Gambar 23. Interface Criteria Result Page

Interface Alternatif Calculate per Criteria Page

Sama dengan halaman kriteria dimana pengguna memasukkan nilai kepentingan pada perbandingan pasangan alternatif tiap kriteria. Jika sudah pernah dihitung maka *button view result* akan muncul.



Gambar 24. Interface Alternatif Calculate per Criteria Page

Interface Final Result Page

Pada halaman ini pengguna akan mendapatkan hasil terakhir perhitungan yaitu alternatif terbaik / saham terbaik dimana pengguna harus menghitung nilai pada kriteria dan juga alternatif. Jika pengguna sudah *login* maka dapat melakukan penyimpanan kedalam sistem dengan klik *button save*.



Gambar 25. Interface Final Result Page

KESIMPULAN

Dari hasil pembahasan maka dapat ditarik beberapa kesimpulan yaitu :

1. Sistem Pendukung Keputusan yang dibangun dapat membantu pengguna

untuk memilih saham berdasarkan analisis fundamental dengan lebih mudah. *Website* yang dibangun menggunakan *Framework Angular* sebagai *front-end* dan *Express* sebagai *back-end*

2. Metode AHP yang diimplementasikan pada website dapat membantu dalam pencarian alternatif dimana hasil perhitungannya sama dengan perhitungan manual. Sehingga sistem yang telah dibangun dapat membantu pengguna untuk lebih cepat dalam melakukan perhitungan dan mendapatkan hasil.

Komputer Volume 6 Nomor 1, 2020.

- [5] F. Sonata, V. W. Sari, Pemanfaatan UML (Unified Modeling Language) Dalam Perancangan Sistem Informasi *E-Commerce Jenis Customer-To-Customer*, Jurnal Komunika Volume 8 Nomor 1, 2019
- [6] A. Hendini, Pemodelan UML Sistem Informasi Monitoring Penjualan dan Stok Barang (Studi Kasus : Distro Zheza Pontianak), Jurnal Khatulistiwa Volume IV Nomor 2, 2016.

DAFTAR PUSTAKA

- [1] D. Nababan, Prioritas perbaikan jalan di kota Medan dengan menggunakan metode *Analitycal Hierarchy Process* (AHP), Jurnal Mandiri Bina Prestasi Volume 3 Nomor 2.
- [2] A. B. Paksi, Penerapan F-TOPSIS Dalam Analisis Fundamental Berdasarkan Rasio Keuangan Untuk Menunjang Keputusan Berinvestasi Saham. Seminar Nasional Teknologi Informasi dan Multimedia, 4(1), 3-3, 2016.
- [3] M. Astradanta, I. M. Wirawan, & I. K. Arthana, Pengembangan Sistem Penunjang Keputusan Pemilihan Tempat Kuliner Dengan Menggunakan Metode AHP Dan SAW Studi Kasus : Kecamatan Buleleng. Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika.
- [4] Amrin, M. D. Larasati, & I. Satriadi, Model Waterfall Untuk Pengembangan Sistem Informasi Pengolahan Nilai Pada SMP Kartika XI-3 Jakarta Timur, Jurnal Teknik

PREDIKSI KESEMBUHAN PASIEN COVID-19 DI INDONESIA MELALUI TERAPI MENGGUNAKAN METODE NAÏVE BAYES

Okky Putra Barus^{1*)}, Anton Tehja²⁾

¹Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pelita Harapan

E-mail : okky.barus@uph.edu¹⁾

²Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan

E-mail : at70004@student.uph.edu²⁾

**Penulis Korespondensi*

Abstract – This study aims to predict the recovery of COVID-19 patients in Indonesia by using Data Mining calculations. The method used to predict the recovery of COVID-19 patients is the Naïve Bayes method. The collection of datasets through trusted sources, the NIHR Innovation Observatory and datasets on an international/global scale, totaling 367 pieces of raw data that have not been filtered. After conducting the data feasibility test, the remaining 286 pieces of data will be divided into 70% of training data of 200 pieces of data and 30% of testing data of 86 pieces of data. Based on the test results, the use of the Naïve Bayes method in predicting the recovery of COVID-19 patients obtained an Accuracy of 96.51%, a Success Precision (Yes) of 100% and a Failure (No) of 95.71%, and a Success Sensitivity (Yes) of 84.21% and Failed (No) by 100%. Therefore, it is concluded that calculations using the Naïve Bayes method in this study will produce an accuracy rate of COVID-19 recovery of 96.51%, which means that the results of the predictions' calculation of success and failure in a therapy given to patients can be accounted for as data reference in a more detailed subsequent research..

Keywords: COVID-19, data mining, Naïve Bayes, prediction

Abstrak – Penelitian ini bertujuan untuk memprediksi kesembuhan pasien COVID-19 di Indonesia dengan menggunakan perhitungan Data Mining. Metode yang digunakan untuk memprediksi kesembuhan pasien COVID-19 adalah metode Naïve Bayes. Pengumpulan dataset melalui sumber yang terpercaya, NIHR Innovation Observatory dan dataset bersifat skala internasional/global yang berjumlah 367 buah data mentah yang belum dilakukan penyaringan. Setelah melakukan uji kelayakan data, yang tersisa adalah 286 buah data yang akan dibagi menjadi 70% data training sebesar 200 buah data dan 30% data testing sebesar 86 buah data. Berdasarkan hasil pengujian, penggunaan metode Naïve Bayes dalam memprediksi kesembuhan pasien COVID-19 mendapatkan Akurasi sebesar 96,51%, Presisi Sukses (Yes) sebesar 100% dan Gagal (No) sebesar 95,71%, dan Sensitifitas Sukses (Yes) sebesar 84,21% dan Gagal (No) sebesar 100%. Maka dari itu, diperoleh kesimpulan perhitungan menggunakan metode Naïve Bayes pada penelitian ini akan menghasilkan tingkat keakuratan kesembuhan COVID-19 sebesar 96,51% yang artinya hasil perhitungan

Diterima <22062021>, Revisi <24062021>, Diterima untuk publikasi <26072021>.

Copyright © 2021 Published by PSDKU Sistem Informasi UPH Kampus Medan,

ISSN : 2528-5114

prediksi sukses dan gagalnya dalam suatu terapi yang diberikan kepada pasien dapat dipertanggungjawabkan sebagai acuan data dalam penelitian berikutnya yang lebih mendetail.

Kata Kunci: COVID-19, data mining, Naïve Bayes, prediksi

PENDAHULUAN

Pneumonia yang belum diketahui secara pasti oleh ilmuwan penyebabnya terdeteksi pertama kali di Wuhan, Cina, dilaporkan ke *World Health Organization* (WHO) di Cina pada 31 Desember 2019. Sejak itu, jumlah kasus virus Corona semakin meningkat seiring dengan tingginya angka kematian. Virus Corona menyebar dari satu kota ke seluruh negeri hanya dalam waktu 30 hari, pada 11 Februari 2020 lalu dinamai COVID-19 oleh WHO.

Selama satu tahun terakhir, virus Corona telah menyebar dengan cepat ke seluruh belahan dunia. Para ilmuwan terus mengembangkan obat untuk melawan penyakit ini. Karena jumlah kasus yang diuji untuk virus Corona meningkat dengan cepat dari hari ke hari, tidak mungkin untuk melakukan pengujian karena faktor waktu dan biaya. Selama beberapa tahun terakhir, *machine learning* dan *data mining* telah menjadi sangat berguna hingga menjadi solusi di bidang medis. Menggunakan teknik *machine learning* dan *data mining* untuk memprediksi COVID-19 pada pasien akan mempersingkat waktu tunda hasil tes medis dan mengorganisir petugas kesehatan untuk memberikan perawatan medis yang tepat kepada mereka.

Data mining memainkan peran penting dalam mencegah penyebaran virus ini. Karena peran ahli epidemiologi perawatan kesehatan telah berkembang, data kesehatan berbentuk *soft-copy* (elektronik) yang tersebar juga semakin meluas. Peningkatan ketersediaan data kesehatan elektronik memberikan peluang besar dalam perawatan kesehatan baik untuk penemuan maupun

pengaplikasian untuk meningkatkan perawatan kesehatan. Data ini dapat digunakan untuk melatih algoritma *data mining* guna menghitung prediksi tingkat kesembuhan pasien COVID-19 di Indonesia dalam hal memprediksi kasus penyakit yang akan datang.[1]

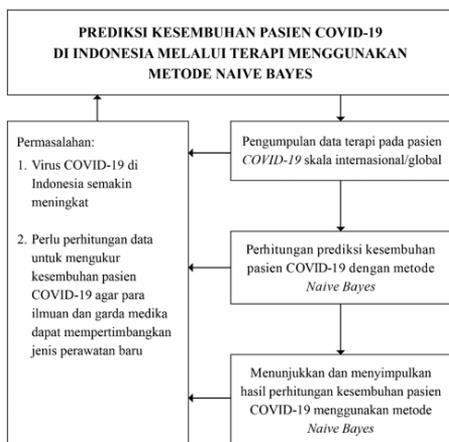
Tingkat kesembuhan pada seorang pasien COVID-19 sangat dipengaruhi banyak oleh beberapa faktor, seperti faktor lingkungan, *antibody*, nutrisi, kegiatan sehari-hari, obat yang dikonsumsi, frekuensi olahraga, dll. Faktor yang akan dibahas pada penelitian ini adalah faktor terapi obat yang diaplikasikan kepada pasien COVID-19, maka dari itu pada penelitian ini lebih difokuskan pada terapi penyembuhan melalui obat-obatan, untuk faktor-faktor lain yang mempengaruhi tidak akan dibahas pada penelitian ini.

Jenis terapi yang diberikan kepada pasien pada dataset penelitian ini terdapat dua yaitu Monoterapi dan Kombinasi (*Monotherapy and Combination*). Dimana Monoterapi, berarti hanya pasien yang diobati dengan obat antibiotik tunggal (satu jenis saja) yang diberikan, sedangkan pada kelompok terapi Kombinasi, pasien yang diobati dengan dua atau lebih jenis antibiotik yang diberikan secara bersamaan pada satu waktu.[2]

Hingga 8 Juni 2021, sebesar 173,694,649 kasus COVID-19 telah terdaftar secara skala global dan sebesar 1,869,325 kasus telah terdaftar di Indonesia beserta jumlah kematian mencapai 3,739,274 jiwa secara skala global dan sebanyak 51,992 jiwa di Indonesia dan jumlah pasien yang telah pulih sebesar 1,377,378,191 jiwa secara skala global

dan sebesar 22,536,306 jiwa di Indonesia [5]. Maka dari itu penelitian ini akan berfokus untuk menemukan terapi terbaik untuk menyembuhkan pasien COVID-19 berdasarkan data yang tersedia sehingga diharapkan dapat memberikan rekomendasi terapi jenis apa yang terbaik bagi pasien COVID-19.

METODE PENELITIAN



Penelitian ini dimulai dari pengumpulan data secara mendasar menggunakan metode studi Pustaka (dari Internet) untuk kasus COVID-19 di berbagai belahan dunia. Sumber yang didapatkan melalui situs *NIHR Innovation Observatory*, dimana *NIHR Innovation Observatory* merupakan fasilitas pemindaian medis nasional yang berlokasi di Universitas Newcastle[4]. Hasil data yang dikumpulkan merupakan dataset yang berisi jenis terapi / *treatment* apa saja yang telah diujikan kepada pasien pengidap virus COVID-19 dan hasil terapi / *treatment* nya berhasil atau tidak.

Dari data-data yang sudah terkumpul, maka dapat digunakan untuk perhitungan berikutnya untuk memprediksi apakah seorang pasien dapat sembuh menggunakan terapi / *treatment* tertentu. Penjelasan mengenai variabel dari dataset sebagai berikut:

1. *Intervention* – sebuah proses atau tindakan yang menjadi fokus studi klinis
 2. *Intervention Classification* – pengelompokan dari jenis *intervention* yang diterapkan
 3. *Developer and Licensing Status* – Tempat dimana penanganan berlangsung beserta dengan izin penanganannya
 4. *Trial ID* – Identitas dari pasien
 5. *Locations* – Asal negara pasien
 6. *In Monotherapy / Combination* – Jenis perawatan yang hanya diberikan 1 jenis obat saja untuk pengobatannya (*Monotherapy*) dan diberikan lebih dari 2 jenis obat secara bersamaan untuk pengobatannya (*Combination*)
 7. *Primary Completion Date* – Tanggal perawatan selesai
 8. *Intervention Type* – Jenis penanganan yang diterapkan apakah terapi menyembuhkan penyakit atau pencegahan
 9. *Trial phase* – Tahap uji klinis yang mempelajari obat atau produk biologis, berdasarkan definisi yang dikembangkan oleh Food and Drug Administration (FDA) US. Tahapan tersebut didasarkan pada tujuan studi, jumlah peserta, dan karakteristik lainnya. Ada lima fase diantaranya, Fase Awal 1 (sebelumnya terdaftar sebagai Fase 0), Fase 1, Fase 2, Fase 3, dan Fase 4. Tidak berlaku digunakan untuk menggambarkan uji coba tanpa fase yang ditentukan FDA, termasuk uji coba perangkat atau intervensi perilaku.
 10. *Registration Date* – Tanggal pendaftaran penanganan
 11. *Trial Registry – Platform* pendaftaran penanganan
 12. *Results* – Hasil dari penanganan
 13. *Status* – Status dari perekrutan penanganan
 14. *URL hyperlink* – Link detail dari penanganan setiap pasien
- Setelah mengumpulkan *dataset*, maka akan dilakukan proses *Data Selection* dimana terdapat variabel yang tidak diperlukan pada penelitian ini, sehingga

akan di-*filter* kembali sesuai dengan variabel yang dibutuhkan nantinya. Karena tujuan penelitian ini ialah untuk memprediksi apakah dengan suatu jenis penanganan bisa menyembuhkan pasien yang mengidap virus COVID-19 atau tidak, maka dari itu ada beberapa variabel yang kurang diperlukan pada penelitian ini yang bersifat sangat mendetil.

Adapun beberapa variabel yang tidak terpakai sebagai berikut:

1. *Developer and Licensing Status* – variabel ini berisi organisasi yang melakukan penanganan ini dan surat izin tidak diperlukan karena pada penelitian ini hanya membahas mengenai jenis penanganannya saja. Jikalau jenis suatu penanganan berhasil pada pasien di suatu tempat, maka penanganan itu juga akan berhasil di tempat asal manapun.

2. *Trial ID* – variabel ini berisi identitas dari pasien yang tidak diperlukan pada penelitian ini.

3. *Locations* – variabel ini berisi kewarganegaraan pasien yang bukan bagian dari penelitian ini. Jikalau jenis suatu penelitian berhasil pada pasien, maka penanganan itu juga akan berhasil di negara manapun.

4. *Primary Completion Date* – variabel ini berisi kapan pasien selesai ditangani, dimana pada penelitian ini tidak membahas mengenai berapa lama penanganan akan diberikan kepada pasien, melainkan hanya prediksi kesembuhannya saja.

5. *Trial phase* – variabel ini berisi fase ke- berapa penanganan ini dilakukan dan tidak diperlukan karena pada penelitian ini tidak membahas mengenai fase penanganan dilakukan.

6. *Registration Date* – variabel ini berisi kapan pendaftaran pasien untuk penanganannya dan juga tidak diperlukan karena pada penelitian ini tidak membahas durasi penanganannya.

7. *Trial Registry* – variabel ini berisi detail dari penanganan pasien ditangani oleh organisasi apa dan tidak diperlukan

karena pada penelitian ini tidak membahas mengenai asal organisasi penanganannya.

8. *Status* – variabel ini berisi status perekrutan pasien sudah di tahap mana dan tidak diperlukan karena pada penelitian ini semua dataset yang dipakai sudah ter-*filter* menggunakan status “*Completed*” sehingga tidak perlu dicantumkan kembali lagi.

9. *URL hyperlink* – variabel ini berisi situs yang menunjukkan detail dari setiap penanganan dari pasien yang tidak diperlukan karena pada penelitian ini hanya membahas mengenai prediksi kesembuhannya saja.

Variabel yang dipakai untuk penelitian ini sebagai berikut:

1. *Intervention* – variabel ini berisi deskripsi secara detil pengobatan apa yang digunakan dan diperlukan karena pada penelitian ini membahas kesembuhan pasien yang didasarkan dengan jenis obat-obatan yang dipakai.

2. *Intervention Classification* – variabel ini berisi jenis pengelompokan penanganannya yang lebih terstruktur datanya dibandingkan dengan hanya *Intervention* yang sangat detil. Sehingga cocok digunakan variabel ini untuk penelitian ini.

3. *In Monotherapy / Combination* – variabel ini berisi apakah terapi yang diberikan hanya satu jenis atau lebih dari satu jenis (kombinasi). Sehingga pada perhitungan penelitian ini bisa digunakan untuk mempengaruhi jumlah terapi yang diberikan ke pasien.

4. *Intervention Type* – variabel ini berisi jenis pengobatan yang diberikan itu apakah terapi dalam mengobati penyakit atau pencegahan terinfeksi penyakit. Sehingga mempengaruhi perhitungan prediksi kesembuhan pada penelitian ini.

5. *Results* – variabel ini berisi apakah terapi yang diberikan kepada pasien sukses atau tidaknya. Sehingga ini

menjadi output dari prediksi kesembuhan pada penelitian ini.

PERSIAPAN DATA

Berikut merupakan proses dalam menguji kelayakan tersebut:

1. *Data Selection* (Pemilihan Data)

Pemilihan Data merupakan proses pemilihan data-data yang berguna untuk digunakan nantinya terdapat beberapa variable-variabel yang tidak diperlukan dalam perhitungan nantinya seperti *Trial ID*, *Registration Date*, *Primary Completion Date*, *Status*, *Trial Phase*, dan *Locations*.

2. *Data Cleaning* (Pembersihan Data)

Pembersihan Data merupakan proses penghapusan kesalahan data, nilai yang hilang dan data yang tidak konsisten, pada penelitian ini telah dilakukan pembersihan data yang isinya tidak relevan yaitu pada isi variabel *Intervention Classification* yang berisi nilai "Others".

Setelah melakukan beberapa proses diatas, total jumlah dataset yang diperoleh menjadi sebanyak 286 buah data dari sebelumnya sebanyak 367 buah data.

PENGUJIAN DATA

Berikut merupakan rumus dari metode Naïve Bayes[3]:

$$P(Q|X) = \frac{P(X|Q) \cdot P(Q)}{P(X)}$$

Keterangan:

X : Data dengan kelas yang tidak diketahui

Q : Hipotesis *X* adalah kelas tertentu

$P(Q|X)$: Probabilitas hipotesis *Q* mengacu pada *X* (*posteriori probability*)

$P(Q)$: Probabilitas hipotesis *Q* (*prior probability*)

$P(X|Q)$: Probabilitas *X* dalam hipotesis *Q*

$P(X)$: Probabilitas *X*

Adapun langkah perhitungan probabilitas kesembuhan pasien yang mengidap virus COVID-19 yaitu:

1. Tahap 1:

Memisahkan 2 kelas menjadi *C1* untuk "Results = Yes" dan *C2* untuk "Results = No"

2. Tahap 2:

Mencari *Prior Probability* ($P(Q)$):

a. Probabilitas Results dengan hasil Yes

b. Probabilitas Results dengan hasil No

3. Tahap 3:

Mencari *Likelihood* ($Q1 \dots Qn$) dari setiap atribut

4. Tahap 4:

Mencari *Likelihood Probability* ($P(X|Q)$):

a. Probabilitas Terapi ($P(X)$) dengan hasil Results = Yes berdasarkan hasil penanganan pasien

b. Probabilitas Terapi ($P(X)$) dengan hasil Results = No berdasarkan hasil penanganan pasien

5. Tahap 5:

Mencari *Posteriori Probability* ($P(X|Q)$):

a. Mengalikan Probabilitas Terapi ($P(X)$) dengan hasil Results = Yes berdasarkan hasil penanganan pasien dari *Likelihood Probability* ($P(X|Q)$) dengan Probabilitas Results = Yes dari *Prior Probability* ($P(Q)$)

b. Mengalikan Probabilitas Terapi ($P(X)$) dengan hasil Results = No berdasarkan hasil penanganan pasien dari *Likelihood Probability* ($P(X|Q)$) dengan Probabilitas Results = No dari *Prior Probability* ($P(Q)$)

6. Tahap 6:

Membandingkan hasil dari *Posteriori Probability* ($P(Q|X)$) antara Results = Yes dan No. Nilai yang lebih besar merupakan hasil dari penanganan terapi yang diberikan kepada pasien apakah sukses atau tidak dengan menggunakan metode Naïve Bayes.

HASIL PENELITIAN

Pada bagian ini akan dibahas mengenai hasil yang diperoleh dari perhitungan dengan menggunakan pembagian data sebesar 70% untuk data training 200 buah dan 30% untuk data testing 86 buah.

Dari 70% data training sebanyak 200 buah data dan 30% data testing yaitu sebanyak 86 buah data, menghasilkan sebagian jumlah hasil prediksinya hampir menyerupai hasil aktualnya. Sehingga, kombinasi dari metode Naïve Bayes dengan data yang diperoleh tingkat akurasi adalah sebesar 96,51%.

Precision (Presisi)

Untuk perhitungan *Precision* (Presisi), didapatkan dari perbandingan jumlah hasil prediksi Sukses/Gagal (*Yes/No*) yang sama dengan hasil aktualnya, dibagi dengan total prediksi Sukses/Gagal (*Yes/No*).

Pada kasus Presisi Sukses (*Yes*), didapatkan 16 hasil prediksi Sukses (*Yes*) yang hasilnya sama dengan data aktualnya, dibagi dengan total hasil prediksi Sukses (*Yes*) sebanyak 16 data, maka akan memperoleh hasil sebagai berikut:

$$= \frac{16(\text{Prediksi Sukses (Yes) yang hasilnya sama dengan data aktual})}{16(\text{total prediksi Sukses (Yes)}} \times 100\% \\ = 100\%$$

Pada kasus Presisi Gagal (*No*), didapatkan 67 hasil prediksi Gagal (*No*) yang hasilnya sama dengan data aktualnya, dibagi dengan total hasil prediksi Gagal (*No*) sebanyak 70 data, maka akan memperoleh hasil sebagai berikut:

$$= \frac{67(\text{Prediksi Gagal (No) yang hasilnya sama dengan data aktual})}{70(\text{total prediksi Gagal (No)}} \times 100\% \\ = 95,71\%$$

Recall (Sensitifitas)

Untuk perhitungan *Recall* (Sensitifitas), didapatkan dari perbandingan jumlah hasil prediksi Sukses/Gagal (*Yes/No*) yang sama dengan hasil aktualnya, dibagi dengan total hasil Sukses/Gagal

(*Yes/No*) yang aktual dari data *testing* yang digunakan.

Pada kasus Sensitifitas Sukses (*Yes*), didapatkan 16 hasil prediksi Sukses (*Yes*) yang hasilnya sama dengan data aktualnya, dibagi dengan total hasil Sukses (*Yes*) yang aktual dari data testing yang digunakan sebesar 19 data, maka akan memperoleh hasil sebagai berikut:

$$= \frac{16(\text{Prediksi Sukses (Yes) yang hasilnya sama dengan data aktual})}{19(\text{total hasil Sukses (Yes) aktual di data testing})} \times 100\% \\ = 84,21\%$$

Pada kasus Sensitifitas Gagal (*No*), didapatkan 67 hasil prediksi Gagal (*No*) yang hasilnya sama dengan data aktualnya, dibagi dengan total hasil Gagal (*No*) yang aktual dari data testing yang digunakan sebesar 67 data, maka akan memperoleh hasil sebagai berikut:

$$= \frac{67(\text{Prediksi Gagal (No) yang hasilnya sama dengan data aktual})}{67(\text{total hasil Gagal (No) aktual di data testing})} \times 100\% \\ = 100\%$$

Accuracy (Akurasi)

Untuk perhitungan *Accuracy* (Akurasi), didapatkan dari jumlah hasil prediksi Sukses (*Yes*) yang sama dengan hasil aktualnya, dibagi dengan total data testing yang digunakan. Kemudian dijumlahkan dengan jumlah hasil prediksi Gagal (*No*) yang sama dengan hasil aktualnya, dibagi dengan total data testing yang digunakan.

Kemudian didapatkan 16 hasil prediksi Sukses (*Yes*) yang hasilnya sama dengan data aktualnya, dibagi dengan total data testing yang digunakan sebanyak 86 data, maka akan memperoleh hasil sebagai berikut:

$$= \frac{16(\text{Prediksi Sukses (Yes) yang hasilnya sama dengan data aktual})}{86(\text{total data testing})} \times 100\% \\ = 18,60\%$$

Dalam penelitian ini didapatkan 67 hasil prediksi Gagal (*No*) yang hasilnya sama dengan data aktualnya, dibagi dengan total data testing yang digunakan sebanyak 86 data, maka akan memperoleh hasil sebagai berikut:

$$= \frac{67(\text{Prediksi Gagal (No) yang hasilnya sama dengan data aktual})}{86(\text{total data testing})} \times 100\% = 77,91\%$$

Maka jika dijumlahkan dari masing-masing perbandingan yang telah diperhitungkan akan mendapatkan tingkat akurasi sebesar $= 18,60\% + 77,91\% = 96,51\%$.

Jenis terapi yang diberikan kepada pasien dalam *dataset* ini sangat variatif, sehingga pelatihan untuk data training akan sedikit lebih menantang karena luasnya perbedaan isi pada variabel Intervention, tetapi dengan adanya perbedaan yang luas maka semakin banyak alternatif dalam memprediksi tingkat kesembuhan pasiennya dengan jenis terapi yang berbeda-beda.

Berikut merupakan jenis terapi yang sukses menyembuhkan pasien secara data aktual yaitu jenis terapi *Acalabrutinib (Calquence)*, *Ampion*, *Anakinra (Kineret)*, *anti-SARS-CoV-2 IgY*, *ArtemiC*, *Asbrip*, *Aspirin*, *Astegolimab (RG-6149)*, *AT-001*, *Atovaquone*, *AV-COVID-19*, *Aviptadil (Zyesami)*, *Azithromycin*, *Bamlanivimab (LY3819253, LY-CoV555)*, *Bardoxolone Methyl (RTA 402)*, *Baricitinib (Olumiant)*, *Berberine*, *Bevacizumab (Avastin)*, *BGB-DXP593*, *Brensocatic (INS1007)*, *Brequinar*, *BRII-198*, *Bromhexine hydrochloride*, *BTL-TML001*, *Camostat mesylate*, *Canakinumab (Ilaris)*, *CAPEo (Cretan IAMA)*, *CD24Fc*, *CERC-002*, *Cetylpyridinium chloride*, *Chloroquine*, *Cholecalciferol (Vitamin D3)*, *Clarithromycin*, *Clindamycin*, *Colchicine*, dan *Convalescent plasma*.

Berikut merupakan hasil prediksi jenis terapi menyembuhkan pasien yang sukses menggunakan metode Naïve Bayes dengan 30% data testing sebanyak 86 data yang dipilih dari 286 data secara acak dan hasilnya sama dengan data aktualnya yaitu jenis terapi *Azithromycin*,

Doxycycline, *Acalabrutinib (Calquence)*, *Astegolimab (RG-6149)*, *Atovaquone*, *Bardoxolone Methyl (RTA 402)*, *Camostat mesylate*, *Canakinumab (Ilaris)*, *Chloroquine*, *Colchicine*, *Crizanlizumab (Adakveo)*, dan *Human Immunoglobulin*. Sehingga jenis terapi yang dihasilkan cukup variatif dan memiliki banyak alternatif dalam mengatasi penyakit COVID-19 jika digabungkan dengan data aktualnya.

Maka dari itu, diperoleh kesimpulan perhitungan menggunakan metode Naïve Bayes pada penelitian ini akan menghasilkan tingkat keakuratan kesembuhan COVID-19 sebesar 96,51% yang artinya hasil perhitungan prediksi sukses dan gagal dalam suatu terapi yang diberikan kepada pasien dapat dipertanggungjawabkan sebagai acuan data dalam penelitian berikutnya yang lebih mendetail. Tingkat keakuratan ini masih dapat ditingkatkan dengan cara memperoleh jumlah data yang lebih banyak lagi sehingga dapat meningkatkan akurasi ketika melakukan pengujian data.

KESIMPULAN

Dari hasil penelitian dan perhitungan yang dilakukan menggunakan metode Naïve Bayes, dapat ditarik beberapa kesimpulan seperti:

1. Hasil perhitungan *precision* (presisi) yang didapatkan dari perbandingan jumlah hasil prediksi Sukses/Gagal (Yes/No) yang sama dengan hasil aktualnya, dibagi dengan total prediksi Sukses/Gagal (Yes/No) menghasilkan nilai sebesar 100% untuk Presisi Sukses (Yes) dan 95,71% untuk Presisi Gagal (No).
2. Perhitungan *recall* (sensitifitas) yang didapatkan dari perbandingan jumlah hasil prediksi Sukses/Gagal (Yes/No) yang sama dengan hasil aktualnya, dibagi dengan total hasil Sukses/Gagal (Yes/No) yang aktual dari *data testing* yang

digunakan yang menghasilkan nilai sebesar 84,21% untuk Sensitifitas Sukses (Yes) dan 100% untuk Sensitifitas Gagal (No).

3. Perhitungan *Accuracy* (Akurasi) yang didapatkan dari jumlah hasil prediksi Sukses (*Yes*) yang sama dengan hasil aktualnya, dibagi dengan total *data testing* yang digunakan. Kemudian dijumlahkan dengan jumlah hasil prediksi Gagal (*No*) yang sama dengan hasil aktualnya, dibagi dengan total *data testing* yang digunakan. Maka akan menghasilkan tingkat keakuratan sebesar 96,51%.

Adapun saran untuk penelitian berikutnya adalah melibatkan faktor-faktor lain diluar faktor terapi seperti faktor lingkungan, *antibody*, nutrisi, kegiatan sehari-hari, obat yang dikonsumsi, frekuensi olahraga, dan sebagainya.

DAFTAR PUSTAKA

- [1] Matta, D. M., & Saraf, M. K. (2020). Prediction of COVID-19 using Machine Learning Techniques. Karlskrona: Blekinge Institute of Technology.
- [2] Adrian Schmid, A. W. (2019). Monotherapy versus combination therapy for multidrug-resistant Gram-negative infections: Systematic Review and Meta-Analysis. *Scientific Reports Nature Research*, 2.
- [3] Aji Prasetya Wibawa, A. C. (2019). Naïve Bayes Classifier for Journal Quartile Classification. *International Journal of Recent Contributions from Engineering Science & IT (iJES)*, 94-96.
- [4] Excellence, N. I. (2020). National Institute for Health and Care Excellence. Retrieved from National Institute for Health and Care Excellence: <https://www.nice.org.uk/covid-19/rapid-c19>
- [5] Gardner, L. (2020, January 23). Coronavirus COVID-19 (2019-nCov). Retrieved from COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU): <https://www.arcgis.com/apps/opsdashboard/index.html#/bda7594740fd40299423467b48e9ecf6>

KECERDASAN BUATAN UNTUK BLOKUS CLASSIC MENGGUNAKAN HEURISTIK, FLOODFILL, DAN ALGORITMA GREEDY

Hong Liang Cai¹⁾, Sebastian Aldi²⁾, Winston Renatan^{3*)}

¹⁾Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan
Email: liangcai.stdnt@gmail.com¹⁾

²⁾Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan
Email: sebastian.aldi17@gmail.com²⁾

³⁾Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pelita Harapan
Email: winstonrenatan@gmail.com³⁾

**Penulis Korespondensi*

Abstract – Blokus is an abstract strategy game which has complex variable to determine ones move. We propose an idea for Artificial Intelligence in Blokus using Heuristics, FloodFill, and Greedy algorithm, which will be called as LeakyAI. Both the game and its implementation are established in Java language. To test the result, we performed benchmarking towards Brute Force AI, itself, and the developers.

Keywords: *Blokus, Artificial Intelligence, Game Theory, Heuristic Search, FloodFill Algorithm, Greedy*

Abstrak – Blokus merupakan sebuah permainan strategi abstrak yang memiliki variabel kompleks untuk menentukan pergerakan pemain. Kami mengusulkan ide Kecerdasan Buatan pada Blokus dengan Algoritma Heuristik, FloodFill, dan Greedy, yang akan disebut LeakyAI. Baik permainan maupun implementasinya dibuat menggunakan bahasa Java. Untuk menguji hasilnya, kami melakukan perbandingan terhadap Kecerdasan Buatan Brute Force, dirinya sendiri, dan dengan para pengembang.

Kata Kunci: *Blokus, Kecerdasan Buatan, Teori Permainan, Pencarian Heuristik, Algoritma FloodFill, Greedy*

PENDAHULUAN

Blokus adalah permainan strategi abstrak yang dirancang oleh Bernard Tavitian dan diterbitkan pada tahun 2000. Permainan ini melibatkan empat pemain dengan warna berbeda, masing-masing dengan 21 potongan permainan berbentuk seperti tetris untuk diletakkan di papan persegi dengan ukuran 20x20 unit. Tujuan permainan ini adalah untuk

menempatkan semua potongan di papan, permainan dimulai di keempat sudut papan dan setiap bagian harus menyentuh hanya bagian sudut dari potongan warna yang sama (tanpa menyentuh potongan milik pemain sendiri di bagian samping). Poin akan dihitung dari persegi unit pemain yang tersisa (1 unit persegi = -1 poin), +15 poin akan diberikan jika pemain dapat menempatkan semua 21

buah, dan tambahan +5 poin jika bagian terakhir ditempatkan adalah persegi satu unit [1].

Kami mendesain ulang permainan menggunakan bahasa Java karena cukup berguna ketika berhadapan dengan pemrograman berorientasi objek, dimana desain permainan akan diuraikan nanti. Membangun permainan dan Kecerdasan Buatan untuk permainan papan ini cukup menantang dalam prosesnya. Di sini Kecerdasan Buatan berfungsi untuk memberikan cara untuk mencapai tujuan pada lingkungan yang kompleks [2], banyak faktor berkontribusi pada satu keputusan seperti potongan yang dimiliki, potongan pemain lain, kerja tim dari pemain lain, dan lainnya. Beberapa pendekatan telah digunakan di Blokus Duo (jenis Blokus lain dengan hanya dua pemain dan papan persegi 14x14 unit) [3] seperti MiniMax dengan Alpha-Beta Pruning [4] dan Monte Carlo Tree Search [5]. Ide yang kami usulkan di sini adalah menggunakan Heuristik, FloodFill, dan Algorithma Greedy untuk mengatasi masalah ini.

DESAIN PERMAINAN

Blokus memiliki tiga bagian utama dalam permainannya, yang secara umum terdiri dari papan, pemain, dan potongan-potongannya. Untuk mensimulasikan permainan papan nyata secara digital, kita perlu memahami dan mengerti setiap komponen permainan dengan cara yang lebih rinci termasuk bagaimana ia dapat bergerak dan keadaan apa yang membuat permainan berakhir.

Papan dan Pemain

Papan terdiri dari persegi berukuran 20x20 unit, di mana masing-masing persegi unit memiliki nomor dan warna mereka sendiri yang menunjukkan pemilik persegi tersebut. Jumlah dan warna persegi akan berubah setelah pemain meletakkan potongannya di persegi berwarna putih atau ruang kosong

di papan. Persegi berwarna tidak dapat diubah dalam kondisi apa pun (misalnya menempatkan di atas bagian lain, di luar batas, dan lainnya). Informasi terperinci diberikan dalam tabel berikut.

Tabel 1 Sistem Angka dan Pewarnaan

Warna atau Pemain	Angka
Putih	-1
Merah	0
Hijau	1
Biru	2
Kuning	3

Warna putih berarti saat ini persegi kosong (tidak dimiliki oleh pemain mana pun), sementara warna lainnya menunjukkan persegi tersebut dimiliki oleh pemain dengan warna yang sesuai. Warna dan penomoran tersebut diperlukan untuk mengetahui langkah mana yang mungkin dapat dilakukan oleh pemain dalam persegi tertentu. Pemain kemudian akan diinisialisasi di kelas Blokus menggunakan kode di bawah ini. LeakyAI atau Player dapat diubah ke AI lain yang akan ditempatkan di papan.

```

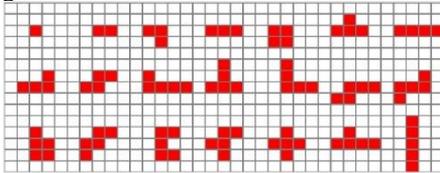
1 public static void main(String args[]) {
2     players[0] = new Player(0);
3     players[1] = new LeakyAI(1);
4     players[2] = new LeakyAI(2);
5     players[3] = new LeakyAI(3);
6     mainGui = new Gui(players, board);
7     mainGui.renderBoard();
8     setTurn(0);
9 }
    
```

Gambar 1 Kode Menginisialisasi Pemain di Papan

Potongan

Ada 21 buah potongan dalam permainan untuk setiap pemain yang didistribusikan dari 1 unit persegi hingga 5 unit persegi. Semua 21 potongan ini akan disimpan dalam bentuk berbagai matriks. Selain itu, potongan-potongan itu tidak statis, tetapi itu adalah bagian dinamis yang dapat diputar dan dicerminkan (*mirror*) oleh pemain untuk mencocokkan penggunaannya di papan permainan.

Semua potongan ditampilkan dalam gambar berikut.



Gambar 2 Semua Potongan di Blokus

Setiap bagian diwakili dengan cara menggunakan penomoran koordinat kartesius, mulai dari paling atas kiri. Sebuah potongan akan memiliki matriks[x][y] dan menyimpan nilai baik 1 atau 2. Masing-masing nilai memiliki interpretasinya sendiri, 1 berarti bahwa potongan hadir pada koordinat [x][y], sedangkan 2 berarti potongan hadir dan merupakan titik tengah potongan.

Tabel 2 Representasi Potongan

Potongan	Representasi
	matriks[0][0] = 1 matriks[0][1] = 2 matriks[0][2] = 1
	matriks[0][1] = 1 matriks[0][2] = 1 matriks[1][0] = 1 matriks[1][1] = 2
	matriks[0][1] = 1 matriks[0][2] = 1 matriks[1][0] = 1 matriks[1][1] = 2 matriks[2][1] = 1

Ketika ada rotasi, pencerminan secara horizontal atau vertikal yang dilakukan pada bagian tersebut maka harus ada penyesuaian yang harus dilakukan. Di pencerminan horizontal, kolom pertama akan menjadi kolom kelima dan kolom kedua akan menjadi kolom keempat, sedangkan kolom ketiga tetap sama. Pencerminan vertikal identik dengan yang horizontal, tetapi dilakukan pada baris. Rotasi agak berbeda di mana potongan pertama akan diubah urutannya

dan hasilnya akan dicerminkan horizontal, yang akan menjadi produk akhir.

```

1 public Integer[][] rotate() {
2     Integer newMatrix[][] = new Integer[5][5];
3     for (int i = 0; i < 5; i++) {
4         for (int j = 0; j < 5; j++) {
5             newMatrix[j][i] = matrix[i][j];
6         }
7     }
8     newMatrix = horizontalMirror(newMatrix);
9     matrix = newMatrix;
10    return newMatrix;
11 }
    
```

Gambar 3 Kode Fungsi Cermin Horizontal

```

1 public Integer[][] horizontalMirror(Integer[][] newMatrix) {
2     for (int i = 0; i < 5; i++) {
3         for (int j = 0; j < 3; j++) {
4             int temp = newMatrix[i][4-j];
5             newMatrix[i][4-j] = newMatrix[i][j];
6             newMatrix[i][j] = temp;
7         }
8     }
9     matrix = newMatrix;
10    return newMatrix;
11 }
    
```

Gambar 4 Kode Fungsi Cermin Vertikal

```

1 public Integer[][] verticalMirror(Integer[][] newMatrix) {
2     for (int i = 0; i < 5; i++) {
3         for (int j = 0; j < 3; j++) {
4             int temp = newMatrix[4-i][j];
5             newMatrix[4-i][j] = newMatrix[i][j];
6             newMatrix[i][j] = temp;
7         }
8     }
9     matrix = newMatrix;
10    return newMatrix;
11 }
    
```

Gambar 5 Kode Fungsi untuk Memutar

Status dan Gerakan Permainan

Pemain akan mengambil giliran untuk bergerak persis dalam urutan merah, hijau, biru, dan kuning kemudian kembali lagi ke pemain merah. Untuk memeriksa apakah pemain memiliki opsi untuk bergerak dalam persegi tertentu, teknik *Brute Force* akan dilakukan melalui pengecekan dan verifikasi potongan-potongan yang tersisa pada pemain dan potongan tersebut kemudian akan dicoba untuk diputar sebanyak empat kali, kemudian dicerminkan dan diputar lagi beberapa kali. Jika salah satu proses benar (bagian cocok di papan), maka fungsi akan mengembalikan bahwa pemain memiliki kemungkinan bergerak dan permainan belum berakhir dan dapat dilanjutkan. Praktik ini akan dilakukan untuk semua potongan yang tersisa dari para pemain, jika tidak ada potongan

yang cocok di papan maka pemain kehilangan giliran mereka.

```

1 IF piece is fully used → RETURN false
2 FOR i in board side:
3   FOR j in board side:
4     FOR ii in players piece:
5       IF piece is used → pass check
6       FOR 4:
7         rotate piece
8       IF valid move → RETURN true
9       horizontal mirror piece
10      FOR 4:
11        rotate piece
12      IF valid move → RETURN true
13 RETURN false
    
```

Gambar 6. Fungsi Algoritma hasMove

Sementara itu, fungsi *hasMove()* juga memanggil dan menggunakan *isValidMove()* untuk memeriksa apakah penempatan sebuah potongan tertentu dimungkinkan untuk dilakukan atau tidak. Kualifikasinya seperti: di luar batas atau papan, menumpuk kemungkinan pada potongan-potongan lain, giliran pertama pemain (harus berada di sudut papan), sisi potongan bertabrakan, dan tidak menempatkan potongan dengan secara diagonal.

N.B. "warning" pada Gambar 7, adalah panel yang menunjukkan bahwa langkah tertentu tidak dapat dilakukan dan bahwa pemain harus membuat langkah lain.

```

1 IF out of bounds → warning
2 IF stack other piece on placement → warning
3 IF player is on first turn:
4   IF place not in corner of board → warning
5 ELSE:
6   check [x-1][y-1], [x][y-1], [x+1][y-1], ..., [x+1][y+1]
7   IF there exist block in [x-1][y-1], [x+1][y-1], [x-1][y+1],
   [x+1][y+1] → warning touch diagonal
8   IF there exist block in [x][y-1], [x-1][y], [x+1][y], [x][y+1]
   → warning touch side
9 IF no warning → move is valid
    
```

Gambar 7. Fungsi Algoritma isValidMove

Akhir Permainan

Permainan berakhir baik saat tidak ada lagi gerakan yang tersisa untuk setiap pemain atau tidak ada lagi potongan yang tersisa untuk ditempatkan di papan permainan. Beberapa skenario akhir dari permainan seperti: semua pemain menempatkan semua potongan mereka, beberapa pemain meletakkan semua potongan mereka dan beberapa lainnya tidak memiliki langkah lagi, dan tidak ada lagi gerakan untuk setiap pemain. Sistem akan memeriksa bagian yang

tersisa yang dilakukan setiap pemain dan langkah apa yang diambil pemain terakhir. Akan ada sebuah tampilan yang menampilkan peringkat setiap pemain dengan poin mereka di akhir permainan.

Tabel 3 Sistem Penilaian

Kondisi	Poin
Untuk setiap potongan yang tersisa per satu unit persegi	-1
Semua potongan ditempatkan di atas papan	+15
Potongan terakhir yang ditempatkan adalah persegi satu unit	+5

STRATEGI PERMAINAN

Seperti yang disebutkan sebelumnya, kami menerapkan Heuristik, FloodFill, dan Algoritma Greedy dalam merancang Kecerdasan Buatan untuk permainan Blokus yang akan lebih diuraikan lebih lanjut pada bagian ini.

Fungsi Potongan

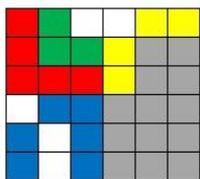
Ketika seorang pemain memulai langkah mereka, hanya akan ada sepotong yang akan digunakan pada giliran itu, tentu saja kita ingin memiliki potongan terbaik untuk ditempatkan di papan untuk menang. Umumnya, potongan terbaik untuk beberapa putaran pertama adalah menggunakan potongan-potongan yang terbesar. Ada beberapa fungsi yang dibuat untuk mengetahui mana potongan yang tersisa dari pemain adalah yang terbaik untuk ditempatkan. Kami akan memanggil fungsi eksternal dari kelas lain yang berfungsi untuk memeriksa apakah langkah tertentu valid atau tidak, bersama dengan melakukan iterasi dari bagian terbesar ke terkecil. Yang pertama adalah iterasi dan memeriksa potongan mana saja yang masih tersisa adalah dengan *getLargestAvailablePiece()*. Sementara itu, kami ingin juga mendapatkan nilainya dalam satuan unit persegi yang akan menggunakan fungsi *getPieceSize()*, algoritma akan iterasi

melalui blok potongan 5x5 dan memeriksa blok mana yang diisi dan mencatatnya.

Berikutnya adalah `largestMoveAt()` yang akan berfungsi untuk mengetahui dan menghitung potongan terbesar untuk ditempatkan dalam koordinat (misalnya dalam koordinat x, y bagian yang dapat ditempatkan adalah 5 unit persegi) fungsi ini akan mengembalikan bilangan bulat. Kemudian ada juga `largestPieceAt()` yang dapat dikatakan sangat mirip dengan `largestMoveAt()` tetapi ini akan mengembalikan potongan yang mana bukan nilai bilangan bulat. Kemudian, akhirnya kita tentu saja perlu mengetahui indeks potongan mana yang harus digunakan menggunakan fungsi `largestPieceIndexAt()`. Terdapat juga suatu kebutuhan untuk menciptakan kembali atau mengkloning papan karena kami ingin melakukan perhitungan FloodFill di papan kloning untuk setiap pemain alih-alih di papan utama di mana ia dapat diakses oleh semua pemain lain.

Algoritma FloodFill

Seperti apa yang telah disebutkan sebelum pada pendahuluan, hal pertama yang perlu kita miliki untuk pengambilan keputusan adalah algoritma FloodFill, yang kemudian akan banyak membantu. FloodFill akan membantu Kecerdasan Buatan yang diterapkan untuk menentukan apakah layak atau tidak untuk menempatkan sepotong ke ruang kosong dalam mendapatkan poin dengan jumlah tertentu. Representasi angka tentang cara kerja FloodFill dapat dilihat di bawah ini, pada Gambar 8.



Gambar 8 Sistem Keputusan FloodFill

Algoritma FloodFill akan diimplementasikan dengan algoritma lain yang dijelaskan kemudian. Dari gambar tersebut, misalnya Kecerdasan Buatan bermain sebagai Potongan Merah, FloodFill kemudian akan menyelidiki kotak abu-abu (ruang kosong) dan melihat nilainya apakah layak untuk meletakkan potongan di sana atau di tempat lain. Jumlah ruang kosong yang tersedia bersama dengan nilai heuristik internal lain yang dibahas nantinya akan menentukan nilai sebuah langkah. Algoritma FloodFill dapat dilihat pada Gambar 9.

```

1 FloodFill (board, x, y, points) {
2   IF unit square value is -1:
3     SET unit square value = -999
4     ADD points
5     check four sides → CALL FloodFill
6   RETURN points
7   ELSE:
8     RETURN points
9 }

```

Gambar 9 Algoritma FloodFill

Heuristik

Selain itu, kami juga datang dengan nilai heuristik untuk melengkapi algoritma yang ada. Pada awalnya kami memiliki pemikiran dengan dua jenis heuristik di papan, angka pada setiap kotak unit mewakili nilainya. Rencana awal yang kami adalah dengan tetap fokus untuk mendapatkan semua potongan ke tengah papan dan kemudian didistribusikan secara merata di antara papan permainan. Dengan membidik pusat (bagian tengah) papan sesegera mungkin, harapannya adalah akan ada lebih banyak kemungkinan bergerak dibandingkan dengan penempatan yang tetap berada di sudut awal. Tapi, model ini tidak bekerja dengan cara yang cukup memuaskan. Alasan di baliknya adalah karena ada kesempatan bagi seseorang untuk memenuhi papan ujung pada bagian mereka memulai permainan, alih-alih maju dan menempati area papan pemain lain.

perhitungan, nilai heuristik ≤ 0 maka dapat disimpulkan bahwa tidak ada lagi gerakan untuk pemain. Algoritma untuk LeakyAI ditunjukkan di bawah ini.

N.B. nilai awal untuk CurrentHeuristic adalah 0, fungsi FloodFill mengembalikan jumlah persegi yang telah diisi, fungsi LargestMoveAt mengembalikan potongan terbesar yang dapat dimainkan di lokasi tersebut, dan array HeuristicBoard sesuai dengan Rencana Heuristik yang disebutkan di atas.

```

1 TRVERSE Row 1 Until 20:
2 TRVERSE Column 1 Until 20:
3 SET Leaky = false
4 IF Right AND Bottom is NOT vacant and ARE enemies:
5 SET Leaky = true
6 IF Left AND Top is NOT vacant and ARE enemies:
7 SET Leaky = true
8 IF Left AND Bottom is NOT vacant and ARE enemies:
9 SET Leaky = true
10 IF Top AND Right is NOT vacant and ARE enemies:
11 SET Leaky = true
12
13 IF Leaky is true:
14 FloodFillAmount += FloodFill(Row, Column, 0)
15 IF FloodFillAmount > Largest Piece Available:
16 CurrentHeuristic += FloodFillAmount
17 CurrentHeuristic += LargestMoveAt(Row, Column)
18 IF HeuristicBoard[Row][Col] is 10:
19 ACTIVATE Aggressive Mode
20 Place Highest Heuristic Piece at the Location
    
```

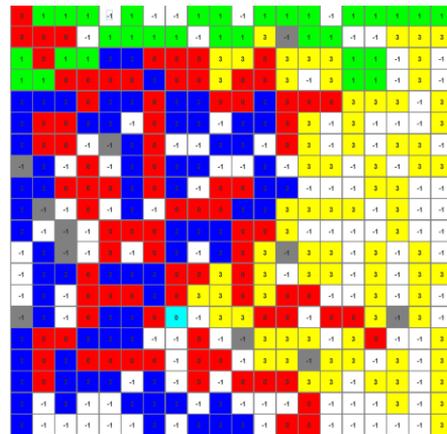
Gambar 12 Algoritma LeakyAI

IMPLEMENTASI DAN HASIL

Pada implementasinya kami telah mencoba menempatkan LeakyAI untuk bersaing dengan 3 Kecerdasan Buatan Brute Force lainnya. Brute Force AI hanya menempatkan bagian terbesar yang tersedia di dek dan memutarkannya untuk menemukan apakah cocok atau tidak, maka ia pergi dengan cara menurun menuju bagian terkecil. Hasilnya cukup konstan dengan LeakyAI memenangkan 100% permainan dengan total 20 poin di setiap pertandingan. LeakyAI ditempatkan sebagai Pemain Merah. Dengan papan skor seperti di bawah ini.

Tabel 4 LeakyAI vs Brute Force AI

Pemain	Poin
Merah	20
Hijau	-58
Biru	-13
Kuning	-10

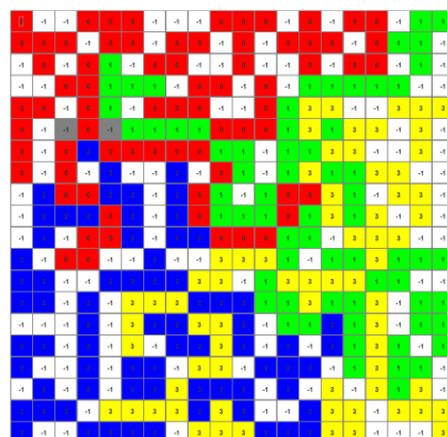


Gambar 13 LeakyAI vs Brute Force AI

Kemudian, kami mencoba untuk membandingkannya terhadap dirinya sendiri. Kami menempatkan 4 LeakyAI pada papan permainan dan tindakan yang dilakukan menunjukkan bahwa tidak ada LeakyAI yang berhasil menempatkan semua potongannya pada papan. Hasilnya ditunjukkan di bawah ini, dengan poin yang diraih tertinggi adalah -17.

Tabel 5 LeakyAI vs LeakyAI

Pemain	Poin
Merah	-17
Hijau	-22
Biru	-17
Kuning	-20



Gambar 14 LeakyAI vs LeakyAI

Dari permainan antara LeakyAI dengan dirinya sendiri dan LeakyAI dan Brute Force AI, tidak ada elemen keacakan dalam Kecerdasan Buatan yang dibangun. Kami menyimpulkan bahwa tidak peduli berapa banyak ronde/permainan yang dilakukan, hasilnya akan selalu menunjukkan apa yang ditampilkan pada Gambar 13 dan Gambar 14. Tapi, beberapa langkah pertama yang dilakukan oleh Kecerdasan Buatan ditempatkan di papan yang berbeda yang mengakibatkan potongan yang berbeda ditempatkan dengan cara yang berbeda.

Akhirnya, kami sendiri juga mencoba bersaing dengan LeakyAI sendiri setiap pengembang memainkan 3 putaran permainan (dengan total 9 putaran) bersaing dengan 3 LeakyAI yang tidak bekerja sama. Hasilnya diperlihatkan dalam gambar berikut.

N.B. "Dev." berarti pengembang dan "R" berarti putaran.

	Merah (Dev.)	Hijau	Biru	Kuning
Poin R1	-17	-27	-27	-34
Poin R2	-27	-15	-21	-9
Poin R3	-8	-23	-6	-21
Poin R4	-5	-27	-34	-20
Poin R5	20	-26	-8	-26
Poin R6	-20	-20	-8	-16
Poin R7	20	-32	-24	-39
Poin R8	-8	-32	-16	-17
Poin R9	-4	-8	-29	-18
Rata-Rata	-5,44	-23,33	-19,22	-22,22
MAX	20	-8	-6	-9
MIN	-27	-32	-34	-39

Gambar 15 Hasil Pengembang vs LeakAI

Berbeda dengan apa yang terjadi ketika papan dipenuhi dengan pemain Kecerdasan Buatan, bermain dengan manusia nyata (pengembang dalam hal ini) membawa gerakan acak dari LeakyAI yang menyesuaikan langkahnya ke langkah pemain lain. Dengan poin yang ditunjukkan dari sisi pengembang, masih mungkin bagi pengembang untuk

mengalahkan LeakyAI dengan rata-rata tingkat kemenangan 66,67% (peringkat satu dari empat) dan dengan hanya rata-rata 22,22% mendapatkan skor penuh (tempatkan semua bagian pemain). Dengan rata-rata -5,44 poin dibandingkan dengan rata-rata LeakyAI terbaik dengan -19,22 poin. Pada keadaan optimalnya pengembang bisa mendapatkan skor penuh dengan 20 poin, sedangkan LeakyAI mendapat maksimal -6 poin. Di sisi lain, pada putaran permainan terburuk, pengembang berhasil mencapai -27 poin dibandingkan dengan LeakyAI terburuk adalah -39 poin. Tidak seperti apa yang dipraktikkan sebelumnya, ketika LeakyAI ditempatkan di papan dengan pemain nyata itu akan secara konsisten memberikan gerakan yang berbeda tergantung pada pemainnya. Kami menganalisis bahwa alur permainan LeakyAI adalah seperti yang telah disebutkan, bahwasannya cenderung agresif dan bergerak keluar dari sudutnya. Kami juga merasa sulit untuk menempatkan beberapa potongan tertentu karena LeakyAI selalu terlihat untuk mengisi celah atau kebocoran, di antara potongan-potongan.

KESIMPULAN

Makalah ini telah mengusulkan model permainan *Blokus Classic* bersama dengan implementasi Kecerdasan Buatan menggunakan bahasa pemrograman Java. Di sini kami telah menerapkan Kecerdasan Buatan untuk *pemain Blokus Classic* menggunakan algoritma Heuristik, FloodFill, dan Greedy secara bersamaan. Rencana Heuristik cukup banyak mempengaruhi seberapa LeakyAI akan memutuskan langkahnya dan mempengaruhi kinerjanya secara keseluruhan. Berdasarkan uji coba 9 permainan dengan para pengembang, masih banyak perbaikan yang bisa dilakukan kepada LeakyAI agar bisa mendapatkan peringkat terbaik. Selain

menggunakan algoritma berdasarkan teori, disarankan juga untuk menerapkan strategi dan trik dalam permainan Blokus. Merupakan suatu ide yang baik untuk memprediksi langkah pemain lain ke langkah-langkah tertentu untuk sistem keputusan yang lebih baik. Kemampuan ini dapat disediakan saat menerapkan *Monte Carlo Tree Search* atau *MiniMax* untuk pengembangan berikutnya.

DAFTAR PUSTAKA

- [1] "Blokus | Board Game | BoardGameGeek." [Online]. Available: <https://boardgamegeek.com/boardgame/2453/blokus>. [Diakses: 01-Apr-2020].
- [2] B. Goertzel, "The Hidden Pattern: A Patternist Philosophy of Mind," p. 470, 2006.
- [3] "Blokus Duo | Board Game | BoardGameGeek." [Online]. Available: <https://boardgamegeek.com/boardgame/16395/blokus-duo>. [Diakses: 02-Apr-2020].
- [4] N. Sugimoto, T. Miyajima, T. Kuhara, Y. Katuta, T. Mitsuichi, and H. Amano, "Artificial intelligence of Blokus Duo on FPGA using Cyber Work Bench," *FPT 2013 - Proc. 2013 Int. Conf. F. Program. Technol.*, pp. 498–501, 2013.
- [5] A. Jahanshahi, M. K. Taram, and N. Eskandari, "Blokus Duo game on FPGA," *Proc. - 17th CSI Int. Symp. Comput. Archit. Digit. Syst. CADs 2013*, pp. 149–152, 2013.

ISSN 2528-5114



Universitas Pelita Harapan Kampus Medan

Lippo Plaza Medan, Lantai 5 - 7,
Jl. Imam Bonjol No.6, Petisah Tengah, Medan Petisah, Medan
City, North Sumatra 20112
Telp. (061) 80511117