

## DAFTAR PUSTAKA

- Aprianti, W., & Maliha, U. (2016). Sistem Informasi Kepadatan Penduduk Kelurahan Atau Desa Studi Kasus Pada Kecamatan Bati-Bati Kabupaten Tanah Laut. *Jurnal Sains Dan Informatika*, 2(1), 21–28.
- Ardiansyah. (2017). *Analisis Perbandingan Kinerja Keuangan Perusahaan Telekomunikasi Yang Terdaftar di BEI (Studi Kasus PT. XL Axiata Tbk dan PT. Indosat Tbk)*. Institut Agama Islam Negeri (IAIN): Padangsidempuan.
- Ashari, M., & Mintarsih, F. (2017). Aplikasi Pemilihan Bibit Budidaya Ikan Air Tawar dengan Metode MOORA – Entropy. *Jurnal Sistem Informasi, Volume: 01*(October), 63–72.
- Basuki, A., & Cahyani, A. D. (2020). *Sistem Pendukung Keputusan*. Yogyakarta: Deepublish.
- Britama.com. (2022, 02 Februari). Sejarah dan Profil Singkat BUKA (Bukalapak.com Tbk). Diakses pada 22 Mei 2022, dari <https://britama.com/index.php/2022/02/sejarah-dan-profil-singkat-buka/>
- Budiman, R. (2020). *Investing is Easy (Edisi Revisi)*. Jakarta: PT. Elex Media Komputindo.
- Darmadji, T., & Hendy M, F. (2012). *Pasar Modal Di Indonesia (Pendekatan Tanya Jawab)*. Jakarta: Salemba Empat.
- Dedi Irawan, M., & Hervina. (2018). Implementasi Logika Fuzzy Dalam Menentukan Jurusan Bagi Siswa Baru Sekolah Menengah Kejuruan (Smk) Negeri 1 Air Putih. *Jurnal Teknologi Informasi*, 2(2), 129–137.
- Elgamar. (2020). *Buku Ajar Konsep Dasar Pemrograman Website Dengan PHP* (N. Pangesti (Ed.)). Malang: CV. Multimedia Edukasi.
- Fahmi. (2012). *Analisis Laporan Keuangan (Edisi Kedu)*. Bandung: Alfabeta.
- Fitriani, P., & Alasi, T. S. (2020). *Sistem Pendukung Keputusan dengan Metode WASPAS, COPRAS dan EDAS: Menentukan Judul Skripsi Mahasiswa*. Yayasan Kita Menulis.
- Guntara, D., Irwan, M., Nasution, P., & Nasution, A. B. N. (2020). Implementasi Metode Economic Order Quantity Pada Aplikasi Pengendalian Bahan Produksi Sandal Mirado. *Jurnal Teknik Informatika*, 13(1), 31–42.
- Handari Bancin, R. (2021). *Pengaruh Piutang Lancar Dan Piutang Tidak Lancar Terhadap Arus Kas Pada Pt. Charoen Pokphand Indonesia Tbk (Cpin)*. Universitas Islam Negeri Sumatera Utara: Medan.

- Hartono, J. (2017). *Teori Portofolio Dan Analisis Investasi* (Edisi Kese). Yogyakarta: BPFE.
- Hendini, A. (2016). Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak). *Jurnal Khatulistiwa Informatika*, 2(9), 107–116.
- Hermawan, K. S., & Hartomo, K. D. (2021). Sistem Pendukung Keputusan Menyeleksi Saham LQ45 untuk Generasi Milenial Menggunakan Metode SAW. *Jurnal Nasional Informatika Dan Teknologi Jaringan*, 2, 259–264.
- Hidayanti, T. (2022). *Analisis Perbandingan Kinerja Portofolio Optimal Saham Syariah Pada Jakarta Islamic Index (Jii) Menggunakan Single Index Model Dan Model Markowitz Pada Saat Pandemi Covid-19 Periode 2019-2020*. Universitas Islam Negeri Raden Intan: Lampung.
- Idxchannel.com. (2022, 28 Juli). Sejarah dan Profil EMTK, Perusahaan Media yang Banyak Diincar Investor. Diakses pada 22 Mei 2022, dari <https://www.idxchannel.com/market-news/sejarah-dan-profil-emtk-perusahaan-media-yang-banyak-diincar-investor>
- Idxchannel.com. (2022, 05 September). Intip Sejarah dan Profil BRPT, Perusahaan Energi Terbarukan yang Jadi Sorotan Investor. Diakses pada 22 Mei 2022, dari <https://www.idxchannel.com/market-news/intip-sejarah-dan-profil-brpt-perusahaan-energi-terbarukan-yang-jadi-sorotan-investor>
- Iqbal. (2019). *5 Jam Belajar PHP MySQL Dengan Dreamweaver CS3*. Yogyakarta: Deepublish.
- Irawan, M. D., & Nasution, M. K. I. (2018). Rancang Bangun Sistem Pakar Mendiagnosa Penyakit Tanaman Kelapa Sawit Menggunakan Metode Bayes Berbasis Android (Studi Kasus : Perkebunan PTPN 4 Air Batu). *Jurnal Teknologi Informasi*, 2(1), 15–23.
- Ishak, I. chaidir, Sinsuw, A., & Tulenan, V. (2017). Sistem Pendukung Keputusan Kelayakan Sertifikasi Guru Menggunakan Metode Simple Additive Weighting (SAW). *Jurnal Teknik Informatika*, 10(1), 1–10.
- Issn, P., & Israwan, F. (2019). Penerapan Multi-Objective Optimization On The Basis Of Ratio ( Moora ) Dalam Penentuan Asisten Laboratorium. *Jurnal Ilmiah Ilmu Komputer*, 5(1), 1–5.
- Kariyoto. (2017). *Analisa Laporan Keuangan*. Malang: UB Media.
- Kurniawan, R., & CSA, L. S. (2021). *Fundamental VS Technical*. Jakarta: PT. Gramedia.
- Kusuma, A. S., & Aryawan, I. M. G. (2019). Sistem Pendukung Keputusan

- Pemilihan Saham BUMN dengan Model AHP. *Jurnal Sistem Informasi Dan Komputer Terapan Indonesia (JSIKTI)*, 1(4), 225–234.
- Magdalena, H. (2017). Strategi Meningkatkan Kualitas Bimbingan Skripsi Mahasiswa Strata Satu Stmik Atma Luhur. *Jurnal Sisfokom*, 2(1), 1–9.
- Maryam. (2021). *Pengaruh Penghindaran Pajak Terhadap Nilai Perusahaan Dengan Transparansi Sebagai Variabel Pemoderasi ( Studi pada Perusahaan yang terdaftar di JII-BEI Tahun 2015-2019)*. Universitas Islam Negeri Sulthan Thaha Saifuddin: Jambi.
- MF, M. (2018). *Bukti Sakti Pemrograman Web Seri PHP*. Anak Hebat Indonesia.
- Muawanah, S., Ekonomi, F., & Riau, U. I. (2019). *Pengaruh Pertumbuhan Aset, Pertumbuhan Penjualan, Dan Profitabilitas Terhadap Keputusan Pendanaan Pada Perusahaan Retail Yang Terdaftar Dibursa Efek Indonesia*. Universitas Islam Riau: Pekanbaru.
- Najmiah, Edy, S., & Ni, K. S. (2014). Pengaruh Price to Book Value (PBV), Price Earning Ratio (PER), dan Debt to Equity Ratio (DER) Terhadap Return Saham Pada Industri Real Estate dan Property yang Terdaftar di Bursa Efek Indonesia Periode 2009-2013. *E-Journal SI AK Universitas Ganesha*.
- Nofriansyah, D., & Defit, S. (2020). *Multi Criteria Decision Making (MCDM) Pada Sistem Pendukung Keputusan*. Yogyakarta: Deepublish.
- Rahardjo, B. (2009). *Jeli Investasi Saham Ala Warren Buffett*. Yogyakarta: CV. ANDI.
- Rani, F. P., Khairina, D. M., Hatta, H. R., Samarinda, U. M., & Prestasi, I. (2019). Sistem Pendukung Keputusan Pemilihan Pramuka Pandega Berprestasi Menggunakan Metode Multi Objective Optimization On The Basis Of Ratio Analysis. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 6(2), 155–162.
- Rerung, R. R. (2018). *Pemrograman Web Dasar*. Yogyakarta: Deepublish.
- Rosa, A. S., & Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Samsudin. (2018). Penentuan Penerimaan Remunerasi Dosen Dengan Rule Based Reasoning. *Journal of Materials Processing Technology*, 1 (1), 1–8.
- Sanjaya, R. (2020). Sistem Pengambilan Keputusan Untuk Menentukan Perumahan Terbaik Berdasarkan Kondisi dan Lokasi Menggunakan Metode ENTROPY dan ARAS. *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)*, 447–452.
- Sari, F. (2018). *Metode dalam Pengambilan Keputusan*. Yogyakarta: Deepublish.

- Sejarah dan Profil Singkat BRPT (Barito Pacific Tbk)*. (2012). Britama.Com. <https://britama.com/index.php/2012/10/sejarah-dan-profil-singkat-brpt/>
- Sinaga, G. R. U., & Samsudin. (2021). Implementasi Framework Laravel dalam Sistem Reservasi pada Restoran Cindelaras Kota Medan. *Jurnal Janitra Informatika Dan Sistem Informasi*, 1(2), 73–84.
- Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer Dan Informatika*, 3(1), 1–9.
- Supono, & Putratama, V. (2018). *Pemrograman Web dengan Menggunakan PHP dan Framework Codeigniter*. Yogyakarta: CV. Budi Utama.
- Sya'ban, M. R. (2021). *Pengaruh E-Service Quality, Brand Trust, Dan Digital Marketing Terhadap Keputusan Nasabah Dalam Memilih Indopremier Sebagai Broker Saham Dan Dampak Terhadap Kepuasan Nasabah*. Universitas Islam Negeri Syarif Hidayatullah: Jakarta.
- Syafitri, A. N. (2020). *Analisis Faktor Penunjang Profitabilitas Bumn Yang Terdaftar Di Bursa Efek Indonesia: Perspektif Internal Dan Makro*. Universitas Islam Negeri Syarif Hidayatullah: Jakarta.
- Tanjung, S. R., Mesran, M., Sarwandi, S., & Siagian, M. V. (2021). Penerapan Metode COPRAS dan ENTROPY dalam Pemilihan Anggota Badan Pengawas Pemilihan Umum (BAWASLU). *Journal of Informatics Management and Information Technology*, 1(2), 48–59.
- Tohari, H. (2018). *ASTAH Analisis serta Perancangan Sistem Informasi melalui pendekatan UML*. Yogyakarta: ANDI.
- Waruwu, T. S., & Nasution, S. (2020). Sistem Pendukung Keputusan Pemilihan Investasi Saham Berbasis Web Menggunakan Metode SMART. *Jurnal Mahajana Informasi*, 5(1), 8–13.
- Widianto, T., & Khristiana, Y. (2021). Analisis Under Pricing Saham Pada Penawaran Pasar Perdana Di Bursa Efek Indonesia Sebelum Pandemi Covid-19 Di Dunia. *Jurnal Manajemen, Bisnis Dan Pendidikan*, 8(1), 79–91.
- Yendrizal. (2020). Penentuan Siswa SMK Kimia Analisa Terbaik Yang Akan Dikirim Mengikuti Olimpiade Kimia Tingkat Nasional Menerapkan Metode Entropy dan MOORA. *Jurnal Media Informatika Budidarma*, 4(4), 963–969.
- Zufria, I., Irawan, M. D., Suendri, & Muhyi, H. A. (2020). Aplikasi Tracking Real Time Angkutan Kota Medan Berbasis Android. *JISTech (Journal of Islamic Science and Technology)*, 5(2), 63–74.

Zulfikar. (2016). *Pengantar Pasar Modal dengan Pendekatan Statistika*. Yogyakarta: Deepublish.



UNIVERSITAS ISLAM NEGERI  
SUMATERA UTARA MEDAN

## LAMPIRAN

### Lampiran I

### Surat Izin Riset UINSU Medan – IndoPremier Sekuritas Medan

Firefox

<https://siselma.uinsu.ac.id/pengajuan/cetakaktif/NTU3MTQ=>



KEMENTERIAN AGAMA REPUBLIK INDONESIA  
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA MEDAN  
FAKULTAS SAINS DAN TEKNOLOGI  
Jl. Willem Iskandar Pasar V Medan Estate 20371  
Telp. (061) 6615683-6622925 Fax. 6615683

Nomor : B.1138/ST./ST.V.2/TL.00/12/2021

13 Desember 2021

Lampiran : -

Hal : Izin Riset

Yth. Bapak/Ibu Kepala Indo Premier Sekuritas Medan

*Assalamualaikum Wr. Wb.*

Dengan Hormat, diberitahukan bahwa untuk mencapai gelar Sarjana Strata Satu (S1) bagi Mahasiswa Fakultas Sains dan Teknologi adalah menyusun Skripsi (Karya Ilmiah), kami tugaskan mahasiswa:

Nama : Merliana Putri Hasibuan  
NIM : 0702173189  
Tempat/Tanggal Lahir : Pasir Pengaraian, 22 Maret 1999  
Program Studi : Sistem Informasi  
Semester : XI (Sebelas)  
Alamat : JLN. DIPONEGORO NO. 177, PASIR PENGARAIAN, KAB. ROKAN HULU, RIAU Kelurahan PASIR PENGARAIAN Kecamatan RAMBAH

untuk hal dimaksud kami mohon memberikan Izin dan bantuannya terhadap pelaksanaan Riset di Jl. Iskandar Muda No 47 C Kel. Babura, Medan Baru, Kota Medan, Sumatera Utara 20153, guna memperoleh informasi/keterangan dan data-data yang berhubungan dengan Skripsi (Karya Ilmiah) yang berjudul:

***Sistem Pengambilan Keputusan Pemilihan Investasi Saham LQ45 Menggunakan Metode Entropy dan Moore***

Demikian kami sampaikan, atas bantuan dan kerjasamanya diucapkan terima kasih.

Medan, 13 Desember 2021  
a.n. DEKAN  
Wakil Dekan Bidang Akademik dan  
Kelembagaan



*Digitaly Signed*

**Dr. Abdul Halim Dawlay, ST., M.Si**  
NIP. 198111062005011003

Tembusan:

- Dekan Fakultas Sains dan Teknologi UIN Sumatera Utara Medan

Agar : Pastikan scan QR Code diatas dan UIN link yang muncul, untuk mengetahui keabsahan surat

## Lampiran II

### Surat Balasan Izin Riset IndoPremier Sekuritas Medan

**INDOPREMIER**

Hal : **Balasan**  
 Kepada Yth :  
**Dekan Fakultas Sains dan Teknologi**  
**UIN Sumatera Utara**  
 Di- Tempat

Dengan Hormat,

Sesuai surat yang kami terima sebelumnya dengan nomor B.1138/ST.I/ST.V.2/TL.00/12/2021, maka dengan ini saya yang bertanda tangan dibawah ini:

Nama : Ade Rakhmat Suryanto  
 Jabatan : West Regional Manager  
 Perusahaan : PT Indo Premier Sekuritas

Menerangkan bahwa,

Nama : Merliana Putri Hasibuan  
 NIM : 0702173189  
 Jurusan : Sistem Informasi

Telah kami setuju untuk melaksanakan penelitian pada perusahaan kami sebagai syarat penyusunan skripsi dengan judul:

***Sistem Pengambilan Keputusan Pemilihan Investasi Saham LQ45  
 Menggunakan Metode Entropy dan Moora.***

Demikian surat ini kami sampaikan, dan atas kerjasamanya kami ucapkan terimakasih.

Medan, 17 Desember 2021  
 Hormat Kami  
 West Regional Manager  
 PT Indo Premier Sekuritas

  
**Ade Rakhmat Suryanto**

**PT INDO PREMIER SECURITIES**

Wisma GKBH 7/F Suite 718  
 Jl. Jend. Sudirman No.28, Jakarta 10210 - Indonesia  
 p +62.21.5793.1168 f +62.21.5793.1167

## Lampiran III

## Hasil Wawancara Kriteria dan Sub Kriteria

## WAWANCARA

## PENENTUAN KRITERIA DAN SUB KRITERIA

NO.	PERTANYAAN	PERNYATAAN
1.	Terkait dengan sistem pendukung keputusan yang akan dirancang dan dibangun, kriteria apa saja yang dapat dipakai untuk menentukan pemilihan saham LQ45?	<ul style="list-style-type: none"> <li>- Net profit Margin (NPM)</li> <li>- Earning per share (Eps)</li> <li>- Price to Earning Ratio (PER)</li> <li>- price to Book Value (PBV)</li> <li>- Debt to Equity Ratio (DER)</li> <li>- Return On Asset (ROA)</li> <li>- Return On Equity (ROE)</li> </ul>
2.	Untuk bobot (W) setiap kriterinya berapa yang digunakan?	<ul style="list-style-type: none"> <li>- NPM 0,30</li> <li>- Eps 0,20</li> <li>- PER 0,12</li> <li>- PBV 0,15</li> <li>- DER 0,10</li> <li>- ROA 0,08</li> <li>- ROE 0,05</li> </ul>
3.	Dari 7 kriteria yang sudah ditetapkan, bagaimana penjabaran dari sub kriterianya?	



Catatan :

NPM	$L=1$	1	PER	$L=0.5$	1	PMU	$L=1$	1
	$1.0 - L=10$	2		$0.50 - L=1$	2		$1.0 - L=2$	2
	$10.0 - L=20$	3		$1.0 - L=2$	3		$2.0 - L=3$	3
	$20.0 - L=30$	4		$2.0 - L=3$	4		$3.0 - L=4$	4
	$>30.0$	5		$L=3$	5		$74.0$	5

SPS	$L=100$	1	PER	$L=1$	1
	$100.0 - L=300$	2		$1.0 - L=10$	2
	$300.0 - L=600$	3		$10.0 - L=20$	3
	$600.0 - L=900$	4		$30.0 - L=30$	4
	$>900.0$	5		$>30.0$	5

PER	$L=10$	1	PER	$L=1$	1
	$10.0 - L=20$	2		$1.0 - L=10$	2
	$30.0 - L=30$	3		$10.0 - L=20$	3
	$30.0 - L=40$	4		$20.0 - L=30$	4
	$>40.0$	5		$>20.0$	5

Medan, 20 APRIL 2017

Agus Priyanto  
AGUS PRIYANTO

## Daftar Kriteria dan Sub Kriteria

No.	Kriteria	Sub Kriteria	Keterangan	Nilai
1.	NPM	$\leq 1$	Sangat Tidak Baik	1
		1.0 - $\leq 10$	Tidak Baik	2
		10.0 - $\leq 20$	Cukup Baik	3
		20.0 - $\leq 30$	Baik	4
		$>30.0$	Sangat Baik	5
2.	EPS	$\leq 100$	Sangat Tidak Baik	1
		100.0 - $\leq 300$	Tidak Baik	2
		300.0 - $\leq 600$	Cukup Baik	3
		600.0 - $\leq 900$	Baik	4
		$>900.0$	Sangat Baik	5
3.	PER	$\leq 10$	Sangat Baik	1
		10.0 - $\leq 20$	Baik	2
		20.0 - $\leq 30$	Cukup Baik	3
		30.0 - $\leq 40$	Tidak Baik	4
		$>40.0$	Sangat Tidak Baik	5
4.	PBV	$\leq 1$	Sangat Baik	1
		1.0 - $\leq 2$	Baik	2
		2.0 - $\leq 3$	Cukup Baik	3
		3.0 - $\leq 4$	Tidak Baik	4
		$>4.0$	Sangat Tidak Baik	5
5.	DER	$\leq 0.5$	Sangat Baik	1
		0.50 - $\leq 1$	Baik	2
		1.0 - $\leq 2$	Cukup Baik	3
		2.0 - $\leq 3$	Tidak Baik	4
		$>3$	Sangat Tidak Baik	5

No.	Kriteria	Sub Kriteria	Keterangan	Nilai
6.	ROA	$\leq 1$	Sangat Tidak Baik	1
		1.0 - $\leq 10$	Tidak Baik	2
		10.0 - $\leq 20$	Cukup Baik	3
		20.0 - $\leq 30$	Baik	4
		$>30.0$	Sangat Baik	5
7.	ROE	$\leq 1$	Sangat Tidak Baik	1
		1.0 - $\leq 10$	Tidak Baik	2
		10.0 - $\leq 20$	Cukup Baik	3
		20.0 - $\leq 30$	Baik	4
		$>30.0$	Sangat Baik	5

Medan, 20 April 2022


  
 AGUS PRIYONO

## Lampiran IV

## Data Laporan Keuangan Tahun 2021

No	Nama Perusahaan	Harga Saham	Net Profit	Listed Share	Total Ekuitas	Total Aset	Total Liabilitas	Revenue
1.	ADRO	2.250	14.676.993.517.000	31.985.962.000	63.615.696.735.000	108.257.989.784.000	44.642.293.049.000	56.972.093.142.000
2.	AMRT	1.215	1.988.750.000.000	41.524.501.700	8.989.798.000.000	27.493.748.000.000	18.503.950.000.000	84.904.301.000.000
3.	ANTM	2.250	1.710.459.702.000	24.030.764.725	20.343.881.627.000	33.300.839.872.000	12.956.958.245.000	26.476.256.076.000
4.	ASII	5.700	25.586.000.000.000	40.483.553.140	215.615.000.000.000	367.311.000.000.000	151.696.000.000.000	233.485.000.000.000
5.	BRPT	855	4.223.723.883.000	93.747.218.044	60.886.893.175.000	131.867.691.219.000	70.980.798.044.000	45.028.055.464.000
6.	BUKA	430	1.675.743.735.000	103.062.019.354	23.495.618.749.000	26.615.549.957.000	3.119.931.208.000	1.869.122.325.000
7.	CPIN	5.950	3.619.010.000.000	16.398.000.000	25.149.999.000.000	35.446.051.000.000	10.296.052.000.000	51.698.249.000.000
8.	EMTK	2.280	6.019.825.801.000	61.241.751.483	33.668.866.284.000	38.168.511.114.000	4.499.644.830.000	12.840.734.345.000
9.	ERAA	600	1.117.917.248.000	15.950.000.000	6.462.361.670.000	11.372.225.256.000	4.909.863.586.000	43.466.976.696.000
10.	EXCL	3.170	1.016.399.000.000	10.724.674.776	19.821.710.000.000	68.585.018.000.000	48.763.308.000.000	19.800.193.000.000
11.	GGRM	30.600	4.134.576.000.000	1.924.088.000	57.720.347.000.000	83.251.441.000.000	25.531.094.000.000	92.070.856.000.000
12.	HMSP	965	5.554.491.000.000	116.318.076.900	27.399.042.000.000	48.852.802.000.000	21.453.760.000.000	72.519.260.000.000
13.	HRUM	10.325	1.402.451.295.634	2.703.620.000	9.284.420.974.557	12.479.975.596.131	3.195.554.621.574	4.796.888.452.073
14.	ICBP	8.700	6.081.778.000.000	11.661.908.000	52.847.054.000.000	107.306.713.000.000	54.459.659.000.000	42.622.053.000.000

15.	INCO	4.680	2.366.08 8.987.00 0	9.936.3 38.720	30.746.3 12.931.0 00	35.289. 728.38 8.000	4.543.415. 457.000	13.602.74 6.154.000
16.	INDF	6.325	8.001.38 4.000.00 0	8.780.4 26.500	82.960.6 79.000.0 00	172.12 7.169.0 00.000	89.166.49 0.000.000	72.808.32 0.000.000
17.	INKP	7.825	7.520.31 9.491.00 0	5.470.9 82.941	67.902.0 32.990.0 00	128.11 3.431.7 05.000	60.211.39 8.715.000	50.178.16 5.634.000
18.	INTP	12.100	1.208.25 8.000.00 0	3.681.2 31.699	21.549.6 54.000.0 00	26.256. 521.00 0.000	4.706.867. 000.000	10.608.65 3.000.000
19.	ITMG	20.400	6.836.58 3.590.00 0	1.129.9 25.000	17.279.6 19.979.0 00	23.962. 183.05 9.000	6.682.563. 080.000	29.866.64 7.753.000
20.	JPFA	1.720	2.130.89 6.000.00 0	11.726. 575.20 1	13.102.7 10.000.0 00	28.589. 656.00 0.000	15.486.94 6.000.000	44.878.30 0.000.000
21.	KLBF	1.615	2.324.31 0.806.85 3	46.875. 122.11 0	19.991.7 41.323.8 29	24.266. 776.39 0.675	4.275.035. 066.846	19.098.69 5.082.934
22.	MDKA	3.890	476.396. 249.200	22.904. 850.81 5	11.121.4 05.527.8 93	18.244. 238.65 1.271	7.122.833. 123.378	5.435.868. 997.681
23.	MEDC	466	893.250. 116.019	25.136. 231.25 2	17.541.4 01.962.0 09	81.103. 342.77 9.391	63.561.94 0.817.382	18.881.33 4.162.096
24.	MIKA	2.260	1.361.52 3.557.33 3	14.246. 349.50 0	5.925.14 3.836.67 1	6.860.9 71.097. 854	935.827.2 61.183	4.352.868. 253.731
25.	MNCN	900	1.810.41 7.000.00 0	15.049. 787.71 0	16.760.8 48.000.0 00	20.137. 879.00 0.000	3.377.031. 000.000	7.072.764. 000.000
26.	PGAS	1.375	5.201.53 7.572.31 5	24.241. 508.19 6	46.872.5 88.518.1 02	107.17 3.729.8 82.000	60.301.14 1.364.536	43.322.12 4.541.164
27.	PTBA	2.710	8.036.88 8.000.00 0	11.520. 659.25 0	24.253.7 24.000.0 00	36.123. 703.00 0.000	11.869.97 9.000.000	29.261.46 8.000.000
28.	PTPP	990	361.421. 984.159	6.199.8 97.354	14.330.1 49.681.0 57	55.573. 843.73 5.084	41.243.69 4.054.027	16.763.93 6.677.996
29.	SMGR	7.250	2.082.34 7.000.00 0	5.931.5 20.000	39.782.8 83.000.0 00	76.504. 240.00 0.000	36.721.35 7.000.000	34.957.87 1.000.000

30.	TBIG	2.950	1.601.35 3.000.00 0	22.656. 999.44 5	9.789.23 8.000.00 0	41.870. 435.00 0.000	32.081.19 7.000.000	6.179.584. 000.000
31.	TINS	1.455	1.302.84 3.000.00 0	7.447.7 53.454	6.308.42 0.000.00 0	14.690. 989.00 0.000	8.382.569. 000.000	14.607.00 3.000.000
32.	TKIM	7.525	3.553.06 6.614.00 0	3.113.2 23.570	25.055.4 79.322.0 00	45.116. 209.34 6.000	20.060.73 0.024.000	14.618.00 5.471.000
33.	TLKM	4.040	25.663.0 00.000.0 00	99.062. 216.00 0	118.813. 000.000. 000	246.50 0.000.0 00.000	127.687.0 00.000.00 0	106.043.0 00.000.00 0
34.	TOWR	1.125	3.447.87 5.000.00 0	51.014. 625.00 0	12.062.0 16.000.0 00	65.828. 670.00 0.000	53.766.65 4.000.000	8.635.346. 000.000
35.	TPIA	7.325	2.168.94 5.076.00 0	21.627. 886.27 3	41.774.8 51.885.0 00	71.245. 973.14 0.000	29.471.12 1.255.000	36.820.08 4.325.000
36.	UNTR	22.150	10.608.2 67.000.0 00	3.730.1 35.136	71.822.7 57.000.0 00	112.56 1.356.0 00.000	40.738.59 9.000.000	79.460.50 3.000.000
37.	UNVR	4.110	5.758.14 8.000.00 0	38.150. 000.00 0	4.321.26 9.000.00 0	19.068. 532.00 0.000	14.747.26 3.000.000	39.545.95 9.000.000
38.	WIKA	1.105	214.424. 794.000	8.969.9 51.372	17.435.0 77.712.0 00	69.385. 794.34 6.000	51.950.71 6.634.000	17.809.71 7.726.000
39.	WSKT	635	1.838.73 3.441.97 5	13.573. 951.00 0	15.461.4 33.243.8 30	103.60 1.611.8 83.340	88.140.17 8.639.510	12.224.12 8.315.553

Lampiran V  
Formulir *Black-box Testing*

## LAMPIRAN

FORMULIR PENGUJIAN *BLACK BOX*

Tanggal Pengujian : 25 Juni 2022

Nama Aplikasi : Sistem Pendukung Keputusan Pemilihan Investasi Saham

LQ45 Menggunakan Metode Entropy dan MOORA

No.	Deskripsi	Prosedur Pengujian	Hal yang Diharapkan	Hasil Akurasi	
				Berhasil	Gagal
1.	Pengujian Form Login	Mengisis <i>username</i> dan <i>password</i> yang salah	Muncul pesan "Login Gagal, Periksa Kembali Username dan Password"	✓	
		Mengosongkan <i>username</i> dan <i>password</i>	Muncul pesan "This is a required field"	✓	
		Mengisi <i>username</i> dan <i>password</i> yang sesuai untuk login	Sistem akan menampilkan menu <i>dashboard</i>	✓	
2.	Pengujian Form Register	Mengosongkan nama pengguna, <i>username</i> dan <i>password</i>	Muncul pesan "This is a required field"	✓	
		Mengisi nama pengguna, <i>username</i> dan <i>password</i>	Sistem akan menampilkan menu halaman <i>login</i>	✓	
3.	Pengujian Form Alternatif	Memilih menu alternatif	Sistem akan menampilkan menu alternatif	✓	
		Memilih button "Tambah Data"	Sistem akan menampilkan menu tambah data	✓	

		Memilih button "Edit" pada salah satu data alternatif	Sistem akan menampilkan menu edit data	✓	
		Memilih button "Hapus" pada salah satu data alternatif	Sistem akan menghapus salah satu data alternatif yang dipilih	✓	
4.	Pengujian Form Kriteria	Memilih menu kriteria	Sistem akan menampilkan menu kriteria	✓	
		Memilih button "Edit" pada salah satu data kriteria	Sistem akan menampilkan menu edit data	✓	
5.	Pengujian Form Sub Kriteria	Memilih menu sub kriteria	Sistem akan menampilkan menu sub kriteria	✓	
		Memilih button "Tambah Data"	Sistem akan menampilkan menu tambah data	✓	
		Memilih button "Edit" pada salah satu data sub kriteria	Sistem akan menampilkan menu edit data	✓	
		Memilih button "Hapus" pada salah satu data sub kriteria	Sistem akan menghapus salah satu data sub kriteria yang dipilih	✓	
6.	Pengujian Form Penilaian	Memilih button "Edit"	Sistem akan menampilkan form edit data	✓	
		Memilih button "Simpan" pada form edit penilaian	Sistem akan menyimpan perubahandata yang telah diubahdan kembali ke form data penilaian	✓	

7.	Pengujian <i>Form</i> Perhitungan	Memilih menu perhitungan	Sistem akan menampilkan <i>form</i> data perhitungan dengan data yang sudah diolah dengan metode Entropy dan Moora	✓	
8.	Pengujian <i>Form</i> Hasil	Memilih menu hasil	Sistem akan menampilkan <i>form</i> data hasil dengan data yang sudah di rangking	✓	
		Memilih button "Cetak"	Sistem dapat mencetak data yang sudah di rangking dalam bentuk PDF	✓	
9.	Pengujian <i>Form</i> User	Memilih menu <i>user</i>	Sistem akan menampilkan menu <i>user</i>	✓	
		Memilih button "Tambah Data"	Sistem akan menampilkan menu tambah data	✓	
		Memilih button "Edit" pada salah satu data <i>user</i>	Sistem akan menampilkan menu edit data	✓	
		Memilih button "Hapus" pada salah satu data <i>user</i>	Sistem akan menghapus salah satu data <i>user</i> yang dipilih	✓	

Medan, 29 Juni 2021

  
 AGUS PRIONO



## Lampiran VI

### Source Code

#### 1. Alternatif.php

```

<?php
namespace App;
use PDO;
include "Koneksi.php";
class Alternatif extends Koneksi{
    public function __construct()
    {
        $call = new Koneksi();
        $this->db = $call->koneksi();
    }
    public function tampil()
    {
        $sql = "SELECT * FROM
alternatif ";
        $stmt = $this->db->prepare($sql);
        $stmt->execute();
        $data = [];
        while ($rows = $stmt->fetch()) {
            $data[] = $rows;
        }
        return $data;
    }
    public function hapus($val)
    {
        $sql = "SELECT * FROM
alternatif WHERE id = :id";
        $stmt = $this->db->prepare($sql);
        $stmt->bindParam(":id", $val);
        $stmt->execute();

        $sql = "DELETE FROM
alternatif WHERE id = :id";
        $stmt = $this->db->
>prepare($sql);
        $stmt->bindParam(":id", $val);
        $stmt->execute();

        $sql = "SELECT * FROM
alternatif WHERE id = :id";
        $stmt = $this->db->
>prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        $sql = "UPDATE alternatif
SET nama = :nama, perusahaan =
:perusahaan WHERE id = :id";
        $stmt = $this->db->
>prepare($sql);
        $stmt -> bindParam(":id",
$_GET['id']);
        $stmt ->
bindParam(":nama",
$_POST['nama']);
        $stmt ->
bindParam(":perusahaan",
$_POST['perusahaan']);
        $stmt->execute();
    }
}

```

```

}
public function tambah()
{
    $sql = "INSERT INTO
alternatif VALUES (:, :nama,
:perusahaan, :c1, :c2, :c3, :c4, :c5)";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":nama",
$_POST['kode']);
    $stmt-
>bindParam(":keterangan",
$_POST['keterangan']);
    $stmt->bindParam(":bobot",
$_POST['bobot']);
    $stmt->bindParam(":jenis",
$_POST['jenis']);
    $stmt->execute();
    header("location: kriteria.php");
}
public function simpan()
{
    $sql = "INSERT INTO
alternatif VALUES (:, :nama,
:perusahaan, " , " , " , " , " , " , " );";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":nama",
$_POST['nama']);
    $stmt-
>bindParam(":perusahaan",
$_POST['perusahaan']);
    $stmt->execute();
}

public function get_json($val)
{
    $sql = "SELECT * FROM
alternatif WHERE id=:id limit 1";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":id", $val);
    $stmt->execute();
    //$data['data'] = array();
    $data = $stmt-
>fetchAll(PDO::FETCH_ASSOC);
    return json_encode($data);
}

public function count() //rumus
menghitung jumlah alternatif; yg
ditampilkan di dashboard
{
    $sql = "SELECT COUNT(id)
AS count FROM alternatif";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}

```

```

}
?>
}

2. Kriteria.php
<?php
namespace App;
use PDO;
include "Alternatif.php";
class Kriteria extends Koneksi{
    public function __construct()
    {
        $call = new Koneksi();
        $this->db = $call->koneksi();
    }
    public function count()
    {
        $sql = "SELECT
COUNT(id_kriteria) AS count
FROM kriteria";
        $stmt = $this->db-
>prepare($sql);
        $stmt->execute();
        $data = [];
        while ($rows = $stmt->fetch())
        {
            $data[] = $rows;
        }
        return $data;
    }
    public function tampil()
    {
        $sql = "SELECT * FROM kriteria
ORDER BY kriteria ASC";
        $stmt = $this->db->prepare($sql);
        $stmt->execute();
        $data = [];
        while ($rows = $stmt->fetch()) {
            $data[] = $rows;
        }
        return $data;
    }
    public function hapus()
    {
        $sql = "SELECT * FROM kriteria
WHERE id_kriteria = :id";
        $stmt = $this->db->prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        $sql = "DELETE FROM
kriteria WHERE id_kriteria = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        header("location: kriteria.php");
    }
    public function update()
    {
        $sql = "SELECT * FROM
kriteria WHERE id_kriteria = :id";
        $stmt = $this->db-
>prepare($sql);

```

```

        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        $sql = "UPDATE kriteria
SET kriteria = :kode, jenis = :jenis ,
keterangan = :keterangan,
bobot_awal = :bobot_awal WHERE
id_kriteria = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":kode",
$_POST['kode']);
        $stmt-
>bindParam(":keterangan",
$_POST['keterangan']);
        $stmt->bindParam(":bobot",
$_POST['bobot']);
        $stmt->bindParam(":jenis",
$_POST['jenis']);
        $stmt->execute();
        header("location: kriteria.php");
    }
    public function simpan()
    {
        $sql = "INSERT INTO kriteria
VALUES (:kriteria, :keterangan,
:bobot_awal, 0)";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":kriteria",
$_POST['kriteria']);
        $stmt-
>bindParam(":keterangan",
$_POST['keterangan']);
        $stmt-
>bindParam(":bobot_awal",
$_POST['bobot_awal']);
        $stmt->execute();
    }
    public function tambah()
    {
        $sql = "INSERT INTO kriteria
VALUES (:kode, :keterangan,
:bobot, :jenis)";

```

```

public function get_json($val)
{
    $sql = "SELECT * FROM
kriteria WHERE id_kriteria=:id limit
1";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":id", $val);
    $stmt->execute();
    //$data['data'] = array();
    $data = $stmt-
>fetchAll(PDO::FETCH_ASSOC);
    return json_encode($data);
}

public function count_max()
{
    $sql = "SELECT MAX(id) AS
count FROM alternatif";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}

public function spk($b)
{
    //Initalize
    $sql = "SELECT * FROM
alternatif WHERE id = :id";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":id", $b);
    $stmt->execute();
    //Fetch
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}

public function bobot()
{
    $data = array(array());
    $rik = array(
        '1' => 5,
        '2' => 5,
        '3' => 1,
        '4' => 1,
        '5' => 1,
        '6' => 5,
        '7' => 5
    );
    $bobot = new Kriteria();
    $hasil = new Hasil();
    $a = $bobot->count_max();
    $no1 = 1;
    foreach($a as $row) {
        $b = $row['count'];
    }
    $arraykey = 0;
}

```

```

$roww=$hasil->hitung0();
foreach($roww as $rows) {
    //Selection
    for($i=1;$i<=7;$i++)
    {
        if($i>2 && $i <6)
        {
            $data[$arraykey][$i] =
round(( $rik[$i] / $rows['c'.$i]),4);
        }else{
            $data[$arraykey][$i] =
round(($rows['c'.$i] / $rik[$i]),4);
        }
    }
    $data[$arraykey][0] =
$rows['nama'];
    $arraykey++;
}
return $data;
}
//Total Normalisasi Matriks
public function bobot2($id = 0)
{
    $bobot = new Kriteria();
    $linear = $bobot->bobot();
    $totalc = array();
    $arraykey = 0;
    for($i=1;$i <=7; $i++)
    {
        ${'totalc'.'.i'} = 0;
    }
    foreach ($linear as $p)
    {
        for($i=1;$i <=7; $i++)
        {
            ${'totalc'.'.i'} =
            $no1= 1;
            $data = array(array());
            foreach ($linear as $p )
            {
                for($i=1;$i <=7; $i++)
                {
                    $data[$arraykey][$i] =
                    round( ( $p[$i] / ${'totalc'.'.i'}) *
                    log($p[$i] / ${'totalc'.'.i}), 4);
                }
                $data[$arraykey][0] = $p[0];
                $arraykey++;
            }
        }
    }
    //Redclare
    $totallog = array();
    $counter = 0;
    for($i=1;$i <=7; $i++)
    {
        ${'totallog'.'.i'} = 0;
    }
    foreach( $data as $q)
    {
        for($i=1;$i <=7; $i++)
        {

```

```

        ${'totallog'.$i} =
    ${'totallog'.$i} + $q[$i];
    }
    $counter++;
}
//Input to array
for($i=1;$i <=7; $i++)
{
    $totallog[$i] =
    ${'totallog'.$i};
}
//Entropy
$entropy = array();
$totalentropy = 0;
for($i=1;$i <=7; $i++)
{
    $entropy[$i] = round((-1 /
log($counter))*$totallog[$i],4);
    $totalentropy =
    $totalentropy + $entropy[$i];
}
//Bobot
$bobot = array();
$totalbobot = 0;
for($i=1;$i <=7; $i++)
{
    $bobot[$i] = round( (1 / (7-
$totalentropy))*(1-$entropy[$i]),4);
    $totalbobot = $totalbobot +
    $bobot[$i];
}
//Bobot yand dibulatkan
        $roundbobot = array();
        for($i=1;$i <=7; $i++)
        {
            $roundbobot[$i] =
            round($bobot[$i],4);
        }
        if($id == 0 )
        {
            return $roundbobot;
        }elseif($id == 1)
        {
            return $data;
        }elseif($id == 2)
        {
            return $entropy;
        }elseif($id == 3)
        {
            return $bobot;
        }elseif($id == 4)
        {
            return $totallog;
        }elseif($id == 5)
        {
            return $totalc;
        }
    }
}
public function preparation()
{
    $prep = new Kriteria();
    $toDB = $prep->bobot2();
}

```

```

$counter = 1;
foreach($toDB as $a)
{
    //Execute
    $sql = "SELECT * FROM
kriteria WHERE id_kriteria = :id";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":id",
$counter);
    $stmt->execute();
    $sql = "UPDATE kriteria
SET bobot = :bobot WHERE
id_kriteria = :id";
    $stmt = $this->db-
>prepare($sql);
    $stmt -> bindParam(":id",
$counter);
    $stmt -> bindParam(":bobot",
$a);
    $stmt->execute();
    $counter++;
}
}
}
?>

```

```

public function __construct()
{
    $call = new Koneksi();
    $this->db = $call->koneksi();
}
public function count()
{
    $sql = "SELECT
COUNT(id_sub) AS count FROM
sub_kriteria";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}
public function kriteria()
{
    $sql = "SELECT * FROM
kriteria";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
}

```

### 3. SubKriteria.php

```

<?php
namespace App;
use PDO;
class Subkriteria extends Koneksi{

```

```

{
    $data[] = $rows;
}
}

```



```

        return $data;
    }
    public function tampil($a)
    {
        $sql = "SELECT * FROM
sub_kriteria WHERE id_kriteria =
:id ORDER BY nilai ASC";
        $stmt = $this->db->prepare($sql);
        $stmt->bindParam(":id", $a);
        $stmt->execute();
        $data = [];
        while ($rows = $stmt->fetch()) {
            $data[] = $rows;
        }
        return $data;
    }
    public function tampilkrit()
    {
        $sql = "SELECT * FROM
kriteria";
        $stmt = $this->db->prepare($sql);
        $stmt->execute();
        $data = [];
        while ($rows = $stmt->fetch()) {
            $data[] = $rows;
        }
        return $data;
    }
    public function hapus()
    {
        $sql = "SELECT * FROM
sub_kriteria WHERE id_sub = :id";
        $stmt = $this->db->prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        $sql = "DELETE FROM
sub_kriteria WHERE id_sub = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        header("location:
subkriteria.php");
    }
    public function update()
    {
        $sql = "SELECT * FROM
sub_kriteria WHERE id_sub = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
        $sql = "UPDATE
sub_kriteria SET keterangan =
:keterangan ,nilai = :nilai WHERE
id_sub = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt -> bindParam(":id",
$_GET['id']);

```

```

        $stmt ->
bindParam(":keterangan",
$_POST['keterangan']);
        $stmt ->
bindParam(":nilai", $_POST['nilai']);
        $stmt->execute();
    }
    public function tambah()
    {
        $sql = "INSERT INTO
sub_kriteria VALUES (:kode,
:keterangan, :nilai)";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":kode",
$_POST['kriteria']);
        $stmt-
>bindParam(":keterangan",
$_POST['keterangan']);
        $stmt->bindParam(":nilai",
$_POST['nilai']);
        $stmt->execute();
        header("location:
subkriteria.php");
    }
    public function simpan()
    {
        $sql = "INSERT INTO
sub_kriteria VALUES (:
:keterangan, :nilai, :id_kriteria)";
        $stmt = $this->db-
>prepare($sql);
        $stmt-
>bindParam(":id_kriteria",
$_POST['kriteria']);
        $stmt-
>bindParam(":keterangan",
$_POST['keterangan']);
        $stmt->bindParam(":nilai",
$_POST['nilai']);
        $stmt->execute();
        header("location:
subkriteria.php");
    }
    public function get_json($val)
    {
        $sql = "SELECT * FROM
sub_kriteria WHERE id_sub=:id
limit 1";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id", $val);
        $stmt->execute();
        //$data['data'] = array();
        $data = $stmt-
>fetchAll(PDO::FETCH_ASSOC);
        return json_encode($data);
    }
}
?>

```

#### 4. Hasil.php

```

<?php
namespace App;

```

```

use PDO;
include "Kriteria.php";
class Hasil extends Koneksi{
    public function __construct()
    {
        $call = new Koneksi();
        $this->db = $call->koneksi();
    }
    public function caribobot()
    {
        $sql = "SELECT * FROM
kriteria";
        $stmt = $this->db-
>prepare($sql);
        $stmt->execute();
        //Fetch
        $data = [];
        while ($rows = $stmt->fetch())
        {
            $data[] = $rows;}
        return $data;
    }
    public function countalt()
    {
        $sql = "SELECT COUNT(id)
AS count FROM alternatif";
        $stmt = $this->db-
>prepare($sql);
        $stmt->execute();
        //Fetch
        $data = [];
        while ($rows = $stmt->fetch())
        {
            $data[] = $rows;}
    }
}
    $data[] = $rows;
    }
    public function count()
    {
        $sql = "SELECT MAX(id) AS
count FROM alternatif";
        $stmt = $this->db-
>prepare($sql);
        $stmt->execute();
        $data = [];
        while ($rows = $stmt->fetch())
        {
            $data[] = $rows;
        }
        return $data;
    }
    public function spk($b)
    {
        //Initilize
        $sql = "SELECT * FROM
alternatif WHERE id = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id", $b);
        $stmt->execute();
        //Fetch
        $data = [];
        while ($rows = $stmt->fetch())
        {
            $data[] = $rows;}
    }
}

```

```

return $data;
}
public function max(){
    $sql = "SELECT MAX(nilai)
AS MAX FROM hasil";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}
public function rank()
{
    $sql = "SELECT * FROM
`hasil`, alternatif WHERE
hasil.id_alternatif=alternatif.id
ORDER BY nilai DESC LIMIT 1";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}
public function min(){
    $sql = "SELECT MIN(nilai) AS
MIN FROM hasil";
    $stmt = $this->db-
>prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch())
    {
        $data[] = $rows;
    }
    return $data;
}
public function hitung0()
{
    $data = array(array());
    $arraykey = 0;
    $hasil = new Hasil();
    $a = $hasil->count();
    $c = $hasil->countalt();
    $no1 = 1;
    foreach($a as $row) {
        $b = $row['count'];
    }
    foreach($c as $row) {
        $d = $row['count'];
    }
    do{
        $hasil = new Hasil();
        $roww=$hasil->spk($no1);
        if($roww== NULL){goto skip;}
        foreach($roww as $rows)
        {

```

```

    $data[$arraykey]['id'] =
$rows['id'];
    $data[$arraykey]['nama'] =
$rows['nama'];
    $data[$arraykey]['perusahaan']
= $rows['perusahaan'];
    //C1 Selection
    if($rows['c1'] <= 1)
    {
        $data[$arraykey]['c1'] = 1;
    }elseif( $rows['c1'] <=10)
    {
        $data[$arraykey]['c1'] = 2;
    }elseif( $rows['c1'] <=20 )
    {
        $data[$arraykey]['c1'] = 3;
    }elseif( $rows['c1'] <=30 )
    {
        $data[$arraykey]['c1'] = 4;
    }else{
        $data[$arraykey]['c1'] = 5;
    }
    //C2 Selection
    if($rows['c2'] <= 100)
    {
        $data[$arraykey]['c2'] = 1;
    }elseif( $rows['c2'] <=300)
    {
        $data[$arraykey]['c2'] = 2;
    }elseif( $rows['c2'] <=600 )
    {
        $data[$arraykey]['c2'] = 3;
    }elseif( $rows['c2'] <=900 )
    {
        $data[$arraykey]['c2'] = 4;
    }else{
        $data[$arraykey]['c2'] = 5;
    }
    //C3 Selection
    if($rows['c3'] > 40)
    {
        $data[$arraykey]['c3'] = 1;
    }elseif( $rows['c3'] >30)
    {
        $data[$arraykey]['c3'] = 2;
    }elseif( $rows['c3'] >20 )
    {
        $data[$arraykey]['c3'] = 3;
    }elseif( $rows['c3'] >10 )
    {
        $data[$arraykey]['c3'] = 4;
    }else{
        $data[$arraykey]['c3'] = 5;
    }
    //C4 Selection
    if($rows['c4'] > 4)
    {
        $data[$arraykey]['c4'] = 1;
    }elseif( $rows['c4'] > 3)
    {
        $data[$arraykey]['c4'] = 2;
    }elseif( $rows['c4'] > 2 )
    {
        $data[$arraykey]['c4'] = 3;
    }

```

```

}elseif( $rows['c4'] > 1 )
{
    $data[$arraykey]['c4'] = 4;
}elseif( $rows['c4'] > 2 )
{
    $data[$arraykey]['c4'] = 5;
}
//C5 Selection
if($rows['c5'] > 3)
{
    $data[$arraykey]['c5'] = 1;
}elseif( $rows['c5'] > 2)
{
    $data[$arraykey]['c5'] = 2;
}elseif( $rows['c5'] > 1 )
{
    $data[$arraykey]['c5'] = 3;
}elseif( $rows['c5'] > 0.5 )
{
    $data[$arraykey]['c5'] = 4;
}elseif( $rows['c5'] > 0 )
{
    $data[$arraykey]['c5'] = 5;
}
//C6 Selection
if($rows['c6'] <= 1)
{
    $data[$arraykey]['c6'] = 1;
}elseif( $rows['c6'] <=10)
{
    $data[$arraykey]['c6'] = 2;
}elseif( $rows['c6'] <=20 )
{
    $data[$arraykey]['c6'] = 3;
}
}elseif( $rows['c6'] <=30 )
{
    $data[$arraykey]['c6'] = 4;
}elseif( $rows['c6'] <= 40)
{
    $data[$arraykey]['c6'] = 5;
}
//C7 Selection
if($rows['c7'] <= 1)
{
    $data[$arraykey]['c7'] = 1;
}elseif( $rows['c7'] <=10)
{
    $data[$arraykey]['c7'] = 2;
}elseif( $rows['c7'] <=20 )
{
    $data[$arraykey]['c7'] = 3;
}elseif( $rows['c7'] <=30 )
{
    $data[$arraykey]['c7'] = 4;
}elseif( $rows['c7'] <= 40)
{
    $data[$arraykey]['c7'] = 5;
}
}
$arraykey++;
skip:
$no1++;
} while($no1 <= $b);
return $data;
}
public function hitung()
{

```

```

$data = array();
for($i=1;$i<=7;$i++)
{
    $data[$i]=0;
}
$Hasil = new Hasil();
$a = $Hasil->count();
$c = $Hasil->countalt();
foreach($a as $row) {
    $b = $row['count'];
}
foreach($c as $row) {
    $d = $row['count'];
}
$hasil = new Hasil();
$roww=$hasil->hitung0();
foreach($roww as $rows) {
    //Selection
    for($i=1;$i<=7;$i++)
    {
        $data[$i] = $data[$i] +
pow($rows['c'].$i,2);
    }
}
$sqrtc1 = sqrt($data[1]);
$sqrtc2 = sqrt($data[2]);
$sqrtc3 = sqrt($data[3]);
$sqrtc4 = sqrt($data[4]);
$sqrtc5 = sqrt($data[5]);
$sqrtc6 = sqrt($data[6]);
$sqrtc7 = sqrt($data[7]);
$data = array(
    '1' => $sqrtc1,
    '2' => $sqrtc2,
    '3' => $sqrtc3,
    '4' => $sqrtc4,
    '5' => $sqrtc5,
    '6' => $sqrtc6,
    '7' => $sqrtc7,
);
return $data;
}
public function hitung2()
{
    $hasil = new Hasil();
    $akaran = $hasil->hitung();
    for($i=1;$i <=7; $i++)
    {
        ${'akarc'.$i} = $akaran[$i];
    }
    $no1= 1;
    $data = array(array());
    $hasil = new Hasil();
    $a = $Hasil->count();
    foreach($a as $row) {
        $b = $row['count'];
    }
    $arraykey = 0;
    $hasil = new Hasil();
    $roww=$hasil->hitung0();
    foreach($roww as $rows) {
        //Selection
        for($i=1;$i <=8; $i++)
        {

```

```

        if($i==8){
            $data[$arraykey][$i] =
            round($rows['id'];
                }else{
                    $data[$arraykey][$i] =
                    round($rows['c'. $i]/${'akarc'. $i},4);
                }
            }
            $data[$arraykey][0] =
            $rows['nama'];
            $arraykey++;
        }
        //Tester Variabel
        //echo "Penghuni : ".$penghuni.'
        Nilai : ". $nilai3 ." Utility : ". $C3 .
        '<br>';
        return $data;
    }
    public function hitung3()
    {
        $hasil = new Hasil();
        $bobot = $hasil->caribobot();
        $dist=1;
        $setc = 0;
        foreach ($bobot as $a)
        {
            $setc +=
            $a['bobot_awal']*$a['bobot'];
        }
        foreach($bobot as $p)
        {
            if($i==8){
                $data[$arraykey][$i] =
                round(($p['bobot']*$p['bobot_awal'])
                    /$setc,2);
                $dist++;
            }
            $data = array(array());
            $arraykey = 0;
            $normal = $hasil->hitung2();
            foreach($normal as $a)
            {
                for($i=1;$i <=8; $i++)
                {
                    if($i==8){
                        $data[$arraykey][$i] =
                        $a[$i];
                    }else{
                        $data[$arraykey][$i] =
                        round($a[$i]*${'bobotc'. $i},4);
                    }
                }
            }
            $data[$arraykey][0] = $a[0];
            $arraykey++;
        }
        return $data;
    }
    public function hitung4($set = 0)
    {
        $data = array(array());
        $ymaxmin = array(array());
        $arraykey = 0;
        $max = 0;$min=0;$alt=0;
        $keynew = 0;

```







```

if($start AND $end) {
    $sql = "SELECT * FROM logs
WHERE DATE(tanggal) >= :start
AND DATE(tanggal) <=:end
ORDER BY tanggal DESC";
    $stmt = $this->db->prepare($sql);
    $stmt->bindParam(":start", $start);
    $stmt->bindParam(":end", $end);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch()) {
        $data[] = $rows;
    }
    return $data;
} else {
    $sql = "SELECT * FROM logs
ORDER BY tanggal DESC";
    $stmt = $this->db->prepare($sql);
    $stmt->execute();
    $data = [];
    while ($rows = $stmt->fetch()) {
        $data[] = $rows;
    }
    return $data;
}

public function logs($a)
{
    if( $_GET['id']== TRUE)
    {
        $sql = "SELECT * From
tbl_login WHERE id_login = :id";
        //query mencari datanya
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id",
$_GET['id_login']); //statement bisa
diganti apa aja
        $stmt->execute();
        $login = $stmt->fetch();
        if ($stmt->rowcount() == 0) {
            echo
            '<script>alert("Failed");</script>';//ti
            dak ada datanya
        } else {
            $sql = "INSERT INTO logs
(id_log, keterangan, id_login)
VALUES (, :user :txt, :id_login)";
            $stmt = $this->db-
>prepare($sql);
            $stmt->bindParam(":txt", $a);
            $stmt-
>bindParam(":id_login",
$login['id_login']);
            $stmt->bindParam(":user",
$login['username']);
            $stmt->execute();
        }
    } else {
        $sql = "INSERT INTO logs
(id_log, keterangan, id_login)
VALUES (, :user :txt :kode ,
:id_login)";
    }
}

```

```

    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":kode",
$_POST['kode']);
    $stmt->bindParam(":txt", $a);
    $stmt->bindParam(":id_login",
$_POST['id_login']);
    $stmt->bindParam(":user",
$_POST['user']);
    $stmt->execute();
}
}
public function hapus()
{
    $sql = "SELECT * FROM
tbl_login WHERE id_login = :id";
    $stmt = $this->db->prepare($sql);
    $stmt->bindParam(":id",
$_GET['id']);
    $stmt->execute();
    $sql = "DELETE FROM
tbl_login WHERE id_login = :id";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam(":id",
$_GET['id']);
    $stmt->execute();
    $login = $stmt->fetch();
    //CEK FIELD PASSWORD
TIDAK NULL
    if($_POST['pass']==NULL){
        $sql = "UPDATE tbl_login
SET nama=:nama,
username=:username WHERE
id_login = :id";
    if($_GET['acc']=="Admin"){goto
a;}
    if($_GET['username']==$_POST['us
ername']){goto a;}
    $sql = "SELECT * FROM
tbl_login WHERE
username=:username";
    $stmt = $this->db->prepare($sql);
    $stmt -> bindParam(":username",
$_POST['username']);
    $stmt->execute();
    if ($stmt->rowcount(>0) {
        echo '<script>alert("Username
Sudah dipakai");</script>';//Kode
Sudah Ada
    }else{
        a:
        $sql = "SELECT * FROM
tbl_login WHERE id_login = :id";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam(":id",
$_GET['id']);
        $stmt->execute();
    }
}
}
public function update()
{

```

```

        $stmt = $this->db-
>prepare($sql);
        $stmt -> bindParam(":id",
$_GET['id']);
        $stmt ->
bindParam(":username",
$_POST['username']);
        $stmt -> bindParam(":nama",
$_POST['nama']);
        $stmt->execute();
    }else{
        $sql = "UPDATE tbl_login
SET nama=:nama,
username=:username, password =
md5(:password) WHERE id_login =
:id";
        $stmt = $this->db-
>prepare($sql);
        $stmt -> bindParam(":id",
$_GET['id']);
        $stmt ->
bindParam(":username",
$_POST['username']);
        $stmt -> bindParam(":password",
$_POST['pass']);
        $stmt -> bindParam(":nama",
$_POST['nama']);
        $stmt->execute();
    }
//User RELOG

        $sql = "SELECT * From
tbl_login WHERE id_login = :id";
        //query mencari datanya
        $stmt = $this->db-
>prepare($sql);
        $stmt -> bindParam(":id",
$_GET['id']);
        $stmt->execute();
        $login = $stmt->fetch();
        if ($stmt->rowcount() == 0) {
            echo '<script>alert("Login
Gagal, Periksa Kembali Username
dan Password");
            window.location-
"index.php"</script>';//tidak ada
datanya
        } else {
            session_start();
            $_SESSION["ID"] =
$login['id_login'];
            $_SESSION["nama"] =
$login['username'];
            $_SESSION["namapengguna"] =
$login['nama'];
            $_SESSION["status"] =
$login['level'];
        }
    }
    public function tambah()
    {

```

```

        $sql = "INSERT INTO
tbl_login VALUES (':username
,:nama, md5(:password), :level)";
        $stmt = $this->db-
>prepare($sql);
        $stmt->bindParam("':username",
$_POST['username']);
        $stmt->bindParam("':level",
$_POST['level']);
        $stmt->bindParam("':password",
$_POST['pass']);
        $stmt->bindParam("':nama",
$_POST['nama']);
        $stmt->execute();
        header("location: user.php");
    }

public function get_json($val)
{
    $sql = "SELECT * FROM
tbl_login WHERE id_login=:id limit
1";
    $stmt = $this->db-
>prepare($sql);
    $stmt->bindParam("':id", $val);
    $stmt->execute();
    //$data['data'] = array();
    $data = $stmt-
>fetchAll(PDO::FETCH_ASSOC);
    return json_encode($data);
}
?>

```

