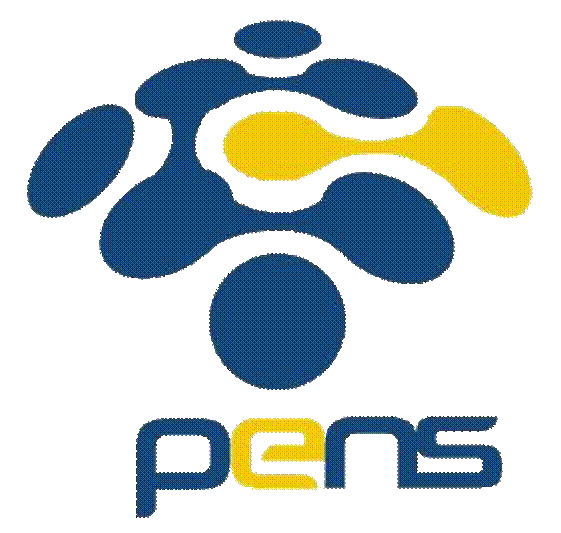
**Laporan Project Akhir**

**Sistem Pendukung Keputusan**

**(AHP) – Kelompok 8**



Dosen :

**Dr Arna Fariza S.Kom., M.Kom.**

Disusun oleh:

**Asy Syaffa Khoirunnisa - 3121500040**

**Muhammad Daniel Krisna Halim Putra - 3121500047**

**Nabila Balqis Azzahra - 3121500059**

**D3 IT B**

**PROGRAM STUDI D3 TEKNIK INFORMATIKA**

**Politeknik Elektronika Negeri Surabaya**

**2021/2022**

**PROGRAM MADM AHP (ANALYTICAL HIERACHY PROCESS)**

**DALAM BAHASA PYTHON**

**Deskripsi Masalah :**

Manajer pembelian Suroboyo Bus bertanggung jawab untuk memutuskan perusahaan mana yang tersedia untuk membeli 50 bus untuk rute baru. Setelah studi kelayakan awal dan analisis merek, model beberapa bus, dan proses yang terkait dengan pembelian, ia mempersempit daftar alternatif keputusannya menjadi tiga opsi;

1. **Proyek 1**: Pembelian model Eksekutif,
2. **Proyek 2**: Pembelian model Medium, dan
3. **Proyek 3**: Pembelian model Ekonomi.

Suroboyo Bus memutuskan bahwa kriteria berikut relevan untuk proses keputusan pembeliannya:

1. **Mil per galon (MPG)**
2. **Kenyamanan**
3. **Gaya**
4. **Harga**

**Penjelasan**

**Cara menjalankan program**

Sebelumnya, pada sebuah pemrograman pasti terdapat libraries. Libraries sendiri yaitu kumpulan modul terkait berisi kumpulan kode yang dapat digunakan berulang kali dalam program yang berbeda. Library ini membuat Pemrograman Python lebih sederhana dan nyaman bagi programmer. Disini kami menyediakan 2 cara mengeksekusi program, yaitu:

1. Menggunakan [*https://colab.research.google.com/*](https://colab.research.google.com/)*.* Masuk pada website pilih file kemudian upload notebook, maka tinggal upload file dengan format .ipynb,
2. Menggunaakan Compiler seperti Visual Studio Code, sebelum menjalankan program, wajib wajib menginstall PIP, PIP merupakan program untuk manajemen paket di Python. Tugasnya untuk menginstal, menghapus, upgrade paket Python, dll. Masuk terlebih dahulu ke git bash/cmd, lalu ketik ***python get-pip.py.*** Setelah selesai install library-librarynya dengan mengetikkan ***Pip install pandas*** dan juga ***pip install matplotlib***. Maka program sudah bisa dijalankan karena library yang dibutuhkan sudah memenuhi.

Pandas sendiri berfungsi mengolah suatu data seperti teknik join, distinct, group by, agregasi, dan teknik lainnya seperti pada SQL. Bedanya, ini dilakukan pada tabel. Kelebihan dari library ini juga dapat membaca file dari berbagai format seperti .txt, . csv, dan .tsv. Sedangkan matpolib berfungsi untuk memvisualisasikan data secara 2D maupun 3D dan menghasilkan gambar berkualitas yang bahkan dapat kamu simpan dalam berbagai format gambar, seperti JPEG dan PNG.

A screenshot of a computer

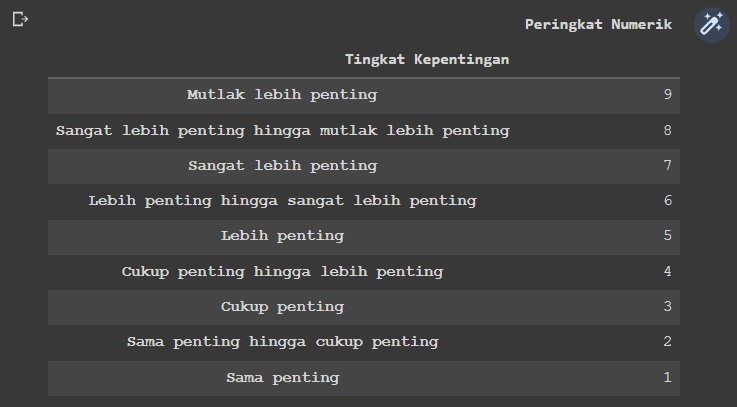
Description automatically generated with medium confidence

**Tabel 1: Penilaian Verbal ke Peringkat Numerik**

Text

Description automatically generated

Kita deklarasikan variabel Rating\_dic dengan menampung tipe data string lalu pada ‘Tingkat Keputusan’ akan menampung list data tablenya yang natinya akan dimasukkan/dikombinasikan dengan library panda. Kemudian adda juga Peringkat Numerik yang menampung number untuk disandingkan dengan list data Tingkat kepentingan. Selanjutnya bisa dilihat pada variable Rating\_comparison, disini variablenya menampung pd, pd sendiri adalah pandas yang sudah di aliaskan. Pd akan dihubungkan dengan class DatFrame yang menampung variablw Rating\_dic tadi, lalu jangan lupa juga membuat method set\_index yang berfungsi meberikan awalan kolom yang ditampilkan yaitu disini Tingkat Kepentingam. Maka ketika Rating\_comparison dicetak akna menghasilkan:



**Tabel 2: Perbandingan Berpasangan Empat Kriteria Soal Pemilihan Bus**

**Text

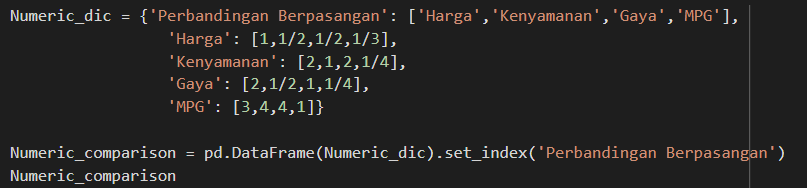
Description automatically generated with medium confidence**

Masuk pada tabel ke 2 kita bandingkan berpasangan dengan 4 kriteria soal pemilihan terdapata Perbandingan Berpasangan, Kriteria Terpenting, Seberapa Penting, dan juga Peringkat Numerik. Sama halnya dengan tadi, kita juga deklarasikan variable comparison\_dic yang menampung tipe data string dan didalamnya terdapat list data dan terdapat tambahan kolom table dan perubahan nama table. Berikut outputnya:

A screenshot of a computer

Description automatically generated

* **Konversi Penilaian Verbal ke Peringkat Numerik**



Disini kita masuk pada penilaian verbal ke peringkat numerik, Kita inisialisasikan Numeric\_dic yang berisi tipe data string dan diberikan value didalamnya. Kemudia kita deklarasika lagi Numeric\_comparison dengan isi class DataFrame yang beris Numeric\_dic dan kita set\_index untuk menampilkan Perbandingan berpasangan pada awal table.

A screenshot of a computer screen

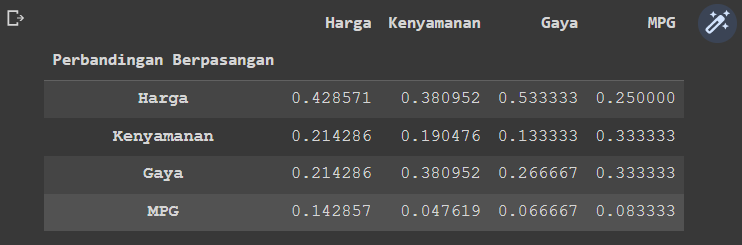
Description automatically generated with medium confidence

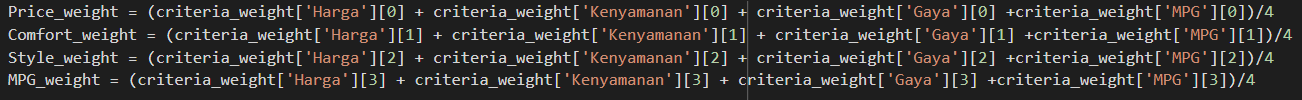
* **Menghitung bobot untuk setiap kriteria**

Text

Description automatically generated

Kita buat variable criteria\_weight yang mana menampung sebuah salinan objek baru dari Numeric\_comparison yang menampung DataFrame dari Numeric\_dic. Lalu kita berikan for pada column supaya perintah for yang digunakan bisa melakukan iterasi dari sebuah nilai sequence atau data koleksi pada python seperti List, Tuple, String dan lain-lain. Disini kita for pada Harga, kenyamanan gata MPG yang nantinya pad criteria\_weight akan dimasukkan column dan akan di Sum() atau dijumlahkan seluruh anggota list arraynya. Dan hasilnya sebagai berikut:

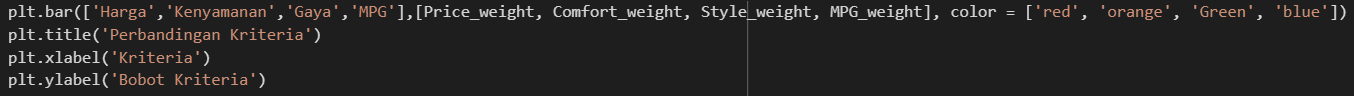




Disini kita buat Price\_weight, Comfort\_weight, Style\_weight, MPG\_weight yang akan menampung rumus untuk mendapatkan bobot. Yang mana criteria\_weight akan di berikan variable global supaya bisa mengambil Harga yang ada pada list list sebelumnya dan diberikan array 0 untuk memanggil kolom 1 dan dijumlahkan dengan kenyamanan pada array 0 juga, dijumlahkan lagi dengan gaya serta MPG lalu terakhir dibagi dengan 4 karena total perbandingan ada 4. Begitu juga berikutnya.



Lanjut kita cetak menggunakan print dan Bobot harga akn mencetak price\_weight dengan F mengacu pada "Format desimal titik mengambang". Angka .3 menunjukkan pembulatan ke 3 tempat setelah titik desimal. Begitu juga seterusnya.



Kita visualisasikan data ke 2D berbentuk diagram bar dengan syntax plt adalah alias dari library matplotlib. Kita isi barnya dengan Harga, kenyamann, gaya, MPG, lalu diisi dengan rice\_weight, Comfort\_weight, Style\_weight, MPG\_weight diberikan juga warna red, orange, green, blue. Kita buat titlenya ‘Perbandingan Kriteria’, label x atau horizontal diisi dengan kriteria dan y vertikal diisi label Bobot Kriteria. Maka akan mencetak:

Chart, bar chart

Description automatically generated

Hasil bobot di atas menunjukkan bahwa kriteria Harga merupakan kriteria yang paling penting dari semua kriteria, dengan bobot 0.398, disusul kriteria Gaya dengan bobot 0.299, disusul kriteria Kenyamanan dengan bobot 0.218, dan kriteria MPG dengan bobot 0.085.

**Tabel 3: Perbandingan Berpasangan yang menunjukkan Preferensi untuk proyek berdasarkan kriteria**

* **Kriteria Harga**

Text

Description automatically generated

Deklarasikan PRICE\_dic yang akan memuat string dari HARGA dan isi dari column HARGA ada Proyek 1, Proyek 2, Proyek 3 . lalu pada Proyek 1, 2, 3 diisi dengan hasil perbandingan. Pada variable PRICE\_comparison akan diisi dengan DataFrame dari PRICE\_dic dan diawali dengan column HARGA. Hasil yang keluar sebagai berikut:

Graphical user interface, text, application

Description automatically generated

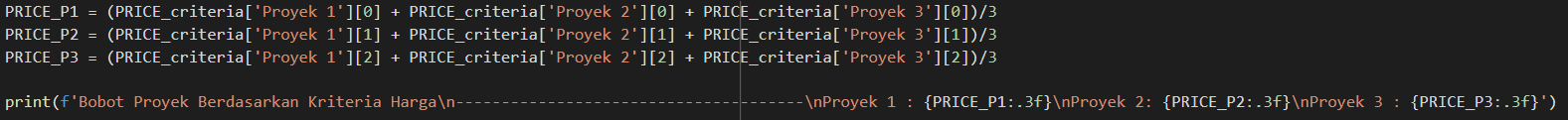
**Text

Description automatically generated**

Dilanjutlan kita copy untuk objek baru dari PRICE\_comparison dan kita berikan for pada column supaya perintah for yang digunakan bisa melakukan iterasi dari sebuah nilai sequence atau data koleksi pada python seperti List, Tuple, String dan lain-lain. Disini kita for pada Proyek 1, 2, dan 3 yang nantinya pada PRICE\_criteria akan dimasukkan column dan akan di Sum() atau dijumlahkan seluruh anggota list arraynya. Dan hasilnya sebagai berikut:

**Graphical user interface, application

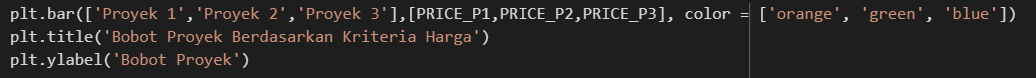
Description automatically generated**

****

Kita buat PRICE\_P1, P2, dan P3 yang akan menampung rumus untuk mendapatkan bobot proyek berdasarkan kriteria harga. Yang mana PRICE\_criteria akan di berikan variable global supaya bisa mengambil Harga yang ada pada list list sebelumnya dan diberikan array 0 untuk memanggil kolom 1 dan dijumlahkan dengan Proyek 2 pada array 0 juga, dijumlahkan lagi dengan Proyek 3 serta dibagi dengan 3 karena total HARGA ada 3. Begitu juga berikutnya. Dan cetak menggunakan print, panggil PRICE\_P1, P2, dan P3 berikan .3F yang mana dengan F mengacu pada "Format desimal titik mengambang". Angka .3 menunjukkan pembulatan ke 3 tempat setelah titik desimal. Begitu juga seterusnya.

**A black screen with white text

Description automatically generated with low confidence**

****

Kita visualisasikan data ke 2D berbentuk diagram bar dengan syntax plt adalah alias dari library matplotlib. Kita isi barnya dengan Proyek 1, Proyek 2, Proyek 3, diisi dengan PRICE\_P1, PRICE\_P2, PRICE\_P3, berikan warna orange, green, dan blue lalu berikan title ‘Bobot Proyek Berdasarkan Kriteria Harga', dan y label dengan naman Bobot Proyek. Maka akan mencetak:

**Chart, bar chart

Description automatically generated**

Hasil di atas menunjukkan bahwa Proyek 3, pembelian model Ekonomi, memiliki bobot tertinggi 0.557. Ini berarti jika 3 proyek dianalisis berdasarkan kriteria Harga saja, Proyek 3 akan menjadi kandidat terbaik untuk diseleksi.

* **Kriteria MPG**

**A picture containing graphical user interface

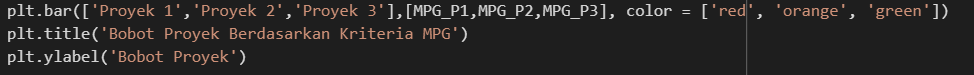
Description automatically generated**

Deklarasikan MPG\_dic yang akan memuat string dari MPG dan isi dari column MPG ada Proyek 1, Proyek 2, Proyek 3 . lalu pada Proyek 1, 2, 3 diisi dengan hasil perbandingan. Pada variable MPG\_comparison akan diisi dengan DataFrame dari MPG\_dic dan diawali dengan column MPG. Dilanjutlan kita copy untuk objek baru dari MPG\_comparison dan kita berikan for pada column supaya perintah for yang digunakan bisa melakukan iterasi dari sebuah nilai sequence atau data koleksi pada python seperti List, Tuple, String dan lain-lain. Disini kita for pada Proyek 1, 2, dan 3 yang nantinya pada MPG\_criteria akan dimasukkan column dan akan di Sum() atau dijumlahkan seluruh anggota list arraynya. Cetak menggunakan print, panggil MPG\_P1, P2, dan P3 berikan .3F yang mana dengan F mengacu pada "Format desimal titik mengambang". Angka .3 menunjukkan pembulatan ke 3 tempat setelah titik desimal. Begitu juga seterusnya.

Hasil yang keluar sebagai berikut:

**Text

Description automatically generated**

****

Visualisasikan data ke 2D berbentuk diagram bar dengan menggunakan library matplotlib alias plt. Kita isi barnya dengan Proyek 1, Proyek 2, Proyek 3, diisi dengan MPG\_P1, MPG\_P2, MPG\_P3, berikan warna red, orange, green. Berikan title ‘Bobot Proyek Berdasarkan Kriteria MPG, dan berikan y vertikal dengan label Bobot Proyek. Maka akan menghasilkan diagram data:

**Chart, bar chart

Description automatically generated**

Hasil di atas menunjukkan bahwa Proyek 3, pembelian model Ekonomi, memiliki bobot tertinggi 0.639. Ini berarti jika 3 proyek dianalisis berdasarkan kriteria MPG saja, Proyek 3 akan menjadi kandidat terbaik untuk diseleksi.

* **Kriteria Kenyamanan**

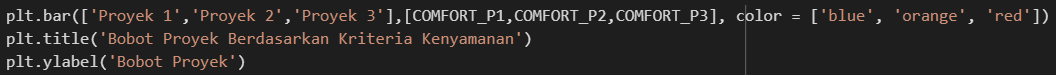
Text

Description automatically generated

COMFORT\_dic dideklarasikan lalu, akan memuat string dari COMFORT yang berisi Proyek 1, Proyek 2, Proyek 3 . Isikan pada semua Proyek dengan hasil perbandingan. Variable COMFORT \_comparison akan diisi dengan DataFrame dari COMFORT \_dic dan diawali dengan column COMFORT. Copy untuk objek baru dari COMFORT \_comparison menggunakan syntax copy() dan berikan for di column agar perintah for yang dipakai bisa melakukan iterasi dari sebuah nilai sequence atau data koleksi pada python seperti List, Tuple, String dan lain-lain. Nantinya pada COMFORT \_criteria akan dimasukkan column dan akan di jumlahkan dengan seluruh anggota list arraynya. Gunakan Print, panggil COMFORT\_P1, P2, dan P3 berikan .3F yang mana dengan F mengacu pada "Format desimal titik mengambang". Angka .3 menunjukkan pembulatan ke 3 tempat setelah titik desimal. Begitu juga seterusnya. Maka hasil outputnya:

Text

Description automatically generated



Data akan divisualisasikan ke bentuk diagram bar 2D dengan menggunakan library matplotlib. Isi barnya sebagai Proyek 1, Proyek 2, Proyek 3, diisi COMFORT\_P1, COMFORT \_P2, COMFORT \_P3, dengan warna blue, orange, red. Title dari diagramnya ‘Bobot Proyek Berdasarkan Kriteria Kenyamanan, tidak lupa diberi label vertikal dengan nama Bobot Proyek. Maka akan tercetak diagram sebagai berikut:

Chart, bar chart

Description automatically generated

Hasil di atas menunjukkan bahwa Proyek 1, pembelian model Eksekutif, memiliki bobot tertinggi sebesar 0.593. Ini berarti jika 3 proyek dianalisis berdasarkan kriteria Kenyamanan saja, Proyek 1 akan menjadi kandidat terbaik untuk diseleksi.

* **Kriteria Gaya**

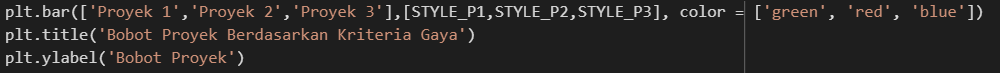
Chart

Description automatically generated with low confidence

Buat deklarasi STYLE\_dic, kemudian akan memuat string dari COMFORT berisi Proyek 1, Proyek 2, Proyek 3 . Masukkan pada semua Proyek hasil perbandingannya. Variable STYLE\_comparison akan diisi dengan DataFrame dari STYLE\_dic dan diawali dengan column STYLE. Buat salinan objek baru dari STYLE\_comparison memakai syntax copy(). Selanjutnya For di column berfungsi supaya bisa melakukan iterasi dari sebuah nilai sequence atau data koleksi pada python seperti List, Tuple, String dan lain-lain. Nantinya pada STYLE\_criteria akan dimasukkan column yang nantinya di jumlahkan dengan seluruh anggota list arraynya. Print untuk mencetak dengan memanggil variable STYLE\_P1, P2, dan P3 berikan .3F yang mana dengan F mengacu pada "Format desimal titik mengambang". Angka .3 menunjukkan pembulatan ke 3 tempat setelah titik desimal. Begitu juga seterusnya. Hasil yang tercetak sebagai berikut:

Text

Description automatically generated



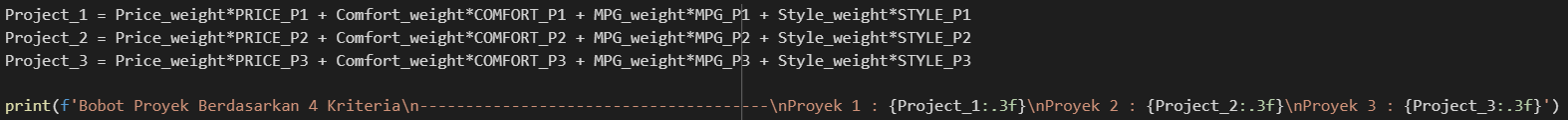
Agar tercetak data diagram gunakan syntax plt disertai dengan bar, plt sendir diambil dari library matplotlib. Pada bar masukkan Proyek 1, Proyek 2, Proyek 3, isi dengan STYLE\_P1, STYLE\_P2, STYLE\_P3, dengan warna blue, orange, red. Title dari diagramnya ‘Bobot Proyek Berdasarkan Kriteria Gaya, beri label pada y vertikal dengan nama Bobot Proyek. Dat Diagram akan terlihat berikut:

Bar chart

Description automatically generated with medium confidence

Hasil di atas menunjukkan bahwa Proyek 2, pembelian model Medium, memiliki bobot tertinggi 0.656. Ini berarti jika 3 proyek dianalisis berdasarkan kriteria Gaya saja, Proyek 2 akan menjadi kandidat terbaik untuk diseleksi.

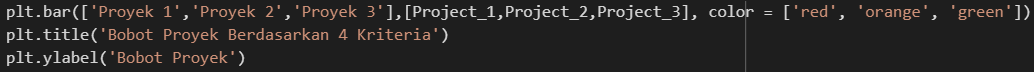
* **Keempat Kriteria**



Dari Project 1, Project 2, Project 3 kita berikan value Price\_weight (diambil dari Perbandingan Berpasangan 4) yang akan dikalikan dengan PRICE\_P1 (mengambil data dari Kriteria Harga) lalu dijumlahkan dengan Comfort\_weight (diambil dari Perbandingan Berpasangan 4) dikalikan dengan COMFORT\_P1 (diambil data dari Kriteria Kenyamanan) ditambahkan dengan MPG\_weight (diambil dari Perbandingan Berpasangan 4) dikalikan dengan MPG\_P1 (mengambil dat dari Kriteria MPG) dan terakhir ditambahkan dengan Style\_weight (diambil dari Perbandingan Berpasangan 4) dikalikan dengan STYLE\_P1 (mengambil data dari Kriteria Gaya). Kita Print mencetak hasil, tinggal panggil variable STYLE\_P1, P2, dan P3 berikan .3F yang mana dengan F mengacu pada "Format desimal titik mengambang". Angka .3 menunjukkan pembulatan ke 3 tempat setelah titik desimal. Hasil yang tercetak:

Text

Description automatically generated



Membuat data diagram dengan menggunakn library matplotlib yang dialiaskan menjadi plt supaya ringkas. Pada bar masukkan Proyek 1, Proyek 2, dan Proyek 3. Kemudian pada setiap Proyek diisi dengan STYLE\_P1, STYLE\_P2, STYLE\_P3, berikan warna pada setiap Proyek supaya mudah membedakan dengan blue, orange, red. Berikan Title dari data diagramnya ‘Bobot Proyek Berdasarkan Kriteria Gaya, terakhir berikan nama Bobot Proyek pada label dengan posisi vertikal (y). Alhasil Data Diagram akan nampak sebagai berikut:

Bar chart

Description automatically generated

Hasil akhir di atas menunjukkan bahwa proyek terbaik berdasarkan kombinasi keempat kriteria adalah Proyek 2, pembelian model Medium.

**Hasil Menggunakan Compiler**

Text

Description automatically generated