

IMPLEMENTASI SISTEM PENGAMBILAN  
KEPUTUSAN DALAM PEMILIHAN  
KARYAWAN

## **BAB I**

Pada setiap perusahaan dituntut untuk memiliki kenaikan target maupun keseimbangan target dalam hal marketing ataupun target pasarnya, sehingga perusahaan dapat terus berkembang dan maju.

Dalam hal ini untuk melakukan pencapaian tersebut, diperlukan sumber daya manusia (SDM) yang mumpuni baik itu pada pimpinan perusahaan maupun staff perusahaan.

Salah satu kegiatan yang dapat mendukung pencapaian tersebut yaitu dengan memberikan fasilitas tambahan bidang ilmu baik itu untuk pimpinan maupun staff yang diadakan untuk internal perusahaan maupun hal luas, dalam hal ini perusahaan dapat memberikan fasilitas pengetahuan seperti memberikan seminar, training, diklat, sertifikasi, dan lain-lain.

Dengan adanya kegiatan tersebut maka perusahaan tersebut diharuskan memilih karyawan atau peserta kegiatan yang tepat dan sesuai dengan bidangnya sehingga kesesuaian tersebut dapat menjadi nilai tambah bagi karyawan karena mendapatkan ilmu yang lebih banyak, sehingga dapat membantu dalam memperluas pengetahuan dan perusahaan mendapatkan keuntungan dalam hal sumber daya manusia yang dimiliki memiliki ilmu yang mumpuni untuk kemajuan perusahaan tersebut.

Dalam pemilihan karyawan ini diperlukan nya beberapa kriteria untuk penyeimbangan banyak faktor, kriteria, yang

dilakukan agar setiap keputusan yang dilakukan tidak melakukan kesalahan dan pada orang yang tepat. Keputusan mungkin merupakan pilihan sederhana antara dua atau lebih alternatif yang terdefinisi dengan baik namun, seringkali masalah keputusan merupakan masalah yang agak rumit yang mencakup informasi yang sifatnya kompleks dan saling bertentangan yang mencerminkan berbagai perspektif. Salah satu pendekatan untuk menyelesaikan masalah tersebut dapat dilakukan dengan menggunakan suatu sistem pemilihan karyawan menggunakan metode *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS).

Hasil dari penerapan metode Topsis ini dapat menjadi suatu acuan nilai dengan hasil pemilihan yang sesuai dengan kriteria yang ditentukan.

## **BAB II**

Berdasarkan uraian dan penjelasan dalam bab sebelumnya. Untuk mewujudkan Sistem Pemilihan Karyawan tersebut memerlukan beberapa *tools* atau alat yang mendukung sistem, bahasa pemrograman yang digunakan, *text editor code* yang

digunakan, dan lain-lain. Berikut merupakan penjelasan mengenai masing-masing pendukung pembuatan sistem yang akan dibuat :

## **2.1 Bahasa Pemrograman**

### **2.1.1 PHP**

PHP merupakan bahasa pemrograman scripting karena bahasa pemrograman PHP diproses pada web server, sehingga semua kode-kode yang ditulis menggunakan PHP akan dijalankan dan disimpan di web server maka PHP menyatu dengan HTML (Kode dasar website) dan dijalankan pada server side. Artinya, semua sintaks PHP yang diberikan akan sepenuhnya dijalankan pada server, sedangkan yang dikirimkan ke browser hanya hasilnya saja.

PHP digunakan untuk membuat website dinamis dan interaktif. Dinamis yaitu website tersebut dapat berubah-ubah tampilan dan kontennya sesuai kondisi tertentu.

Terdapat dua cara yang dapat dilakukan untuk menulis kode-kode PHP untuk membangun sebuah aplikasi dinamis, seperti menggunakan PHP native dan framework.

#### **a. PHP Native**

PHP Native merupakan metode penulisan kode-kode bahasa pemrograman PHP yang benar-benar dimulai dari awal atau dari nol.

Pada PHP native ini programmer akan membuat kerangka kerjanya sendiri untuk mengembangkan website dinamis. Biasanya pemula akan disarankan untuk mempelajari PHP Native.

#### b. Framework PHP

Untuk memudahkan pengembangan pada aplikasi web dengan menggunakan bahasa pemrograman PHP maka gunakanlah framework PHP, karena terdapat beberapa keuntungan yang bisa didapatkan ketika membuat website menggunakan framework, diantaranya :

- 1) Tidak perl melakukan pengulangan coding
- 2) Aplikasi web yang dibangun dengan framework akan lebih stabil
- 3) Mengembangkan aplikasi web dapat lebih cepat
- 4) Membantu pemula dalam membuat aplikasi

Adapun beberapa framework PHP yang sering didengar oleh para developer web, yaitu :

- 1) CodeIgniter



Gambar Logo CodeIgniter

2) Laravel



Gambar Logo Laravel

3) *Symfony*



Gambar Logo *Symfony*

4) Zend Framework



Gambar logo Zend Framework

5) Cake PHP



Gambar Logo Cake PHP

6) Seagull



Gambar Logo Framework Seagull

PHP dimulai dengan tulisan antara tag :

- a. `<? Dan ?>`
- b. `<?php dan ?>`
- c. `<script language="php">` dan `</script>`
- d. `<% dan %>`
- e. Setiap *statement* atau perintah yang digunakan biasanya diakhiri dengan titik-koma (;)
- f. *Case Sensitive* untuk nama identifier yang dibuat user yaitu berupa variable, konstanta, fungsi, dan lain-lain. Namun tidak *Case Sensitive* untuk identifier built-in dari PHP. Maka :
  - a) `$nama ≠ $Nama ≠ NAMA`
  - b) `hitungLuas() ≠ HitungLuas()`
  - c) `echo = ECHO`
  - d) `while = WHILE`
- g. Mudah dipahami

### 2.1.2 Variabel

Variabel merupakan suatu lokasi penyimpanan yang berisikan nilai atau informasi yang nilainya tidak diketahui maupun telah diketahui.

Nilai dari variabel dapat di isi dengan informasi yang diinginkan dan dapat diubah nilainya pada saat kode program sedang berjalan. Sebuah variabel memiliki nama yang digunakan untuk mengakses nilai dari variabel itu. Jika anda memiliki pengetahuan dasar tentang bahasa pemrograman, tentunya tidak asing dengan istilah variabel.



- a. Variabel digunakan untuk menyimpan sebuah value, informasi atau data
- b. Penamaan variable diawali dengan tanda \$
- c. Panjang yang digunakan tidak ada batas maksimum penggunaan
- d. Setelah menggunakan tanda \$ diawali oleh huruf atau *under-score* (\_). Maka karakter yang berikutnya bias terdiri dari angka, huruf dan karakter tertentu yang diperbolehkan.
- e. Case yang digunakan bersifat case-sensitive  
PHP membedakan variabel yang ditulis dengan huruf besar dan kecil (bersifat case sensitive), contoh \$nilai\_kuadrat tidak sama dengan \$NilaiKuadrat dan \$NILAIKUADRAT ketiganya akan dianggap sebagai variabel yang berbeda.
- f. Tidak boleh mengandung spasi
- g. Tidak perlu di deklarasikan.

### **2.1.3 Tipe Data**

Pada PHP tipe data *variable* secara otomatis ditentukan oleh interpreter PHP. Tetapi, PHP mendukung 8 buah tipe data primitive, seperti :

- a. Integer

Integer merupakan semua angka bulat baik itu positif ataupun negative bukan pecahan, seperti : 1,2,3,5,dst

b. Float

Float atau floating point atau yang disebut dengan nomor pecahan atau juga bilangan real, seperti : 0.1,0.25,0.2, 0.314E1,dst.

c. Boolean

Boolean merupakan tipe data standar, Boolean hanya menyatakan kebenaran dengan kata kunci perintah TRUE (benar) atau FALSE (salah).

d. String

String merupakan rangkaian karakter, sehingga karakter tersebut sama seperti byte. String dapat didefinisikan dengan cara yang paling mudah yaitu menggunakan tanda satu petik (‘’) terdapat 265 karakter yang dapat dijadikan string.

e. Array

Array merupakan jenis tipe data yang unik, berbeda dengan tipe data lainnya, tipe data array dapat menyimpan satu atau lebih data dalam variabel tunggal. Tipe data array berguna untuk menyimpan banyak data dalam satu variabel.

f. Resource

Resource merupakan jenis tipe data yang menyimpan sebuah referensi external resource seperti bagian koneksi database.

g. Object

Object merupakan keluaran dari class. Object dapat menampilkan atau mengelola isi class.

#### h. Null

Null mendeklarasikan variabel tanpa value. Sebuah variabel dapat menjadi null jika :

- a) Variabel belum diset ke nilai apapun
- b) Menerima unset()
- c) Dialokasikan konstanta NULL

### 2.1.4 Konstanta

Dalam bahasa pemrograman, Konstanta atau *constant* merupakan suatu lokasi penyimpanan yang berisikan nilai yang memiliki sifat tetap dan tidak dapat berubah sepanjang program berjalan.

Konstanta memiliki fungsi yang sama seperti variabel namun nilainya statis/konstan dan tidak bisa berubah.

#### Aturan Penulisan Konstanta PHP

##### 1. Cara Pendefinisikan Konstanta dalam PHP

Jika variabel pada PHP dibuat dengan menambahkan “tanda dollar”, seperti : \$nilai\_kuadrat. Untuk membuat konstanta PHP memiliki 2 cara :

- a. Menggunakan kata kunci (*Keyword*) const
- b. Menggunakan fungsi define

##### 2. Konstanta PHP bersifat case sensitive

Sama seperti variabel, konstanta dalam PHP bersifat case sensitif, sehingga perbedaan huruf besar dan kecil dianggap

berbeda. GAJI, GaJi, dan gaji merupakan 3 konstanta yang berbeda.

3. Nilai konstanta PHP tidak dapat diubah
4. Konstanta hanya dapat berisi tipe data tertentu

Konstanta dalam PHP hanya dapat berisi tipe data sederhana, yaitu :

- a. Boolean
- b. Integer
- c. Float
- d. String

Hal ini berbeda dengan variabel yang dapat berisi tipe data turunan seperti array, objek atau resources.

#### 5. Konstanta Sistem PHP (Predefined Constant)

Sama seperti variabel, PHP juga telah membuat beberapa konstanta yang telah didefinisikan dan tidak bisa di ubah nilainya. Namun karena banyaknya modul yang dapat ditambahkan kedalam PHP, Predefined Constant dalam PHP akan bertambah tergantung modul yang ada. Namun sebagai contoh, berikut adalah Predefined *Constant* dalam sistem inti PHP:

PHP\_VERSION, PHP\_MAJOR\_VERSION,  
PHP\_MINOR\_VERSION,  
PHP\_RELEASE\_VERSION,PHP\_VERSION\_ID,PHP\_EXT

RA\_VERSION,PHP\_ZTS, PHP\_DEBUG,  
 PHP\_MAXPATHLEN, PHP\_OS, PHP\_SAPI, PHP\_EOL,  
 PHP\_INT\_MAX, PHP\_INT\_SIZE,  
 DEFAULT\_INCLUDE\_PATH,PEAR\_INSTALL\_DIR,PEA  
 R\_EXTENSION\_DIR, PHP\_EXTENSION\_DIR,  
 PHP\_PREFIX, PHP\_BINDIR, PHP\_BINARY,  
 PHP\_MANDIR, PHP\_LIBDIR, PHP\_DATADIR, \_\_LINE\_\_ ,  
 \_\_FILE\_\_ , \_\_DIR\_\_ , \_\_FUNCTION\_\_ , \_\_CLASS\_\_ ,  
 \_\_TRAIT\_\_ , \_\_METHOD\_\_ , \_\_NAMESPACE\_\_ ,

### 2.1.5 Mengenal Operator

Operator merupakan simbol yang berfungsi untuk melakukan aksi/operasi tertentu pada nilai operand yang umumnya pada hasil operasi tersebut menghasilkan nilai baru. Operator terdiri dari 4 macam yaitu :

#### a. Operator Assignment

Operator ini untuk mengalokasikan nilai tertentu dengan tanda sama dengan (=). Seperti :

Operator	Contoh	Artinya
=	X=y	X=y
+ =	X+=y	X=x+y
- =	X-=y	X=x-y

$\text{*}=\text{}$	$\text{X}\text{*}=\text{y}$	$\text{X}=\text{x}\text{*}\text{y}$
$\text{/}=\text{}$	$\text{x}/=\text{y}$	$\text{X}=\text{x}/\text{y}$
$\text{.}=\text{}$	$\text{x.}=\text{y}$	$\text{X}=\text{x.y}$
$\text{\%}=\text{}$	$\text{X}\text{\%}=\text{y}$	$\text{X}=\text{x}\text{\%}\text{y}$

### **b. Operator Aritmeka**

Operator ini akan melakukan perhitungan matematika. Seperti :

Operator	Nama	Contoh	Hasil
+	Penambahan	$1 + 4$	5
-	Pengurangan	$1 - 4$	-3
/	Pembagian	$1 / 4$	0.25
*	Perkalian	$1 * 4$	4
%	Sisa pembagian (Modulus)	$5 \% 2$	1
++	Inkremen	$\text{X} = 5$ $\text{x}++$	$\text{X} = 6$

--	Dekremen	X = 5 X --	X = 4
----	----------	---------------	-------

### c. Operator Perbandingan

Operator perbandingan ini digunakan untuk menghasilkan 2 nilai yang hasil akhirnya merupakan nilai Boolean True atau False. Seperti :

Operator	Nama	Contoh	Hasil
==	Sama dengan	6 == 6	True
!=	Tidak sama dengan	3 != 3	False
>	Lebih besar	1 > 5	False
>=	Lebih besar sama dengan	3 >= 4	False
<	Lebih kecil	2 < 4	True
<=	Lebih kecil sama dengan	5 <= 4	False

#### d. Operator Logical

Operator logical ini berfungsi untuk mengoperasikan secara logical menggunakan and, or, atau not. Seperti :

Operator	Description	Example
$\&\&$	And	X = 5 y = 2 (x < 10 $\&\&$ y > 1) hasilnya true
$\ \ $	Or	X = 4 y = 6 (x == 5 $\ \ $ y == 1) hasilnya false
!	Not	X = 5 y = 3 !(x == y) hasilnya true

#### 2.1.6 Kelebihan PHP

Sebagai salahsatu bahasa pemrograman PHP memiliki banyak kelebihan diantaranya :

##### a. Komunitas yang besar

Komunitas PHP sangat besar dan tidak dapat dipungkiri bahwa komunitas ini tersebar diseluruh dunia. Tidak hanya di social media seperti Facebook bahkan komunitas PHP ini berada di Telegram, WhatsApp, dll.



b. Mudah dipelajari

PHP merupakan bahasa pemrograman semua orang yang hampir semua orang yang bergelut dengan dunia Web Development pernah menggunakannya atau mencobanya.

c. Resources yang melimpah

Dengan banyaknya komunitas yang besar maka akan berdampak pada kemudahan mencari resources yang berhubungan dengan PHP baik itu permasalahan yang terjadi di library, software, CMS hingga framework PHP.

d. Sempel

PHP memiliki syntax yang sangat sederhana dan mudah untuk dipelajari. Untuk memulai belajar PHP tidak perlu melakukan pengaturan apapun cukup hanya install XAMPP ataupun WAMP maka dapat memulai langsung belajar PHP.

e. *Maintenance* mudah

Sekali web yang menggunakan PHP berjalan, programmer dapat dengan mudah melakukan update dari software PHP dengan mudah jika memang diperlukan. karena sifat PHP yang merupakan interpreter. Aplikasi web yang dibuat dengan menggunakan PHP dapat dengan mudah diupgrade versi PHP tanpa harus melakukan kompilasi ulang source code. berbeda sekali dengan bahasa pemrograman lain yang membutuhkan kompilasi ulang jika melakukan upgrade versi dari bahasa

pemrograman. PHP juga dapat berjalan pada berbagai macam web server seperti apache, nginx, dan IIS.

*f. Open Source*

PHP merupakan sebuah projek *Open source* dengan *license* yang dikeluarkan oleh PHP grup yaitu PHP license V3.01. Inti dari license ini adalah setiap pengguna program PHP bebas menggunakan PHP secara gratis tanpa harus memberikan royalty apapun ke PHP group namun tetap wajib mencantumkan licensi atas PHP yang dimiliki PHP Group. Dengan kata lain selama pemakai program PHP tidak mengakui produk PHP adalah buatannya maka perjual belian program yang menggunakan PHP diperbolehkan tanpa harus membayar licensi apapun.

*g. Perkembangan pesat*

Karena sifat PHP yang open source, banyak sekali bermunculan projek projek open source besar yang menggunakan PHP seperti Prestashop, WordPress, Drupal, dan lain lain. Hal ini menjadi keunggulan yang sangat besar bagi orang yang menguasai pemrograman PHP. Dengan sangat luasnya perkembangan PHP, maka kesempatan untuk bisnis ataupun kerja pada bidang pemrograman PHP sangatlah luas

### 2.1.7 Kekurangan PHP

Kekurangan utama PHP yaitu PHP merupakan bahasa yang weak type dimana variabel tidak memiliki tipe data sehingga menyulitkan ketika melakukan debugging. Pada weak type ini menyebabkan terjadinya *juggling* dimana variable yang terjadi berisi nilai integer misalnya dapat berubah menjadi nilai string atau tipe data lainnya.

Berikut merupakan kekurangan dalam bahasa pemrograman PHP :

#### a. Banyak kompetisi

Komunitas yang banyak tentu membawa kompetisi yang ketat. Para web developer yang menguasai PHP tiap hari semakin bertambah. Namun kekurangan ini seharusnya menjadi pemacu bagi para pebisnis yang menginginkan produk IT untuk menggunakan PHP sebagai bahasa pemrograman yang digunakan untuk mengembangkan aplikasi bisnisnya karena terbukanya para programmer PHP yang sangat kompetitif dan tiap hari semakin banyak

#### b. Mudah di bajak

Karena sifat PHP yang merupakan interpreter, source code dari aplikasi php dapat dengan mudah di modifikasi dan diubah fungsinya. hal ini membuat PHP tidak cocok untuk digunakan mengembangkan aplikasi jika pemilik aplikasi memiliki source code yang ingin dijaga kerahasiaannya.

Meskipun ada cara untuk mengamankan source code yang menggunakan bahasa pemrograman PHP, namun dibutuhkan sebuah ekstensi yang berbayar yang dikeluarkan oleh Zend sebuah korporasi di bidang pemrograman PHP.

c. Terkesan kurang prestigious

Entry level yang berada pada tingkat pemula, yakni mudah dipelajari oleh programmer pemula membuat bahasa pemrograman PHP terkesan kurang prestisius jika dibandingkan dengan bahasa pemrograman web lain yang terkesan lebih sulit untuk digunakan. Aplikasi web yang dihasilkan dari penggunaan bahasa pemrograman PHP terkesan kurang aman dan memiliki celah. Namun sebenarnya hal ini disebabkan karena faktor pengembang yang mungkin belum mempelajari secara penuh bagaimana standar dan cara membuat aplikasi yang benar dengan menggunakan PHP. Dari sisi performa, pemrograman PHP dan pemrograman lainnya jika digunakan dengan standar dan penerapan yang benar, akan menghasilkan sebuah aplikasi web yang berkualitas.

d. Tidak adanya type data pada PHP

PHP tidak memiliki type data. hal ini membuat kadang pada saat menggunakan bahasa pemrograman PHP muncul bug bug yang tidak diinginkan berkaitan dengan tidak adanya standar type data pada php. contohnya adalah data “1000” dan “1e3” jika dibandingkan akan memiliki type data yang sama karena secara implisit data tersebut dirubah menjadi floating point. namun kekurangan ini berkaitan sekali dengan pengalaman

dari developer yang menggunakan bahasa pemrograman PHP. developer yang sudah ahli tentunya sudah paham betul bagaimana mengatasi permasalahan ini.

#### **2.1.8 Sejarah PHP (Hypertext Pre Processor)**

Pertama kali PHP dikenal dengan kependekan dari (*Personal Home Page*) atau Situs Personal yang dibuat oleh Rasmus Lerdorf pada Tahun 1995. Saat pertama kali dibuat PHP masih bernama *Form Interpreted* (FI) yang berupa sekumpulan skrip untuk pengolahan data formulir dari web.



Gambar Logo PHP Pertama

Kemudian Rasmus merilis kode sumber tersebut pada bulan November tahun 1997 untuk umum dengan memberikan nama PHP/FI. Dengan merilis kode tersebut menjadi kode sumber terbuka sehingga banyak pemrograman yang tertarik untuk mengembangkan PHP tersebut.

Di tahun yang sama, sebuah perusahaan bernama Zend menulis kembali interpreter PHP menjadi lebih cepat, rapih dan

lebih baik. Sehingga pada bulan Juni tahun 1998, perusahaan tersebut merilis kembali interpreter baru untuk PHP dan meresmikan perilisan tersebut dengan nama PHP 3.0 serta mengubah nama menjadi akronim berulang yang diberi nama PHP (*Hypertext Preprocessing*).

Pada pertengahan tahun 1999, perusahaan Zend merilis kembali interpreter PHP dengan nama PHP 4.0 pada versi 4.0 ini banyak digunakan pada awal abad ke-21. Karena versi ini memiliki kemampuan untuk membangun aplikasi web kompleks tetapi tetap memiliki stabilitas dan kecepatan yang tinggi. Setelah 5 tahun berlalu pada akhirnya Zend merilis kembali PHP dengan versi 5.0 pada bulan Juni tahun 2004. Pada versi ini, PHP memasukkan model pemrograman berorientasi objek untuk mengembangkan bahasa pemograman ke arah paradigm berorientasi objek.



Gambar logo PHP 5

Pada tanggal 18 Desember pada tahun 2014 Zend merilis bahasa pemrograman PHP menjadi versi 5.6.4 yang diperuntukan agar mempermudah pengembang menjalankan kode PHP tanpa menginstall *software server* terlebih dahulu. Tidak lama kemudian pada tanggal 17 Februari 2017 versi 5.6.4 di kembangkan kembali menjadi versi terbaru dan stabil pada bahasa pemrograman PHP saat ini menjadi versi 7.0.16 dan 7.1.2.



Gambar Logo PHP 7

## **2.2 Data**

Data merupakan fakta atau observasi mental yang biasanya mengenai fenomena fisik atau transaksi bisnis. Data pun merupakan suatu ukuran objektif dari atribut (karakteristik) dari entitas. Dalam pemrograman tempat penyimpanan data disebut juga dengan basis data.

Basis data merupakan tempat untuk menyimpan data yang diperlukan pada web.

### **2.2.1 Sejarah Basis Data**

Basis data merupakan suatu focus utama pada aplikasi, pada awal tahun 1960 Charles Bachman di perusahaan General electric mendesain generasi pertama DBMS yang disebut dengan penyimpanan data terintegrasi (Integrated Data Store) karena hampir semua data computer disimpan pada pita magnetic, namun karena pita hanya dapat di proses secara berurut, maka data harus disimpan dalam bentuk daftar (atau biasa disebut file sekuensial.

*Database* terdiri dari kelompok table-tabel yang berhubungan. Pada kebanyakan kasus, masing-masing table berisi data dengan satu tema. Hubungan antara table-tabel direpresentasikan dengan cara memberikan masing-masing baris ID yang unik dan menggunakan ID tersebut untuk menghubungkan sebuah baris dalam satu table dengan sebuah baris dalam table baris kedua. Key pada database merupakan kolom yang mengidentifikasi baris yang unik. Kemudian



foreign key merupakan sebuah kolom dalam satu table yang merupakan identifier unik pada table kedua. Foreign key digunakan untuk memperlihatkan hubungan diantara table. Table dapat dikombinasikan dengan menggunakan SQL yang merupakan bahasa standar industry untuk pemrosesan table.

Sebuah sistem database terdiri dari empat unsur dasar yaitu :

- a. User
- b. Aplikasi database
- c. DBMS
- d. Database

User menggunakan sistem database untuk melakukan pekerjaannya. Aplikasi hasilkan form, queries dan report, mengeksekusi logika aplikasi dan mengendalikan pemrosesan DBMS menciptakan, memproses, dan mengadministrasikan database.

Sebuah database adalah sekumpulan records terintegrasi yang menggambarkan dirinya sendiri. Database terdiri dari data user, metadata, indeks, prosedur tersimpan (*stored procedur*), trigger, dan metadata aplikasi database.

Trigger merupakan prosedur yang aktif saat tindakan tertentu terjadi. Teknologi database dapat digunakan dalam aplikasi yang luas. Sebagian database digunakan oleh satu individu, sementara sebagian lagi digunakan oleh

workgroup, dan yang lainnya digunakan oleh organisasi besar.

Seperti sistem informasi, sistem database dapat dikembangkan dengan menggunakan sebuah proses yang mencakup tiga fase, yaitu :

- a. Requirement
- b. Desain
- c. Implementasi

Selama fase requirement, sebuah model data atau representasi logis dari struktur database, dikembangkan.

Model data merupakan bagian penting karena desain database dan aplikasinya diturunkan dari model data.

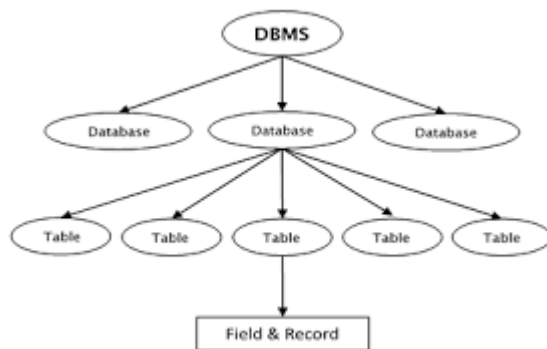
Diagram entity-relationship merupakan sebuah tool yang digunakan untuk merepresentasikan model data.

Model data ditransformasikan menjadi table dan hubungan selama fase desain. Indeks, batasan, prosedur tersimpan, dan trigger juga dirancang dalam fase ini. Diagram struktur data kadang-kadang digunakan untuk mendokumentasikan table dan hubungannya.

- a. Model *Database* Hirarki

Model Database Pertama, dengan suksesnya media penyimpanan disk pada tahun 1960an, maka kita dapat memiliki akses non-sekuensial atau langsung ke records (Perekaman). Dalam hal ini, database dirancang untuk menghilangkan

masalah pemrosesan file sekuensial. Terdapat dua arsitektur atau model yang awalnya sukses. IBM mengembangkan dan mempromosikan DL/I atau Data *language one* yang membuat model data database dalam bentuk pohon hirarki.

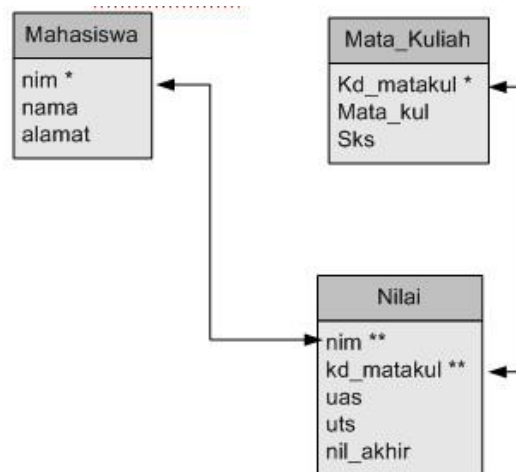


Gambar DBMS Hirarki

#### b. Model *Database* Relasional

Model database relasional pertama kali diajukan oleh E.F. Codd pada tahun 1970. Codd bekerja untuk IBM dan setelah 10 tahun melakukan penelitian, pengembangan dan lobi, dia dan temannya yang lain berhasil meyakinkan IBM untuk mengembangkan produk DBMS berdasarkan model relasional, model yang paling terkenal dalam produk ini yaitu DB2, yaitu DBMS yang masih aktif digunakan hingga saat ini.

Kemudian perusahaan seperti Oracle, Ingres, Sybase dan Informix mengembangkan juga produk DBMS berdasarkan model relasional.



Gambar Database Relasional

Dasar untuk model data jaringan terbentuk lalu di standarisasi oleh CODASYL (Conference on Data System Languages). Pada tahun 1973 Bachman menerima penghargaan CM Turing Award yaitu nobel pada ilmu *computer*.

Menurut sejarah, system pemrosesan basis data terbentuk setelah masa system pemrosesan manual dan system pemrosesan berkas. System pemrosesan manual (Berbasis kertas) merupakan bentuk pemrosesan yang menggunakan dasar berupa setumpuk rekaman yang disimpan pada rak-rak berkas. Jika berkas-berkas tersebut diperlukan, berkas

tersebut harus dicari pada rak-rak tersebut. System pemrosesan berkas merupakan system komputer, dimana sekelompok rekaman disimpan pada sejumlah berkas secara terpisah. Perancangan system ini didasarkan pada kebutuhan individual pengguna, bukan kebutuhan sejumlah pengguna. Sehingga setiap aplikasi menuliskan data sendiri, alhasil ada kemungkinan data yang sama terdapat pada berkas-berkas lain yang digunakan oleh program aplikasi lain.

Sejarah DBMS (*Database Management System*), generasi pertama DBMS didesain oleh Charles Bachman di perusahaan General Electric pada awal tahun 1960, disebut sebagai penyimpanan data terintegrasi (*Integrated Data Store*). Dibentuk dasar untuk model data jaringan yang kemudian distandardisasi oleh CODASYL (*Conference on Data System Languages*).

Pada akhir 1960, IBM mengembangkan system manajemen informasi (*Information Management System*) DBMS. IMS dibentuk dari representasi data pada kerangka kerja yang disebut dengan model data hirarki. Dalam waktu yang sama, dikembangkan system SABRE sebagai hasil kerjasama antara IBM dengan perusahaan penerbangan Amerika. *System* ini memungkinkan user untuk mengakses data yang sama pada jaringan komputer.

Pada tahun 1985, Microsoft dan IBM mengumumkan perjanjian kerjasama jangka panjang untuk mengembangkan system operasi dan produk-produk perangkat lunak lainnya. Pengumuman ini adalah permulaan dimulainya OS/2, sebuah *system* operasi setelah masa kejayaan MS-DOS. OS/2 ini akan lebih hebat dan lebih canggih daripada MS-DOS, ia akan mampu menangani multitasking application dengan memanfaatkan kemampuan processor Intel yang terbaru yaitu 80286. OS/2 secara resmi diumumkan pada bulan april 1987, dan dijanjikan akan tersedia buat end-user pada akhir tahun tersebut.

Tetapi dalam waktu yang sangat singkat setelah pengumuman perjanjian kerjasama tersebut, IBM mengumumkan peluncuran sebuah versi spesial OS/2 yang disebut OS/2 Extended Edition. Versi lebih powerfull ini akan menyertakan sebuah Database SQL yang disebut OS/2 Database Manager, OS/2 Database Manager akan sangat berguna bagi pengembangan aplikasi yang sederhana dan kompatibel dengan DB/2 , sebuah Database server milik IBM yang beroperasi pada mainframe. OS/2 Database manager juga akan menyertakan SNA (System Network Architecture) communication service, yang disebut OS/2 Communication Manager. Sebagai bagian dari SSA (System Application

Architecture)-nya. IBM menjanjikan semua produk-produk tersebut dapat saling bekerjasama pada masa yang akan datang.

Hal inilah yang membuat Microsoft segera mencari solusi sendiri. Pada tahun 1986, Microsoft sudah menghasilkan 197 JutaUS\$ per tahun, dengan 1153 pegawai. (Sepuluh tahun kemudian, Microsoft telah mendapatkan 6 Milyar US\$ dari bisnis softwarenya, dengan hampir sekitar 18.000 pegawai). Produk-produk Microsoft hampir semuanya terfokus pada aplikasi desktop dengan produk utamanya adalah MS-DOS. Komputasi Client/Server pada saat itu belum menjadi fokus utama Microsoft dan industri komputer. Manajemen data pada sebuah PC hanyalah menjadi sebuah fantasi pada masa itu, User biasanya hanya menggunakan LOTUS 1-2-3 untuk menyimpan data. Produk dBASE buatan Ashton Tate segera menjadi sangat populer setelah diluncurkan, Selanjutnya Anca Software merilis Paradox dan Micro Rim dengan prouk RBase-nya. Pada tahun 1986 tersebut, Microsoft belum mempunyai produk manajemen Database sendiri. (Tetapi pada tahun 1992, Microsoft mendapatkan sukses yang luar biasa dari produk manajemen database desktop-nya dengan Microsoft Access dan Microsoft FoxPro).

Tetapi IBM *Database Manager* sangatlah berbeda dari dBASE, Paradox atau RBase. Produk IBM ini walaupun tidak begitu user-friendly, tetapi ia mempunyai SQL query

processor-nya sendiri dan bekerja berdasarkan transactions, hampir sama dengan Database Server yang berjalan diatas Minicomputer dan mainframe, seperti DB/2, Oracle ataupun Informix. Microsoft membutuhkan produk DBMS (Database Management System) sekaliber ini dan membutuhkannya segera!.

Microsoft berpaling kepada Sybase, Inc. Sebuah perusahaan pembuat software DBMS yang sedang 'Naik daun', yang merilis produk Data Server-nya pada bulan mei 1987 untuk Sun Microsystem yang berjalan diatas UNIX. Data Server mendapatkan reputasi berkat inovasi dan kreatifitasnya dengan Stored Procedure dan Trigger serta paradigma baru dalam dunia komputasi yaitu: Client/Server.

Basis data merupakan kumpulan dari elemen data logis yang saling berhubungan. Basis data mengonsolidasi banyak catatan yang sebelumnya disimpan dalam file terpisah. Basis data pun merupakan suatu kumpulan data yang berhubungan secara logis dan deskripsi data tersebut, yang dirancang untuk memenuhi informasi yang dibutuhkan oleh organisasi. Maka basis data merupakan tempat penyimpanan data yang besar dimana dapat digunakan oleh banyak pengguna. Seluruh item basis data tidak dimiliki oleh satu departemen, melainkan menjadi sumber daya perusahaan yang dapat digunakan bersama.



### 2.2.2 *Object Oriented DBMS (OODBMS)*

Pemrograman berorientasi objek (*object oriented*) mulai digunakan pada tahun 1980 dan akan dikembangkan menjadi produk DBMS berorientasi objek. Tujuan produk ini untuk menyimpan objek pemrograman berorientasi objek seperti C++ atau java, dalam sebuah database tanpa harus mentransformasikannya ke format relasional.

### 2.2.3 *Macam-macam Database*

Untuk data yang sangat besar dan kompleks, dapat menggunakan Oracle, IBM DB2, PostgreSQL. Sedangkan untuk alasan kesederhanaan dan kecepatan namun masih mendukung data yang relative besar, MySQL merupakan *database* yang banyak digunakan.

Pada sistem ini penulis menggunakan *database* MySQL untuk mendukung antara *database* yang digunakan dengan bahasa pemrograman PHP maka disarankan untuk menggunakan sistem XAMPP.

### 2.2.4 MySQL

MySQL merupakan sebuah perangkat lunak yang mengatur basis data relasional (RDBMS) yang di distribusikan gratis dengan lisensi GPL (General Public License). Pengguna dapat menggunakan MySQL secara bebas namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan bersifat komersial.

MySQL adalah server yang melayani database. Untuk membuat dan mengolah database, pada dMySQL ini dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL. Database diperlukan jika akan menginput data

dari user menggunakan form HTML untuk diolah PHP agar dapat disimpan kedalam database MySQL.

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya. SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional.



Gambar Logo MySQL

#### 2.2.4.1 Sejarah MySQL

MySQL pada awalnya diciptakan pada tahun 1979, oleh Michael "Monty" Widenius, seorang programmer komputer asal Swedia. Monty mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi low-level ISAM database engine dengan indexing. Pada saat itu Monty bekerja pada perusahaan bernama TcX di Swedia. TcX pada tahun 1994 mulai mengembangkan aplikasi berbasis web, dan berencana menggunakan UNIREG sebagai sistem database. Namun sayangnya, UNIREG dianggap tidak cocok untuk database yang dinamis seperti web. TcX kemudian mencoba mencari alternatif sistem database lainnya, salah satunya adalah mSQL (miniSQL). Namun mSQL versi 1 ini juga memiliki kekurangan, yaitu tidak mendukung indexing, sehingga performanya tidak terlalu bagus.

Dengan tujuan memperbaiki performa mSQL, Monty mencoba menghubungi David Hughes (programmer yang mengembangkan mSQL) untuk menanyakan apakah ia tertarik mengembangkan sebuah konektor di mSQL yang dapat dihubungkan dengan UNIREG ISAM sehingga mendukung indexing. Namun saat itu Hughes menolak, dengan alasan sedang mengembangkan teknologi indexing yang independen untuk mSQL versi 2.

Dikarenakan penolakan tersebut, David Hughes, TcX (dan juga Monty) akhirnya memutuskan untuk merancang dan mengembangkan sendiri konsep sistem database baru. Sistem ini merupakan gabungan dari UNIREG dan mSQL (yang source codenya dapat bebas digunakan). Sehingga pada May 1995, sebuah RDBMS baru, yang dinamakan MySQL dirilis. David Axmark dari Detron HB, rekanan TcX mengusulkan agar MySQL di 'jual' dengan model bisnis baru. Ia mengusulkan agar MySQL dikembangkan dan dirilis dengan gratis. Pendapatan perusahaan selanjutnya di dapat dari menjual jasa "support" untuk perusahaan yang ingin mengimplementasikan MySQL. Konsep bisnis ini sekarang dikenal dengan istilah Open Source.

Pada tahun 1995 itu juga, TcX berubah nama menjadi MySQL AB, dengan Michael Widenius, David Axmark dan Allan Larsson sebagai pendirinya. Titel "AB" di belakang MySQL,

adalah singkatan dari “Aktiebolag”, istilah PT (Perseroan Terbatas) bagi perusahaan Swedia.

Keistimewaan MySQL :

MySQL memiliki beberapa keistimewaan, antara lain:

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Perangkat lunak sumber terbuka. MySQL didistribusikan sebagai perangkat lunak sumber terbuka, di bawah lisensi GPL sehingga dapat digunakan secara gratis.
3. Multi-user. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. 'Performance tuning', MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data. MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

6. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (query).
7. Keamanan. MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix socket (UNIX), atau Named Pipes (NT).
10. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
11. Antar Muka. MySQL memiliki antar muka (interface) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).

12. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (tool) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
13. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.

#### 2.2.4.2 Kelebihan MySQL

Kelebihan MySQL dibandingkan dengan RDBMS lainnya :

1. Berlisensi GPL dan Multi Platform.
2. Dapat diintegrasikan dengan beberapa bahasa Pemrograman seperti .Net, Java, Python, Perl yang merupakan bahasa pemrograman yang paling dominan di kalangan programmer.
3. Mendukung ODBC untuk sistem operasi Windows sehingga bisa digunakan aplikasi yang berjalan di windows.
4. Bisa dijalankan pada spesifikasi hardware yang rendah karena lebih hemat resource memory (dibandingkan database lain) sehingga mudah digunakan untuk bahan pembelajaran.

5. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari 20 bahasa meskipun bahasa indonesia belum termasuk didalamnya.
6. MySQL dapat diintegrasikan dengan Hosting.

#### 2.2.4.3 Kekurangan MySQL

Kekurangan MySQL dengan RDBMS lainnya :

1. Banyak mengklaim kurang support terhadap pemrograman Visual/Desktop, sehingga sedikit yang menggunakan untuk aplikasi visual.
2. Karena berlisensi GPL sehingga sulit mendapatkan update untuk problem yang urgent, sehingga perusahaan skala menengah keatas lebih memilih RDBMS berlisensi dan disupport seperti Oracle dan MS SQL Server.
3. Sangat diragukan dalam menangani data skala besar, karena ada beberapa opini yang pro dan kontra terhadap kemampuan MySQL terhadap pengolahan data yang besar.

#### 2.2.4.4 Kemampuan MySQL

Berikut merupakan fitur yang terdapat pada MySQL :

1. Unjuk kerja yang tinggi dalam memproses query sederhana, dalam arti dapat memproses lebih banyak SQL per satuan waktu.
2. Memiliki lebih banyak tipe data seperti: signed/unsigned integer yang memiliki panjang data



sebesar 1,2,3,4 dan 8 byte, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET dan tipe ENUM.

3. Mendukung field yang dijadikan Index, dengan maksimal 32 index dalam satu tabel.
4. MYSQL memiliki beberapa lapisan keamanan, seperti subnetmask, nama host, dan izin akses user dengan sistem perijinan yang mendetail serta sandi/password terenkripsi.
5. Konektivitas , MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix socket (UNIX), atau Named Pipes (NT).
6. Multi-user. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik
7. Command and function, MySQL memiliki fungsi dan operator secara penuh yang mendukung perintah select dan where dalam query.
8. Structure Table, MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE dibandingkan DBMS lainnya.
9. Mendukung penuh terhadap kalimat SQL GROUP BY dan ORDER BY. Mendukung terhadap fungsi penuh (

COUNT(),COUNT(), DISTINCT() AVG(), STD(),  
SUM(), MAX() dan MIN() )

#### 2.2.4.5 Bahasa Pemrograman MySQL

Terdapat beberapa API (Application Programming Interface) tersedia yang memungkinkan aplikasi-aplikasi komputer yang ditulis dalam berbagai bahasa pemrograman untuk dapat mengakses basis data MySQL antara lain: bahasa pemrograman C, C++, C#, bahasa pemrograman Eiffel, bahasa pemrograman Smalltalk, bahasa pemrograman Java, bahasa pemrograman Lisp, Perl, PHP, bahasa pemrograman Python, Ruby, REALbasic dan Tcl. Sebuah antarmuka ODBC memanggil MyODBC yang memungkinkan setiap bahasa pemrograman yang mendukung ODBC untuk berkomunikasi dengan basis data MySQL. Kebanyakan kode sumber MySQL dalam ANSI C.

### 2.2.5 XAMPP



Gambar Logo XAMPP

XAMPP merupakan server yang banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi programmer pemula. Selain gratis, fiturnya tergolong lengkap dan gampang sehingga banyak digunakan oleh programmer tingkat awal.

Xampp juga merupakan salah satu aplikasi web server apache yang terintegrasi dengan mysql dan phpmyadmin. XAMPP adalah singkatan dari X, Apache Server, MySQL, PHPMyadmin, dan Python. Huruf X di depan menandakan XAMPP dapat diinstal pada berbagai operating system. XAMPP dapat diinstal pada Windows, Linux, MacOS, dan Solaris.

Sampai saat ini, XAMPP masih didapatkan secara gratis, bebas di download dan digunakan tanpa harus membayar.

Xampp merupakan buatan dari *team* apache friends.  
Xampp dapat di download langsung dari situs resminya.  
Xampp 1.8.3 memiliki beberapa fitur seperti :

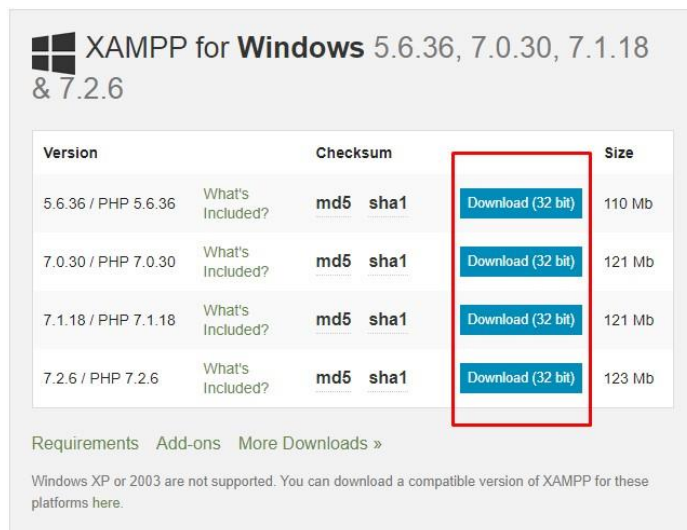
- a. Apache 2.4.6
- b. MySQL 5.6.14
- c. PHP 5.5.6
- d. phpMyAdmin 4.0.9
- e. FileZilla FTP Server 0.9.41
- f. Tomcat 7.0.42
- g. Strawberry Perl 5.16.3.1 Portable
- h. XAMPP Control Panel 3.2.1

#### **2.2.5.1 *Instalasi XAMPP***

Untuk menginstall XAMPP ini dapat di download installer XAMPP pada situs XAMPP. Isi dari aplikasi XAMPP ini pun sudah sangat kumplit.

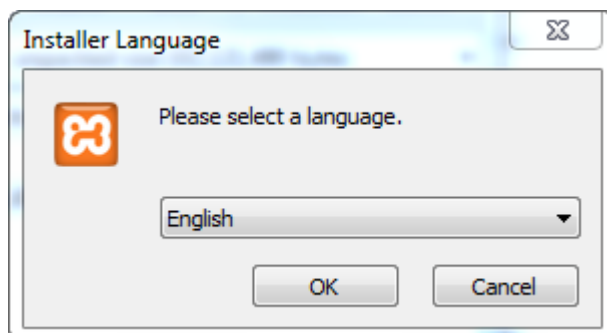
Saat pembuatan sistem ini saya menggunakan windows maka panduan install pun diperuntukan untuk pengguna PC windows.

- a. Download aplikasi XAMPP terbaru, pilih salah satu versi, jika anda bingung pilihlah versi 5.6 dahulu



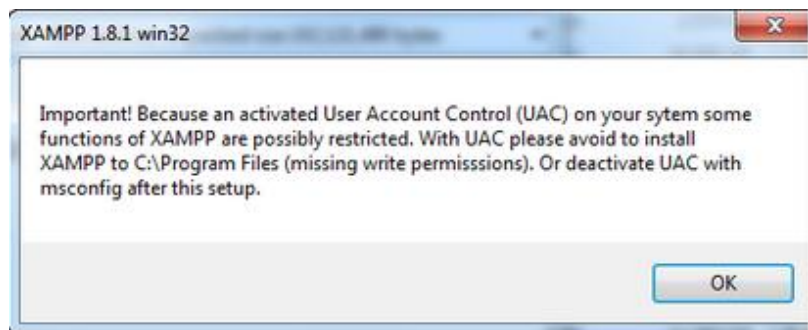
Gambar download XAMPP

- b. Setelah download maka double klik file XAMPP , selanjutnya akan muncul jendela installer language seperti dibawah :

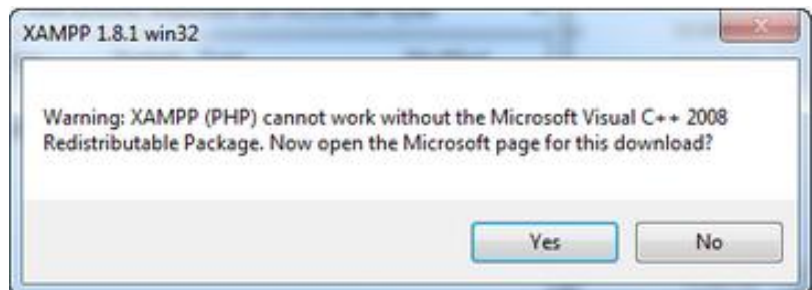


Gambar *installer language*

- c. Selanjutnya pilih bahasa. Pilih Bahasa Inggris (English). Klik OK
- d. Terkadang akan muncul pesan error. Jika ada, biarkan saja dan lanjutkan dengan klik OK dan Yes.



Gambar XAMPP Error



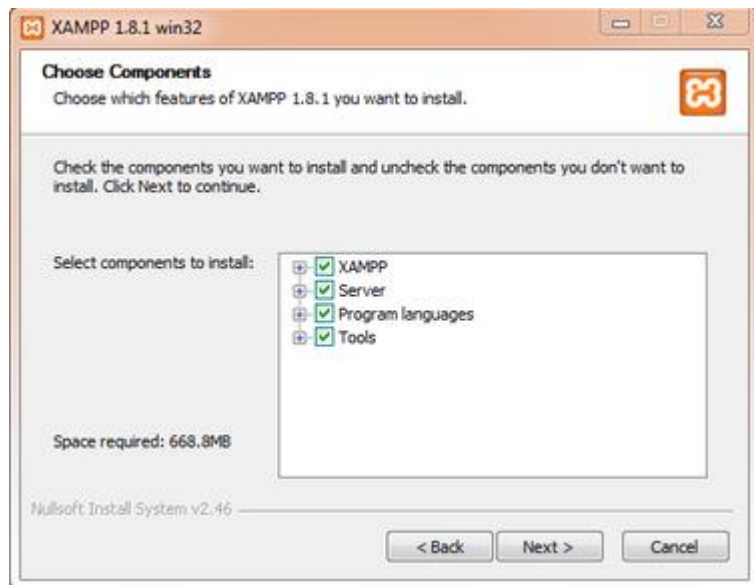
Gambar Warning XAMPP

- e. Berikutnya akan muncul jendela yang berisi permintaan untuk menutup semua aplikasi yang sedang berjalan. Jika semua aplikasi sudah ditutup maka klik tombol **Next**.



Gambar Memulai instalasi

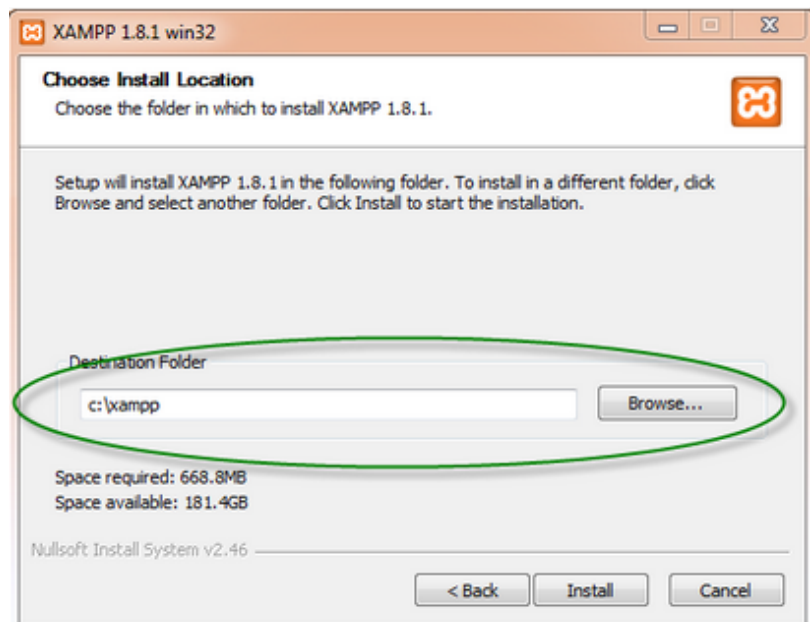
- f. Selanjutnya akan ada permintaan untuk memilih aplikasi yang mau diinstal, centang semuanya dan klik tombol **Next**.



Gambar centang aplikasi

- g. Kemudian Anda akan diminta untuk menentukan lokasi folder penyimpanan file-file dan folder XAMPP. Secara default akan diarahkan ke lokasi **c:\xampp**. Namun jika Anda ingin menyimpannya di folder lain bisa klik **browse** dan tentukan secara manual folder yang ingin digunakan. Jika sudah selesai, lanjutkan dan klik tombol **Install**.





Gambar penyimpanan direktori

- h. Tunggu beberapa menit hingga proses instalasi selesai. Jika sudah muncul jendela seperti di bawah ini, klik tombol **Finish** untuk menyelesaikannya.



Gambar selesai install

- i. Berikutnya, akan muncul jendela dialog seperti gambar di bawah ini yang menanyakan Anda apakah mau langsung menjalankan aplikasi XAMPP atau tidak. Jika ya, maka klik YES.



Gambar akhir Instalasi

#### 2.2.5.2 Cara Menjalankan Aplikasi XAMPP

- a. Bukalah aplikasi XAMPP, bisa melalui Start Menu atau Desktop, dan klik icon XAMPP. Atau, jika Anda membukanya begitu proses instalasi selesai maka klik **Yes** seperti yang terlihat pada gambar di atas.

- b. Setelah terbuka, silahkan klik tombol **Start** pada kolom **Action** sehingga tombol tersebut berubah menjadi **Stop**. Dengan mengklik tombol tersebut, artinya itulah aplikasi yang dijalankan. Biasanya jika saya menggunakan XAMPP, yang saya start hanyalah aplikasi Apache dan MySQL, karena saya tidak memerlukan aplikasi seperti Filezilla, dan lain-lain.



Gambar menjalankan XAMPP

- c. Sekarang bukalah browser kesukaan Anda, dan coba ketikkan **http://localhost/xampp** di address bar. Jika muncul tampilan seperti gambar di bawah ini, instalasi telah berhasil.



Gambar tampilan web XAMPP

- d. Saat ini anda berhasil menginstal aplikasi XAMPP. Sekarang komputer Anda sudah berfungsi seperti server dan bisa menjalankan aplikasi-aplikasi berbasis web.

### 2.3 Bootstrap

Dalam membuat suatu sistem maka diperlukan nya suatu CSS atau *Bootstrap* untuk memperindah penampilan sistem tersebut. *Bootstrap* merupakan framework front-end yang intuitif dan powerful untuk pengembangan aplikasi web yang lebih cepat dan mudah. *Bootstrap* menggunakan HTML,

CSS, dan Javascript. *Bootstrap* dikembangkan oleh Mark Otto dan Jacob.

*Bootstrap* merupakan proyek *open source* maka yang pasti pengguna *bootstrap* setiap pemakaiannya gratis. Saat ini, *bootstrap* menjadi aplikasi paling populer di situs layanan *hosting software open source* di Github.com.

Saat ini banyak websiyang menggunakan *Bootstrap* sebagai *framework desainnya*.

### 2.3.1 Sejarah *Bootstrap*

Pada awalnya bootstrap bernama *Twitter Blueprint*. Tool ini mulai dikembangkan pada tahun 2010 oleh tim desainer dan pengembang di Twitter, yaitu Mark Otto dan Jacob Thornton.

Setelah digunakan sebagai framework internal di situs Twitter, setahun kemudian tool ini dirilis ke public sebagai proyek open source menggunakan nama Bootstrap.

Tahapan pengembangan bootstrap, yaitu :

- a. Bootstrap dirilis Agustus 2011
- b. Januari 2012 bootstrap 2 dirilis kembali dengan memperkenalkan Layout dan Responsive Design. Grid Layout merupakan sistem tata kolom, adapun Responsive Design merupakan sisten desain web yang mobile friendly yaitu versi bootstrap 2.3.2.
- c. Agustus 2013, Bootstrap 3 dirilis dengan memperkenalkan pendekatan *Mobile First* dan *Flat Design*. Mobile First merupakan pendekatan desain web yang menggunakan ukuran layar ponsel sebagai patokan dasar desain. Flat design merupakan pendekatan desain web yang minimalis, simple, dan

datar tanpa banyak efek drop shadow, gradasi warna, atau gambar tekstur, versi ini merupakan bootstrap versi 3.3.5.

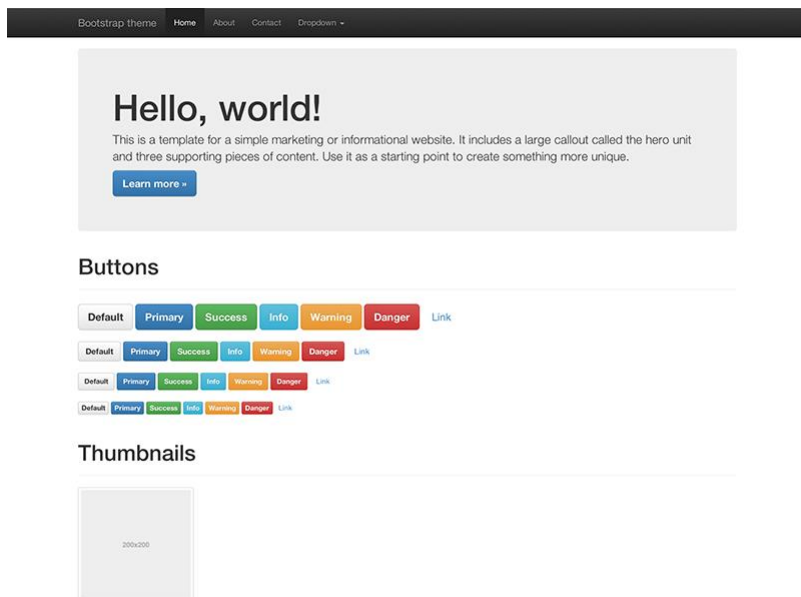
- d. April 2015 bootstrap 4 dikembangkan dan akhirnya Agustus 2015 merilis versi alpha (uji coba awal). Versi stabilnya direncanakan akan dirilis setelah melewati dua versi beta dan satu versi RC (*release candidate*) yang akan datang.

### 2.3.2 Fitur-fitur Bootstrap

Dalam bootstrap ini memiliki fitur-fitur, kelebihan dan manfaat dari implementasi bootstrap pada proyek website atau aplikasi web.

#### 2.3.2.1 Komponen Siap Pakai

Bootstrap telah menyediakan desain beragam komponen tampilan web yang siap dipakai.



Gambar Bootstrap Awal

### 2.3.2.2 Kompabilitas Web Browser

Kode bootstrap kompatibel dengan beragam web browser modern seperti Mozilla Firefox, Gogle Chrome, hingga Internet Explorer versi terbaru.

Untuk mengantisipasi permasalahan pada browser terdahulu yang belum mensupport teknologi saat ini seperti HTML5 dan CS3, maka bootstrap telah menyisipkan kode Normalize.css ke dalam file CSS-nya.

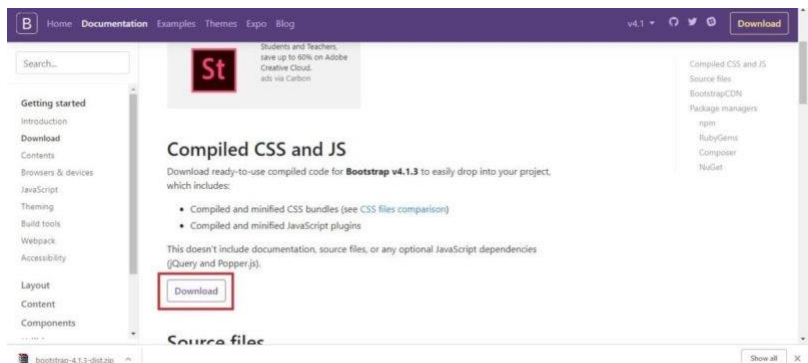
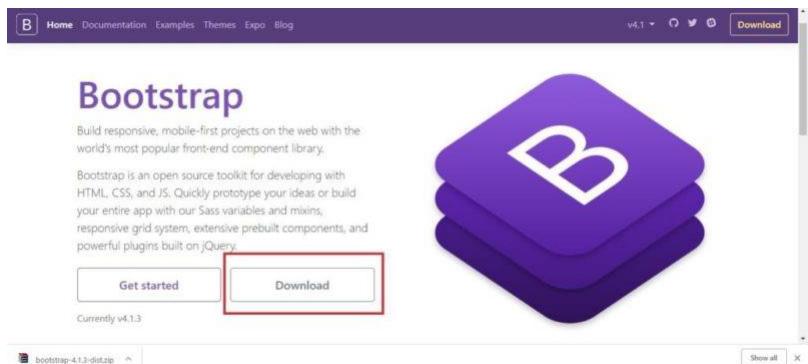
Normalize.css merupakan file CSS populer untuk mereset seluruh elemen HTML, sehingga browser terdahulu dapat merendernya seperti pada browser moder.

### 2.3.3 Cara Install Bootstrap

#### 2.3.3.1 Install Bootstrap Offline

Untuk sistem aplikasi pemilihan karyawan ini penulis menggunakan bootstrap sebagai framework tampilan web, berikut ini merupakan cara install bootstrap yang penulis lakukan :

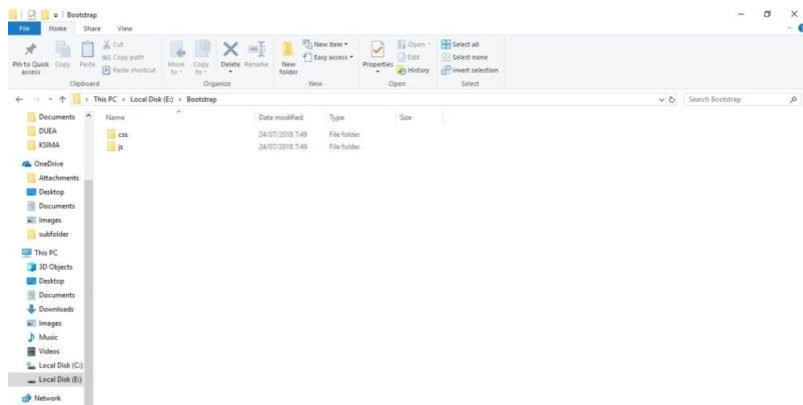
- a. Untuk menginstall boot strap secara offline, langkah pertama yang perlu anda lakukan adalah mendownload file bootsrap melalui website <https://getbootstrap.com>.



Gambar Download Bootstrap

- b. Kemudian, pilih tombol download untuk mendownload file bootstrap. Nah, setelah mendownload file tersebut dalam bentuk zip. Ekstrak file tersebut dalam folder baru. Nantinya, kalian akan menemukan seperti ini.





Gambar Bootstrap setelah di Extract

- c. Kemudian, langkah terakhir adalah kita perlu membuat file .html. Buka text editor kalian, kemudian paste kode dibawah ini.

```
<!Doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta                                name="viewport"
content="width=device-width,          initial-
scale=1">
    <title>Belajar Bootstrap</title>
    <link
href="bootstrap/css/bootstrap.min.css"
rel="stylesheet">
    <script                                src="bootstrap/js/jquery-
3.2.1.js"></script>
```

```

<script
src="bootstrap/js/bootstrap.min.js"></script>
</head>
<body>
<h1> Instal Bootstrap Berhasil</h1>
</html>

```

- d. Jangan lupa file tersebut disimpan dalam bentuk html yaa. Jadi nantinya, di folder boot strap minimal ada 2 folder dan 1 file yaitu file css, js dan file html

### 2.3.3.2 Cara Install Bootstrap Online

Jika install bootstrap secara offline kita perlu mendownload terlebih dahulu. Kalau, secara online kita tidak perlu mendownloadnya. Cukup hanya dengan menuliskan alamat server yang menyimpan library boot strap. Namun, disini anda harus selalu terkoneksi dengan internet (*online*).

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1">
  <title>Belajar Bootstrap</title>
  <link href="bootstrap/css/bootstrap.min.css"
rel="stylesheet">
  <script src="bootstrap/js/jquery-3.2.1.js"></script>
  <script src="bootstrap/js/bootstrap.min.js"></script>
</head>
<body>
<h1> Install Bootstrap Online Selesai </h1>
</body>
</html>

```

## **2.4 Text Editor**

*Text Editor* adalah suatu software aplikasi atau suatu program komputer yang memungkinkan Anda sebagai pengguna untuk membuat, mengubah atau mengedit file teks yang ada berupa plain text. Text editor ini sebenarnya bisa digunakan untuk membuat program-program komputer dan mengedit source code dari bahasa pemrograman. Selain itu, text editor juga bisa dimanfaatkan untuk membuat halaman web atau template web design dan juga membuat aplikasi tertentu. Software aplikasi satu ini memang secara umum ditujukan untuk mempermudah aktivitas pemrograman.

*Text Editor* memiliki fitur-fitur yang sangat kecil dan sederhana. Namun ada juga beberapa text editor kini sudah menawarkan fungsi yang luas dan kompleks. Apa saja contohnya? Unix dan Linux adalah contohnya. Dalam sistem operasinya sudah tersedia Editor VI (atau varian), tapi banyak juga yang mencakup editor Emacs. Sementara sistem operasi dari Windows itu sendiri menyediakan Notepad standar. Walaupun sudah tersedia secara bawaan, banyak programmer lebih menyukai text editor lainnya yang memang fiturnya lebih banyak atau lengkap.

### **2.4.1 Fungsi Text Editor**

Berikut merupakan fungsi text , Seperti :

a. Mencari Suatu Kata pada Banyak File Maupun Folder

Untuk mengolah sedemikian banyaknya teks, fitur find atau search menjadi fitur wajib untuk sebuah text editor. Tentu hal tersebut mudah dilakukan untuk membuka sebuah file pada text editor dan mencari kata yang diinginkan jika fungsi find/search sudah tersedia, tapi bagaimana kalau Anda juga ingin mencari file yang memuat makna tertentu.

b. Mengurutkan Data yang Ada

Pada umumnya mengurutkan data bagi programmer yaitu berkaitan dengan angka atau numerik, semacam mengurutkan angka dari yang terbesar hingga terkecil atau sebaliknya. Tetapi memang adakalanya seorang programmer juga ingin mengurutkan data teks sesuai keperluan pribadi dengan urutan khusus. Disinilah text editor tingkat advanced juga dapat berfungsi untuk mempermudah pekerjaan Anda.

c. Membandingkan Satu File dengan yang Lainnya

Fungsi text editor selanjutnya adalah membandingkan dua file. Misalkan Anda punya dua file yang sekilas tampaknya sama saja, namun terdapat perbedaan konten dari kedua file tersebut yang sulit untuk dilacak secara manual. Nah, dengan begitu Anda tentu perlu bantuan dari sebuah tool untuk membandingkan dan menunjukan apa

yang berbeda dari kedua file tersebut. Ini juga merupakan peranan text editor.

## **2.4.2 Macam-macam Text Editor**

### **2.4.2.1 Atom**

Atom merupakan text editor yang bersifat free source dan open source bagi macOS, Linux, dan Microsoft Windows. Atom dilengkapi dengan plug-in yang ditulis dalam Node.js yang dikembangkan oleh GitHub . Atom merupakan aplikasi desktop yang dibuat memakai teknologi web.



Gambar Logo Atom

#### **2.4.2.1.1 Kelebihan Atom**

- a. Atom merupakan salah satu aplikasi open source alias gratis. Dengan menggunakan Atom Anda tidak akan dituntut untuk membeli lisence product ini dengan tarif tertentu.
- b. Dengan menggunakan text editor Atom Anda sebagai programmer bisa langsung terkoneksi dengan project Anda yang ada di Github. Tentunya jika Anda

menginginkannya, Anda sudah harus melakukan instalasi git svn.

- c. Dengan menggunakan text editor Atom, Anda bisa dengan mudah melakukan instalasi packages yang Anda butuhkan demi mendukung performa dan kebutuhan Anda sebagai programmer dalam menulis code.
- d. Atom juga menyediakan banyak theme yang dapat Anda gunakan untuk mempercantik text editor kesayangan agar Anda tidak bosan saat menulis dan melihat lautan teks coding.
- e. Ada juga fitur customize styling yang artinya Anda bebas merubah settingan Atom yang mau Anda pakai, karena pada dasarnya memang atom dibuat menggunakan electron dan javascript, jadi untuk style sama seperti Anda menulis file css.

#### **2.4.2.2 Sublime**

Sublime merupakan text editor yang secara eksklusif didukung dengan aplikasi pemrograman interface Python (API). API ini secara native mendukung banyak bahasa pemrograman dan bahasa markup. Jadinya fungsi-fungsi tambahan bisa didapatkan oleh pengguna dengan plugin. Biasanya plugin-plugin itu dibuat oleh komunitas dan dipelihara di bawah lisensi free software yang ada.



Gambar Logo Sublime

#### 2.4.2.2.1 Kelebihan Sublime

- a. Adanya *multiple selection*. Fungsinya yaitu untuk menghasilkan perubahan pada sebuah kode dalam waktu yang sama dan dalam baris yang berbeda. Multiple selection ini juga adalah salah satu fitur unggulan dari Sublime Text Editor. Anda bisa saja meletakkan kursor pada kode yang akan di ubah / edit, lalu tekan Ctrl + click atau blok kode yang akan diubah kemudian Ctrl + D. Nah, setelahnya Anda bisa merubah kode secara bersamaan.
- b. Adanya command pallete. Fitur itu mempunyai fungsi yang berguna untuk mengakses file shortcut dengan gampang. Untuk mencari file tersebut Anda bisa tekan Ctrl + Shift + P , kemudian cari perintah yang memang Anda inginkan.
- c. Adanya distraction free mode. Fitur satu ini punya fungsi untuk merubah tampilan layar menjadi penuh dengan menekan SHIFT + F11. Fitur ini sebenarnya memang sangat dibutuhkan ketika Anda ingin fokus

pada bagian pekerjaan tertentu yang sedang dikerjakan saja.

- d. Adanya find in project. Dalam fitur ini Anda bisa mencari dan membuka file di dalam sebuah project dengan cepat dan mudah. Anda tinggal tekan Ctrl + P dan Anda sudah bisa mencari file yang diinginkan.
- e. Plugin yang berbasis Python Plugin API. Teks editor ini punya plugin yang sangat beragam dan ini bisa memudahkan Anda dalam mengembangkan software-nya.

#### **2.4.2.3 Notepad++**

Notepad++ Text Editor merupakan text editor yang digunakan untuk Microsoft Windows. Text editor satu ini mendukung pengeditan tab, yang memungkinkan Anda untuk bekerja dengan banyak file terbuka dalam satu window. Nama proyeknya berasal dari operator peningkatan C. Notepad ++ yang didistribusikan sebagai free software. Sejak 2015 Notepad ++ telah di-host di GitHub. Notepad ++ menggunakan komponen editor Scintilla.



Gambar Logo Notepad++



#### **2.4.2.3.1 Kelebihan Notepad++**

- a. Notepad++ bisa dibilang sangat ringan sebagai sebuah editing text dan coding text.
- b. link pada Notepad++ aktif dengan notasi http (bukan www ya). Jadi semacam http://xxx.com. Untuk menuju halaman website, Anda tinggal klik 2 kali url tersebut. Jadi, ini sangat bagus untuk Anda yang kapanpun ingin menyimpan bookmark pada file .txt dan sering mengunjungi halaman-halaman bookmark Anda.
- c. Dalam pengaturannya juga mudah, misalnya ketika program Notepad++ dijalankan akan langsung terbuka file terakhir yang Anda buka. Dapat di atur juga default directory-nya sesuai dengan keinginan. Misalkan: D:\txt-notes, jadi setiap kali Anda menjalankan Notepad++, dan membuka file txt, Anda tidak perlu bingung mencari-cari lagi file-file txt Anda.
- d. Fitur word autocompletion pada Notepad++ sebenarnya bagus buat Anda yang suka menulis draft artikel yang bersifat SEO (Search Engine Optimatation) friendly. Karena Anda bisa mengetahui apakah sebuah kata kunci yang berkaitan sudah masuk ke dalam draft artikel Anda atau belum lewat text editor satu ini.
- e. Text editor ini punya multi tab sehingga dapat membuka file secara bersamaan.

#### **2.4.2.4 Komodo Edit**

Komodo Edit adalah software editor program multi bahasa gratis yang dapat membuat penulisan kode program berkualitas menjadi mudah. Pengeditan program untuk bahasa dinamis termasuk Perl, PHP, Python, Ruby dan Tcl, ditambah dukungan kode browser sampling termasuk CSS, JavaScript, HTML, dan XHTML.



Gambar Logo Komodo Edit

##### **2.4.2.4.1 Kelebihan Komodo Edit**

- a. Editor Canggih
  1. Semua fundamental: Melipat kode, mengedit kolom / blok, mengomentari blok / baris, dan banyak lagi.
  2. Deteksi bahasa yang cerdas: Menyoroti dan memeriksa sintaks, mendeteksi beberapa bahasa dalam file.

3. Pembagian penampilan dan banyak jendela: Membagi sesi untuk mengedit berdampingan atau dengan beberapa jendela.
4. Penyorotan variabel: Klik pada variabel untuk melihat semua kejadian
5. Kotak alat: Potongan simpanan, makro atau perintah shell untuk memudahkan akses.
6. Cuplikan singkatan secara otomatis: Ketik singkatan untuk secara otomatis memicu pilihan potongan.
7. Pergi ke definisi: Langsung ke definisi variabel, class, dll.
8. Kunci pengikatan: Vi / Vim, OS X, emacs, termasuk Windows / Linux.
9. Makro dan cuplikan yang powerfull: Masukkan potongan dengan singkatan. Gunakan makro untuk membuat sekuen berulang secara otomatis.
10. Tinjauan browser: Pratinjau di jendela terpisah, di editor panel, atau dalam tampilan split.
11. Perluas / sesuaikan: Kunci binding, skema warna sintaks, pemeriksa sintaks, format secara otomatis.
12. Cuplikan dengan kondisi: Gunakan javascript tertanam untuk menentukan isi potongan saat runtime.

b. Pengkodean Keras

1. Calltips dan mengisi secara otomatis: Mendukung banyak bahasa. Mendeteksi beberapa bahasa dalam file.
2. Karakter perangkat lunak, pencocokan pasangan: mengisi secara otomatis pasangan kode, dan menyoroti pasangan kode yang cocok. Memberi spasi secara otomatis dan memformat secara otomatis.
3. Dukungan perpustakaan kelompok 3d: Termasuk katalog API mengaktifkan mengisi secara otomatis dan calltips untuk perpustakaan seperti PyWin32 dan jQuery.
4. Daftar bagian: Melihat dan menyaring daftar fungsi Anda, kelas, atau element.
5. Kode browser: Arahkan struktur sumber kode secara hirarki, penampilan dilipat.
6. Penampil DOM: Memeriksa dan mengedit pohon DOM untuk dokumen XML dan HTML.
7. Dukungan XML: Mengisi secara otomatis untuk dialek XML dapat ditambahkan melalui DTD atau skema RelaxNG.

#### 2.4.2.5 BlueFish

BlueFish merupakan text editor multi-platform yang juga dapat Anda gunakan di berbagai sistem operasi seperti Windows, Mac OS, dan Linux. Walaupun begitu aplikasi ini lebih optimal di sistem operasi Linux.

Meskipun sangat ringan, Bluefish bisa Anda gunakan untuk menangani ukuran file yang cukup besar secara berkala dan bisa menjaga kecepatan pemrosesannya tetap stabil. Anda juga dapat memanfaatkan berbagai macam fitur yang tersedia dalam toolbar maupun tab menu, seperti pencarian, format baris kode, dan penambahan tabel. Terdapat juga fitur snippet yang bisa menambah produktivitas.



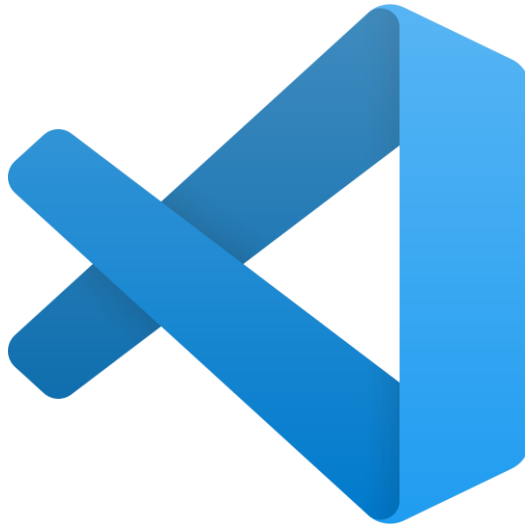
Gambar Logo BlueFish

#### **2.4.2.5.1 Kelebihan BlueFish**

- a. Ringan, editor ini berukuran kira-kira 4,4 MB
- b. Fitur mumpuni, fitur di dalam Bluefish yang paling dicari dan disukai adalah autocomplete syntax.
- c. Dukungan, Bluefish mempunyai keunggulan untuk pemakai yang fokus dalam pemrograman web. Fitur untuk HTML5 hingga CSS telah terselip dengan baik.
- d. Cross Platform, Bluefish dapat berjalan baik di Windows, Linux dan MacOS
- e. Bluefish dapat dimanfaatkan secara maksimal untuk mengeksplor dan mendevelop aplikasi website
- f. Bluefish juga dapat merupakan sebuah proyek open source dimana kita dapat berkontribusi untuk menyempurnakan aplikasi editor ini.

#### **2.4.2.6 Visual Studio Code**

Microsoft Visual Studio Code adalah one-stop shop yang memungkinkan kita fokus pada proses pengembangan dan melupakan tools baru.



Gambar Logo Visual Studio Code

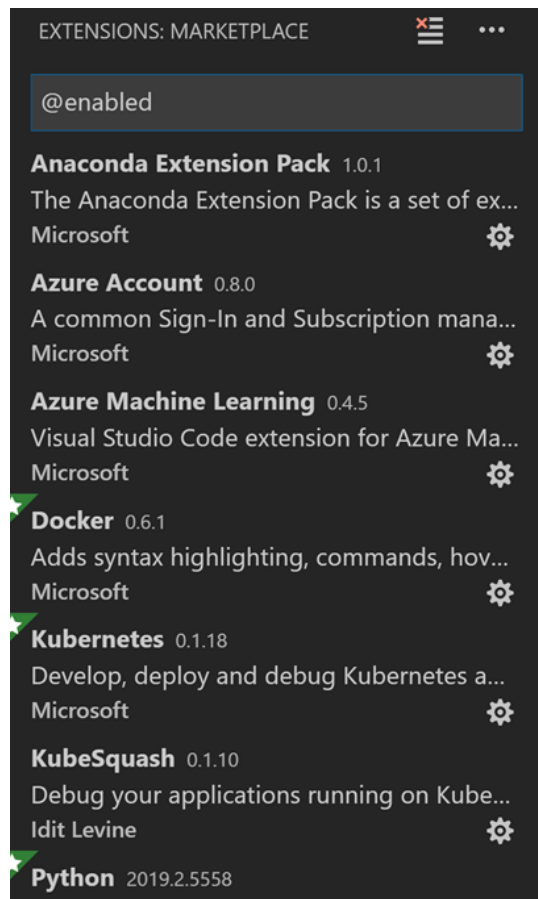
#### **2.4.2.6.1 Fitur-fitur Visual Studio Code**

##### **a. Cross platform**

Tersedia di macOS, Linux dan Windows artinya dapat bekerja pada sistem operasi manapun tanpa khawatir belajar coding tools yang sama untuk sistem yang berbeda-beda.

##### **b. Lightweight**

Tidak perlu menunggu lama untuk memulai. Anda mengontrol sepenuhnya bahasa, tema, debugger, commands dan lain-lainnya sesuai keinginan. Ini dapat dilakukan melalui extensions untuk bahasa populer seperti python, node.js, java dan lain-lainnya di Visual Studio Code Marketplace.



Gambar Pilihan Bahasa

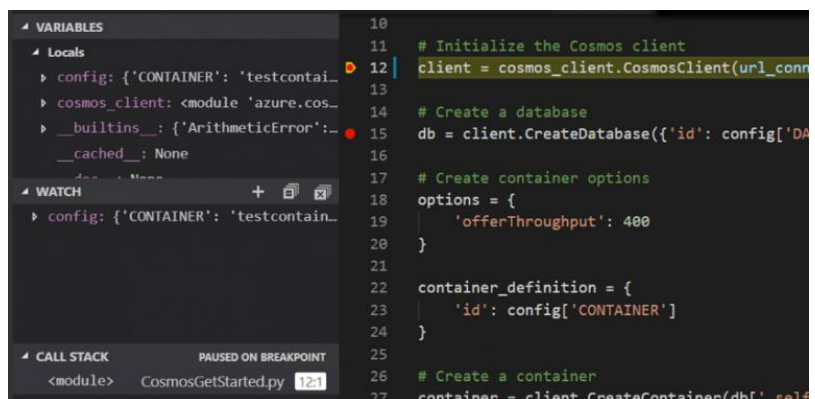
### c. Powerful editor

Memfungsikan fitur untuk source code editing yang sangat produktif, seperti membuat code snippets, IntelliSense, auto correct, dan formatting.



#### d. Code Debugging

Salah satu fitur terkeren yang ditawarkan Visual Studio Code adalah membantu Anda melakukan debug pada kode dengan cara mengawasi kode, variabel, call stack dan expression yang mana saja.



Gambar Debugging Visual Studio

#### e. Source control

Visual Studio Code memiliki integrated source control termasuk Git support in-the-box dan penyedia source code control lainnya di pasaran. Ini meningkatkan siklus rilis proyek Anda secara signifikan.

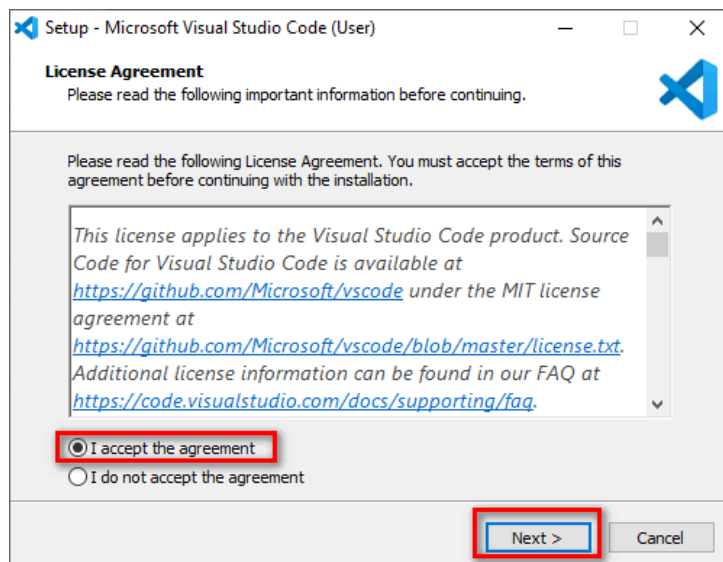
#### f. Integrated terminal

Tiada lagi multiple windows dan alt-tabs. Anda dapat melakukan command-line task sekejap dan membuat banyak terminal di dalam editor.

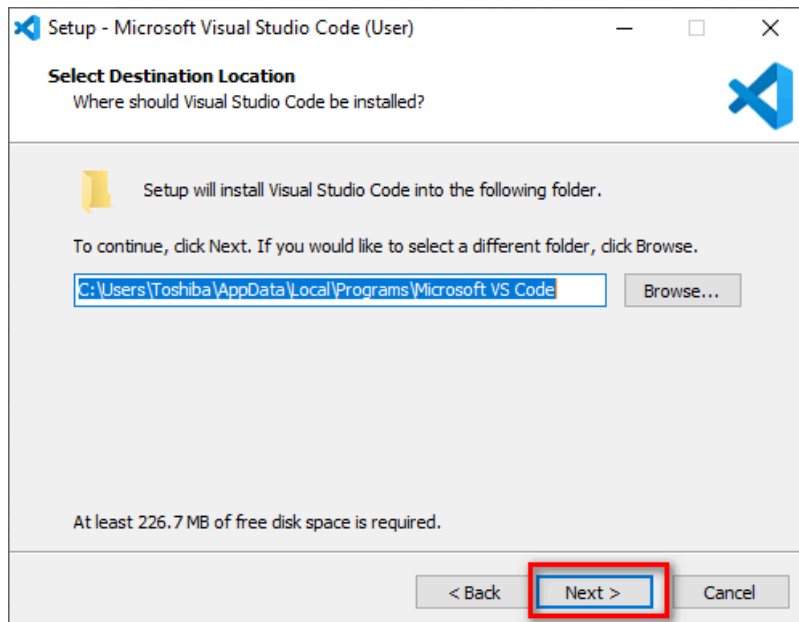
#### 2.4.2.6.2 Instalasi Visual Studio Code

Dalam Sistem ini penulis menggunakan Visual Studio Code sebagai text editor untuk membuat code web sistem, berikut merupakan cara instalasi Visual Studio Code :

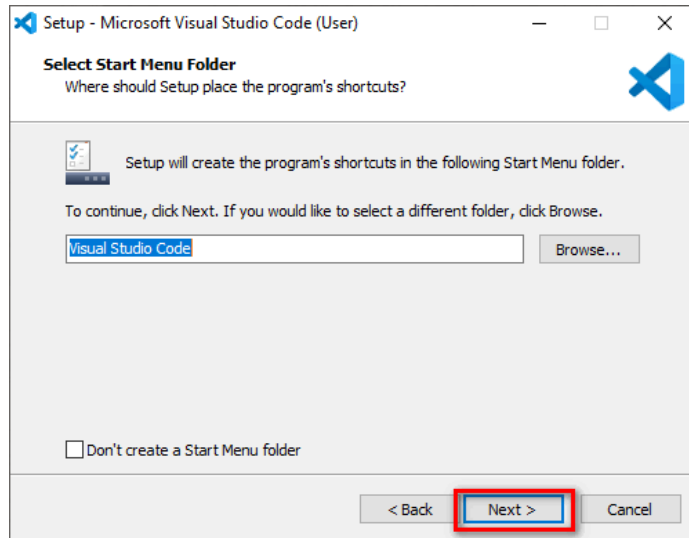
- a. Download Installer Visual Studio Code
- b. Double klik pada file installer nya atau klik kanan kemudian pilih Run as Administrator.
- c. Jika muncul peringatan Run as Administrator, silahkan klik Yes.
- d. Pilih “I accept the aggrement” untuk menyetujui “License Agreement”, kemudian klik Next.



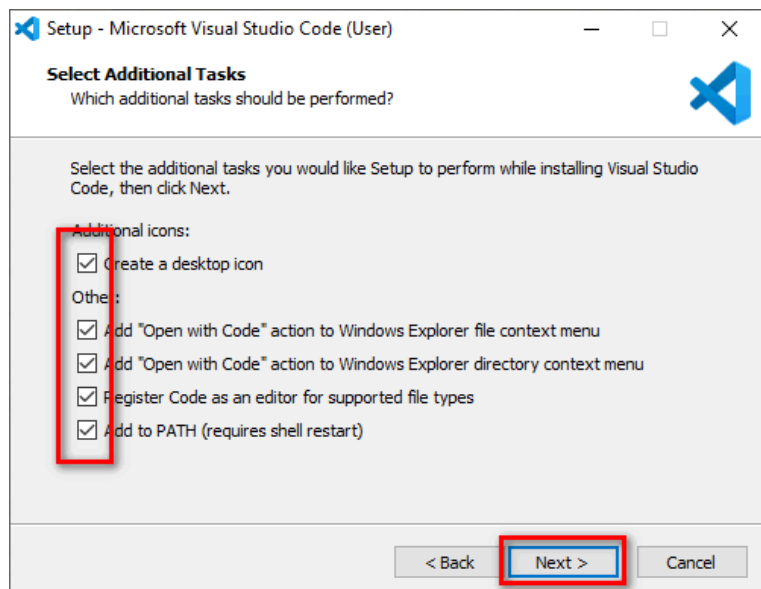
- e. Untuk Select Destination Location bisa di biarkan saja jika lokasi instalasi tidak akan di rubah. Klik Next.



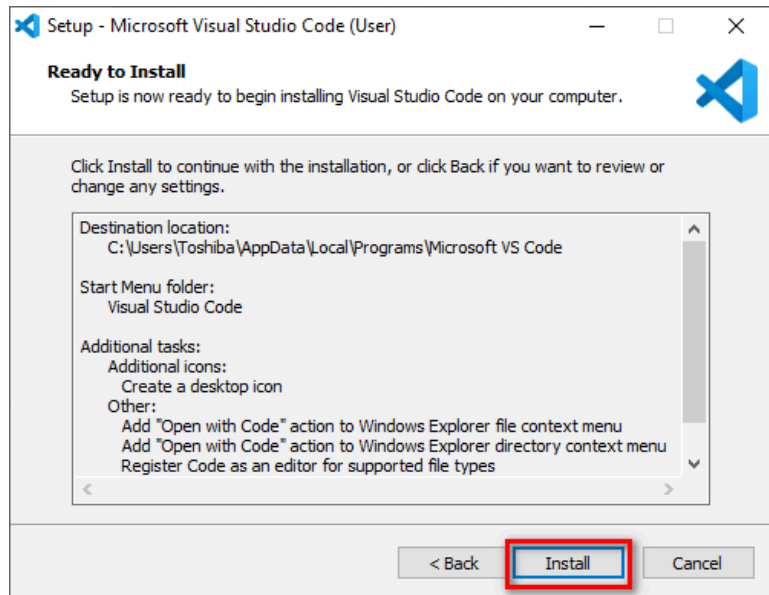
- f. Klik Next lagi jika tidak akan merubah Start Menu Folder.



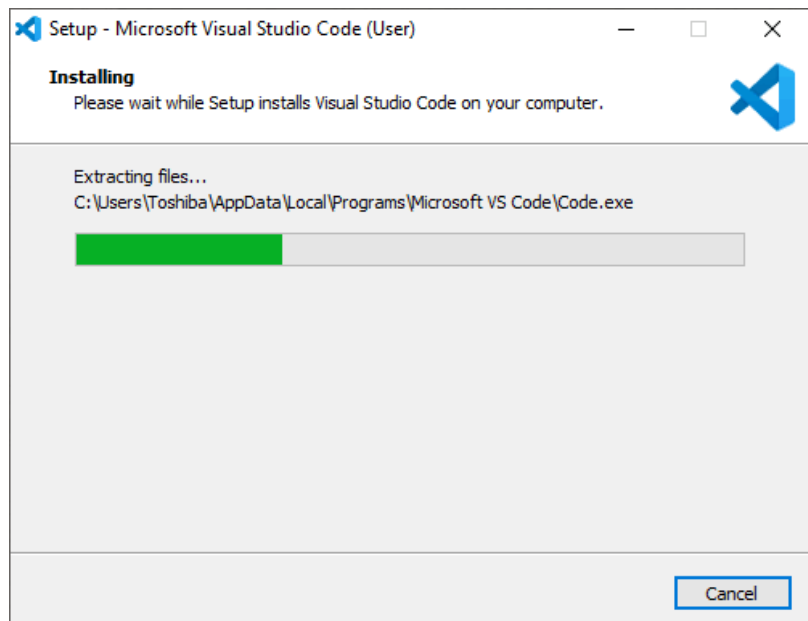
- g. Di bagian Select Additional Tasks centang semua. Kemudian Next.



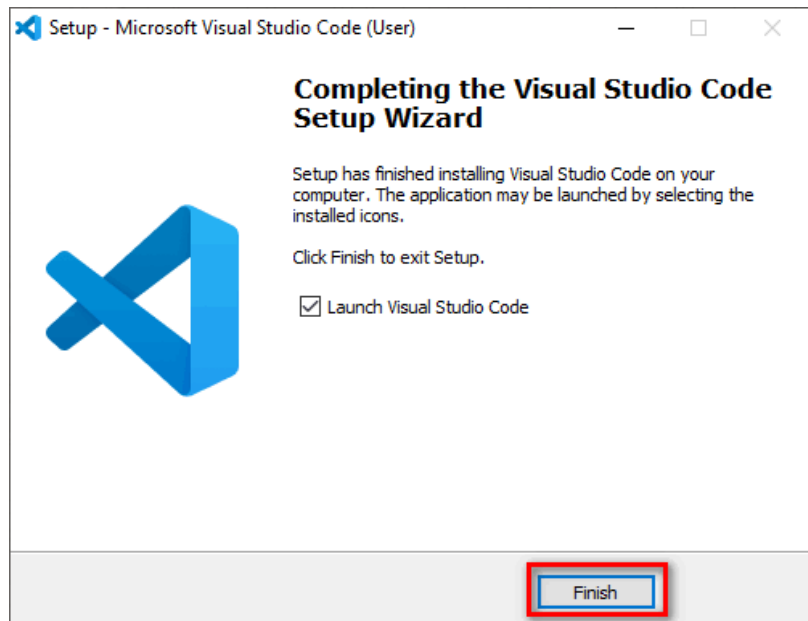
h. Lalu klik Install untuk memulai proses instalasi



i. Tunggu sampai proses instalasi selesai.



j. Setelah selesai klik Finish.



#### 2.4.2.6.4 Install Extensions Visual Studio Code

Setelah proses instalasi berhasil. Selanjutnya saya akan mencontohkan langkah-langkah install Extensions di Visual Studio Code. Saya akan coba install Extensions Python. Jika anda ingin melakukan coding Python, maka anda harus melakukan instalasi Python terlebih dahulu di laptop anda.

- a. Buka Visual Studio Code.
- b. Klik tab Extensions yang ada di sebelah kiri ( 1 ).
- c. Kemudian ketik “Python” di pencarian ( 2 ).
- d. Pilih Python ( 3 ).
- e. Kemudian klik Install pada Extensions: Python ( 4 ).



- f. Maka anda dapat mengcoding python di Visual Studio Code

## **BAB III**

### **3.1 Implementasi**

implementasi menurut KBBI (Kamus Besar Bahasa Indonesia) yaitu pelaksanaan atau penerapan. Sedangkan pengertian umum adalah suatu tindakan atau pelaksana rencana yang telah disusun secara cermat dan rinci (matang).

Kata implementasi sendiri berasal dari bahasa Inggris “to implement” artinya mengimplementasikan. Tak hanya sekedar aktivitas, implementasi merupakan suatu kegiatan yang direncanakan serta dilaksanakan dengan serius juga mengacu pada norma-norma tertentu guna mencapai tujuan kegiatan.

Implementasi merupakan tindakan yang dilakukan untuk mencapai suatu tujuan yang ditetapkan dalam suatu keputusan. Pada tindakan ini berusaha untuk mengubah keputusan tersebut

menjadi pola-pola operasional serta berusaha mencapai perubahan-perubahan besar atau kecil sebagaimana yang telah diputuskan sebelumnya. Implementasi pada hakikatnya juga merupakan upaya pemahaman apa yang seharusnya terjadi setelah program dilaksanakan.

### **3.2 Implementasi Sistem**

Implementasi sistem adalah tahap penerapan sistem yang akan dilakukan jika sistem disetujui termasuk program yang telah dibuat pada tahap perancangan sistem agar siap untuk dioperasikan. Implementasi sistem merupakan tahap penerapan sistem agar sistem tersebut dapat digunakan atau dioperasikan untuk mencapai tujuan.

Adapun beberapa tujuan dari implementasi sistem yaitu sebagai berikut :

- a. Membuat desain sistem selama melakukan penelitian dan analisa.
- b. Menguji dan mendokumentasikan prosedur dan program yang diperlukan
- c. Menyelesaikan desain sistem yang telah distujui.
- d. Memperhitungkan sistem yang telah dibuat sesuai kebutuhan yang akan digunakan

Terdapat 4 tahap dalam pengimplementasian sistem, yaitu :

- a) Membuat dan menguji basis data dan jaringan



Penerapan sistem yang baru atau perbaikan sistem dibuat pada basis data dan jaringan yang telah ada.

b) Membuat dan menguji program

Tahap ini merupakan tahap pertama untuk siklus pengembangan sistem yang spesifik, bertujuan untuk mengembangkan rencana yang lebih terperinci dalam pengembangan dan penguian.

c) Memasang dan menguji sistem baru

Pada tahap ini dilakukan untuk meyakinkan bahwa kebutuhan sistem lama terpenuhi pada sistem yang baru.

d) Serta mengirim sistem baru kedalam sistem yang lama.

Pada tahap ini yaitu mengubah secara perlahan-lahan sistem lama menjadi sistem baru.

### **3.3 Sistem**

Sistem berasal dari bahasa Yunani “*Systema*” yang berarti kesatuan. Pengertian sistem diambil dari asal mula sistem yang berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) yang memiliki pengertian bahwa suatu sistem merupakan suatu kesatuan yang didalamnya terdiri dari komponen atau elemen yang berhubungan satu dengan yang lainnya, yang berfungsi untuk memudahkan aliran informasi, materi atau *energy*.

Ada beberapa elemen yang membentuk sebuah sistem, yaitu :

a. Tujuan

Setiap sistem memiliki tujuan agar setiap sistem yang akan dibuat ataupun telah terbuat menjadi terarah dengan keinginan ataupun kebutuhan pengguna.

b. Masukan

Masukan pada sistem yaitu segala sesuatu yang masuk ke dalam sistem dan selanjutnya dapat menjadi bahan yang akan di proses. Masukan tersebut dapat berupa hal-hal yang berwujud maupun tidak berwujud.

c. Proses

Proses merupakan bagian yang melakukan perubahan ataupun transformasi dari masukan yang menjadi keluaran yang berguna dan lebih bernilai, seperti : informasi dan produk.

d. Keluaran

Keluaran merupakan hasil dari pemrosesan.

e. Batas

Pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

f. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian dengan umpan balik digunakan untuk mengendalikan suatu masukan maupun proses, untuk mengatur agar sistem berjalan sesuai dengan tujuan yang diinginkan.

## Jenis Sistem

Sistem terbagi menjadi 2 kategori :

- a. Atas dasar keterbukaan
  - 1) Sistem terbuka, dimana pihak luar dapat mengakses dan mempengaruhi sistem tersebut.
  - 2) Sistem tertutup
- b. Atas dasar komponen
  - 1) Sistem fisik, menggunakan komponen dengan energy
  - 2) Sistem non-fisik atau konsep yang berisikan ide-ide

## **3.4 Sistem Penjadwalan**

Sistem penjadwalan adalah informasi berupa data dalam proses mengatur rencana kegiatan dengan pembagian waktu pelaksanaan.

### 3.5 Penjadwalan

Penjadwalan Persoalan penjadwalan adalah persoalan pengalokasian pekerjaan ke mesin,

pada kondisi mesin mempunyai kapasitas dan jumlah terbatas.

Secara umum

masalah penjadwalan dapat dijelaskan sebagai  $n$  job ( $J_1, J_2, \dots, J_n$ ) yang harus

diproses di  $m$  mesin ( $M_1, M_2, \dots, M_m$ ). Waktu yang diperlukan untuk memproses  $n$  pekerjaan  $J_i$  pada mesin  $M$  adalah  $P_{iM}$  setiap job harus diproses tanpa dihentikan selama waktu proses  $p$  mesin hanya dapat menangani satu job pada saat yang sama, dan secara terus menerus tersedia sejak waktu nol (time zero). Pemecahan permasalahan yang diinginkan adalah mendapatkan jadwal yang optimal, yaitu menyelesaikan semua pekerjaan dengan mendapatkan jadwal yang optimal yaitu menyelesaikan semua pekerjaan dengan adanya  $k$

Penjadwalan (*scheduling*), didefinisikan sebagai proses pengalokasian sumber untuk memilih sekumpulan tugas dalam jangka waktu tertentu. Definisi ini dapat dijabarkan dalam arti yang berbeda, yaitu :

1. Penjadwalan merupakan sebuah fungsi pengambilan keputusan dalam menentukan jadwal yang paling tepat.

2. Penjadwalan merupakan teori yang berisi sekumpulan prinsip, model, teknik, dan konklusi logis dalam proses pengambilan keputusan. Sistem penjadwalan adalah informasi berupa data dalam proses mengatur rencana kegiatan dengan pembagian waktu pelaksanaan. Priority Scheduling merupakan algoritma penjadwalan berprioritas.
3. Kegiatan akademik adalah kegiatan pembelajaran di dalam dan/atau di luar ruang kuliah, studio dan/atau laboratorium, pengerjaan tugas-tugas, evaluasi pembelajaran, dan kegiatan administrasi yang menyertainya.

Berdasarkan Kamus Besar Bahasa Indonesia, jadwal merupakan pembagian waktu berdasarkan rencana pengaturan urutan kerja. Jadwal juga didefinisikan sebagai daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Sedangkan penjadwalan merupakan proses, cara, perbuatan menjadwalkan atau memasukkan dalam jadwal .

#### Tujuan Penjadwalan

Tujuan penjadwalan adalah untuk mengurangi waktu keterlambatan dari batas waktu yang ditentukan agar dapat memenuhi batas waktu yang telah disetujui dengan konsumen, penjadwalan juga dapat meningkatkan

produktifitas mesin dan mengurangi waktu menganggur. Produktifitas mesin meningkat maka waktu menganggur berkurang, secara tidak langsung perusahaan dapat mengurangi biaya produksi. Semakin baik suatu penjadwalan semakin menguntungkan juga bagi perusahaan dan bisa menjadi acuan untuk meningkatkan keuntungan dan strategi bagi perusahaan dalam memuaskan pelanggan.

### **3.6 Jadwal Kegiatan Akademik**

Jadwal kegiatan akademik adalah pedoman dalam kegiatan akademik untuk jangka waktu satu tahun akademik. Jadwal akademik adalah tulang punggung waktu untuk kegiatan yang akan diselenggarakan dan informasi penting lainnya untuk diuraikan dalam kalender kegiatan

### **3.7 Karyawan**

Karyawan adalah tenaga kerja yang melakukan pekerjaan dan memberikan hasil kerjanya kepada pengusaha yang mengerjakannya dimana hasil karyanya itu sesuai dengan profesi atau pekerjaan atas dasar keahlian sebagai matapencariannya.

### **3.8 Metode Topsis**

Pertama kalinya oleh Yoon dan Hwang dan dinilai oleh surveyor dan operator yang berbeda. Metode Topsis standar dirancang untuk menemukan alternatif optimum yang memiliki

kedekatan minimum dengan penyelesaian ideal positif dan kedekatan maksimum terhadap penyelesaian ideal negatif.

Penyelesaian ideal

positif berupaya untuk memaksimalkan kriteria manfaat dan meminimumkan kriteria biaya, sedangkan penyelesaian ideal negatif justru sebaliknya. Secara umum prosedur TOPSIS mengikuti terdiri dari beberapa langkah dimulai dari membuat matriks keputusan hingga menentukan nilai preferensi untuk pengambilan keputusan yang ideal.

*Technique for Order Performance by Similarity to Ideal Solution* (TOPSIS) merupakan salah satu metode dalam pengambilan keputusan, yang mana dalam menghasilkan sebuah keputusan akan memilih alternatif yang tidak hanya paling mendekati solusi ideal positif, akan tetapi juga paling jauh dari solusi ideal negatif, maka langkah-langkah yang dilakukan dalam metode TOPSIS yaitu :

- a. Membangun matriks keputusan ternormalisasi
- b. Membangun matriks bobot ternormalisasi
- c. Menentukan solusi ideal positif dan negatif.
- d. Menghitung jarak setiap *alternative* keputusan dari solusi ideal positif dan negatif.
- e. Menentukan nilai preferensi untuk setiap *alternative*.

TOPSIS didasarkan pada konsep alternatif terpilih yang terbaik yang banyak digunakan pada beberapa model. Multi Attribute Decision Making (MADM) untuk menyelesaikan masalah keputusan secara praktis. Metode ini banyak diimplementasikan ke berbagai bidang. Pada MADM klasik



metode, seperti teknik untuk preferensi pesanan dengan kemiripan dengan solusi ideal (TOPSIS) yang dikembangkan oleh Hwang dan Yoon, Promethee, Viktor, Electre, bobot masing-masing atribut dan peringkat alternatif disajikan dengan angka tajam. Namun, di dunia nyata, pembuat keputusan mungkin lebih suka mengevaluasi atribut dengan menggunakan variabel linguistik (variabel yang bernilai kata/kalimat, bukan angka. Sebagai alasan menggunakan kata/kalimat dari pada angka karena peranan linguistik kurang spesifik dibandingkan angka namun informasi yang disampingkan lebih informatif) daripada nilai eksak karena pengetahuan parsial tentang atribut dan kurangnya kemampuan pemrosesan informasi dari domain masalah. Di situasi seperti itu, informasi preferensi alternatif disediakan oleh pembuat keputusan mungkin tidak jelas, tidak tepat, atau tidak lengkap. TOPSIS adalah metode beberapa kriteria untuk mengidentifikasi solusi dari satu set alternatif terbatas.

Metode TOPSIS adalah teknik untuk urutan preferensi oleh kesamaan untuk solusi ideal. Solusi ideal (juga disebut solusi ideal positif) merupakan solusi yang dapat memaksimalkan kriteria / atribut manfaat dan meminimalkan kriteria/ atribut biaya, sedangkan solusi ideal negatif (juga disebut solusi anti-ideal) memaksimalkan kriteria/ atribut biaya dan meminimalkan kriteria/ atribut manfaat. Alternatif terbaik

adalah salah satu yang paling dekat dengan solusi ideal positif dan terjauh dari solusi ideal negatif. TOPSIS adalah salah satu metode pengambilan keputusan multikriteria atau alternatif pilihan yang merupakan alternatif yang mempunyai jarak terkecil dari solusi ideal positif dan jarak terbesar dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak *Euclidean* (jarak garis lurus "biasa" antara dua titik dalam ruang Euclidean). Namun, alternatif yang mempunyai jarak terkecil dari solusi ideal positif, tidak harus mempunyai jarak terbesar dari solusi ideal negatif. Maka dari itu, TOPSIS mempertimbangkan keduanya, jarak terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif secara bersamaan. Solusi optimal dalam metode TOPSIS didapat dengan menentukan kedekatan relatif suatu alternatif terhadap solusi ideal positif. TOPSIS akan merangking alternatif berdasarkan prioritas nilai kedekatan relatif suatu alternative terhadap solusi ideal positif. Alternatif-alternatif yang telah dirangking kemudian dijadikan sebagai referensi bagi pengambil keputusan untuk memilih solusi terbaik yang diinginkan .

Metode ini merupakan metode kompromi yang banyak digunakan untuk menangani berbagai kriteria masalah pengambilan keputusan. Dasar pemikiran metode TOPSIS adalah bahwa alternatif yang dipilih harus memiliki jarak

terdekat solusi positif ideal dan jarak terjauh dari solusi *negative* ideal. Bobot kriteria dalam metode TOPSIS diberikan apriori. Selain itu, solusi positif-ideal dan solusi *negative* ideal langsung dihasilkan dari matriks keputusan. Dalam Metode TOPSIS klasik, semua data keputusan diketahui secara tepat atau diberikan sebagai nilai yang tajam. Namun, ini bisa sulit untuk ditentukan data justru karena penilaian manusia sering tidak jelas dalam banyak kondisi dalam praktiknya .

TOPSIS membutuhkan rating kinerja setiap alternatif  $A_i$  pada setiap kriteria  $C_j$  yang ternormalisasi yaitu :

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}$$

Keterangan =  $i=1,2,\dots,m$  dan  $j=1,2,\dots,n$ .

Sebelum mencari nilai  $r_{ij}$ , maka cari mencari nilai dengan rumus seperti ini :

$$|X_n| = \sqrt{\sum_{i=1}^m x_{ij}^2}$$

Menghitung nilai matriks yang ternormalisasi yang terbobot ( $Y$ ), untuk bobot yang sudah ditentukan ( $W$ ), dengan rumus :

$$y_{ij} = w_i * r_{ij}$$

Solusi Ideal Positif  $A^+$  dan solusi ideal negatif  $A^-$  dapat ditentukan berdasarkan rating bobot ternormalisasi ( $y_{ij}$ ) sebagai :

$$A^+ = \max(y_{1+}, y_{2+}, \dots, y_{n+})$$

$$A^- = \max(y_{1-}, y_{2-}, \dots, y_{n-})$$

Jarak antara alternatif  $A_i$  dengan solusi ideal positif dirumuskan sebagai :

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2}$$

Jarak antara alternatif  $A_i$  dengan solusi ideal negatif dirumuskan sebagai :

$$D_i^- = \sqrt{\sum_{j=1}^n (y_i^- - y_{ij})^2}$$

Nilai preferensi untuk setiap alternatif ( $V_i$ ) diberikan sebagai :

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}$$

Keterangan :  $i=1,2,\dots,n$ .

Nilai  $V_i$  yang lebih besar menunjukkan bahwa Alternatif  $A_i$  lebih dipilih

### 3.8.1 Tujuan Metode

Pada dasarnya, proses MADM (*Multi Attribute Decision Making*) dilakukan melalui 3 tahap, yaitu:

1. Penyusunan komponen-komponen situasi.

2. Analisis.
3. Sintesis informasi.

Pada tahap penyusunan komponen, komponen situasi, akan dibentuk tabel taksiran yang berisi identifikasi alternatif dan spesifikasi tujuan, kriteria dan atribut.

MADM (*Multi Attribute Decision Making*) adalah salah satu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria-kriteria tertentu. Inti dari *Multiple Attribute Decision Making* (MADM) adalah menentukan nilai bobot untuk setiap atribut/kriteria, yang kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Pada dasarnya, ada 3 (tiga) pendekatan untuk mencari nilai bobot atribut, yaitu :

1. Pendekatan subyektif
2. Pendekatan obyektif
3. Pendekatan integrasi antara subyektif & obyektif.

Masing-masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subyektif, nilai bobot ditentukan berdasarkan subyektifitas dari para pengambil keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari pengambil keputusan .

### **3.9 Metodologi Penelitian**

Metodologi penelitian yang digunakan dalam pengumpulan data adalah :

1. Data primer adalah data yang diambil langsung dari obyek penelitian atau merupakan data yang berasal dari sumber asli atau pertama. Teknik pengumpulan data primer dilakukan melalui teknik observasi dengan cara mengumpulkan informasi langsung ke lokasi penelitian untuk mengamati bagaimana penentuan untuk pemilihan karyawan yang dilakukan oleh pihak admin dalam kegiatan akademik tersebut.
2. Data sekunder adalah data yang tidak didapatkan secara langsung dari objek penelitian, melainkan data yang berasal dari sumber yang telah dikumpulkan oleh pihak lain. Teknik pengumpulan data sekunder dilakukan dengan cara studi dokumentasi dan studi literature.

Tujuan dan Solusi Pemecahan Masalah Pada Sistem Pendukung Keputusan

a. Tujuan Sistem Pendukung Keputusan (SPK)

Tujuan dari SPK (Sistem Pendukung Keputusan) yang dikemukakan oleh Peter G.W Keen dan Scoot Morton, yaitu :

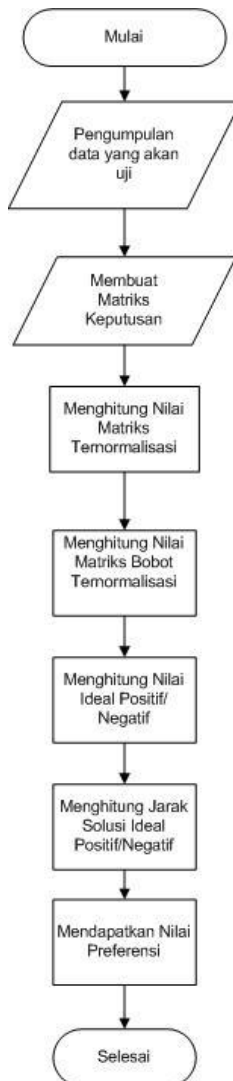
- a) Mendukung penilaian manajer bukan mencoba ataupun menggantinya

- b) Membantu manajer dalam membuat keputusan untuk memecahkan masalah semi terstruktur.
  - c) Meningkatkan efektifitas pengambilan keputusan manajer dari pada efisiensinya.
- b. Komponen-komponen Sistem Pendukung Keputusan (SPK), terdiri :
- a) Data Management. Termasuk database yang mengandung data yang relevan untuk berbagai situasi dan diatur oleh software yang disebut dengan Database Management System (DBMS).
  - b) Model Management. Model ini melibatkan model finansial, statistical, management science, atau berbagai model kuantitatif lainnya, sehingga memberikan suatu kemampuan analitis, dan manajemen software yang diperlukan.
  - c) Communication (dialog subsystem). User dapat berkomunikasi dan memberikan perintah DSS melalui sub sistem.
  - d) Knowledge Management. Subsistem optional ini mendukung sub sistem lain atau bertindak sebagai komponen yang berdiri sendiri.

#### 3.9.1 Alur Metodologi

Pada Penelitian ini penulis menerapkan metode TOPSIS pada proses perhitungan akurasi keakuratan pemilihan peserta kegiatan akademik di PT Pegadaian. Untuk

menyelesaikan masalah tersebut perlu diterapkan nya sebuah metodologi, metodologi yang akan digunakan yaitu metodologi kualitatif, sebagai berikut :



Gambar 4. 1 Alur Metodologi Penelitian



Metodologi kualitatif merupakan data yang bersifat induktif dan hasil penelitian kualitatif lebih melalui pengumpulan data, analisis, kemudian di interpretasikan . Maka langkah-langkah yang dilakukan adalah sebagai berikut :

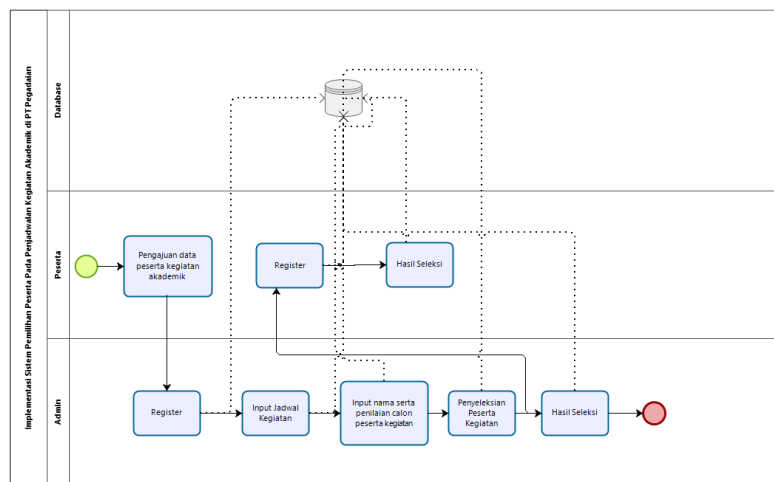
1. Pengumpulan data yang akan di uji.
2. Membuat nilai matriks keputusan.
3. Membangun matriks keputusan ternormalisasi. Dalam TOPSIS kinerja dari setiap alternative dihitung dengan menggunakan Persamaan 1.
4. Membangun matriks bobot ternormalisasi. Solusi ideal positif  $A^+$  dan negatif  $A^-$  dapat ditentukan berdasarkan pada rating bobot ternormalisasi seperti Persamaan 2.
5. Menentukan solusi ideal positif dan negatif Matriks solusi ideal positif dapat dihitung dengan Persamaan 3, sedangkan matriks solusi ideal negatif dapat dihitung berdasarkan Persamaan 4.
6. Menghitung jarak setiap alternatif keputusan dari solusi ideal positif dan negatif. Jarak antara alternatif  $A_i$  dengan solusi ideal positif dapat dihitung dengan Persamaan 5.
7. Menentukan nilai preferensi untuk setiap alternatif.

### 3.9.2 Karakteristik pada penelitian Kualitatif

1. Mendeskripsikan suatu keadaan yang sebenarnya atau fakta, dalam bentuk laporan.
2. Penelitian dilakukan pada kondisi langsung ke sumber data dan penelitian.
3. Penelitian kualitatif dalam cara analisis datanya dilakukan secara induktif.

## **BAB IV**

Dalam pembuatan suatu sistem diperlukan adanya sebuah perancangan sistem yang akan dibuat, berikut merupakan salah satu perancangan sistem yang akan dibangun :



Gambar 4.1 Analisis Prosedur Yang Akan Dibangun

Prosedur yang akan dibangun :

1. Peserta melakukan pengajuan data peserta kegiatan akademik kepada admin.
2. Admin melakukan register untuk menginputkan jadwal kegiatan peserta yang akan mengikuti kegiatan akademik.
3. Setelah menginputkan jadwal admin akan menginputkan nama calon peserta kegiatan.

4. Admin melakukan penyeleksian peserta kegiatan berdasarkan nilai-nilai yang telah didapatkan dari kriteria perusahaan yang diinputkan.
5. Setelah proses penyeleksian maka akan keluar hasil berdasarkan nilai yang diinputkan.
6. Peserta akan melakukan register untuk melihat hasil pemilihan peserta kegiatan.

#### 4.2 Analisis Dokumen Yang Akan Dibangun

Di dalam sistem yang saat ini berjalan terdapat beberapa dokumen yang digunakan sebagai arsip untuk menyimpan data di pos market. Adapun dokumen yang dimaksud adalah sebagai berikut :

Tabel 4.1 Dokumen Admin

<b>Dibuat Oleh</b>	Admin
<b>Dibuat Untuk</b>	Admin
<b>Isi</b>	Berupa jadwal kegiatan akademik dan kuota peserta, pemilihan karyawan dalam kegiatan akademik
<b>Frekuensi</b>	Dibuat form pemilihan peserta akademik
<b>Tujuan</b>	Pendataan peserta, jumlah peserta dan nama peserta kegiatan

Tabel 4.2 Dokumen *User*

<b>Dibuat oleh</b>	Admin
<b>Dibuat untuk</b>	User
<b>Isi</b>	Berupa form hasil penilaian peserta kegiatan akademik
<b>Frekuensi</b>	Dibuat form untuk melihat hasil penilaian
<b>Tujuan</b>	Melihat penilaian peserta kegiatan akademik

#### 4.1.2.3 Analisis Metode TOPSIS

Metode ini merupakan metode kompromi yang banyak digunakan untuk menangani berbagai kriteria masalah pengambilan keputusan. Dasar pemikiran metode TOPSIS adalah bahwa alternatif yang dipilih harus memiliki jarak terdekat solusi positif ideal dan jarak terjauh dari solusi *negative* ideal. Bobot kriteria dalam metode TOPSIS diberikan apriori. Selain itu, solusi positif-ideal dan solusi *negative* ideal langsung dihasilkan dari matriks keputusan.

Langkah-langkah metode TOPSIS dapat dilakukan seperti yang sudah dijelaskan pada BAB II . Berikut merupakan tahap penerapan metode TOPSIS :

### 1. Data Kriteria dan Bobot

No	Kriteria	Atribut	Bobot
1	Grade	Benefit	20
2	Kesesuaian Program kerja Corpu	Benefit	20
3	Keikutsertaan	Cost	15
4	Tema (Jobdesk)	Benefit	25
5	Sharing	Benefit	20

Deskripsi Bobot beserta Nilai :

- a. Grade (20) = Untuk meningkatkan *performance* mengenai kesesuaian dengan tema
- b. Kesesuaian Proker Corpu (20) = Untuk mendukung program kerja dari Coporate University
- c. Keikutsertaan (15) = Mengenai keadilan keikutsertaan karyawan dalam kegiatan pertahun
- d. Tema (Jobdesk) (25) = Untuk meningkatkan kompetensi karyawan sehingga diharapkan dapat meningkatkan *Performance* karyawan dalam bekerja.
- e. Sharing (20) = Materi yang telah diseminarkan harus dibagikan untuk menghidupkan sistem KMS (*Knowledge Management System*) PT Pegadaian (Persero).

## 2. Data Matriks Berdasarkan Kriteria

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin</i>
Eka Pebriansyah	5	4	1	5	5
Supriyanto	5	5	3	5	5
Endang Pertiwi	5	4	1	5	5
Muh Ariyadi	5	4	1	5	5
Hertin Maulida	5	4	1	5	5
Elvi Rofiqatul Hidayah	5	5	4	5	5
Teguh Budi Ismanto	5	4	2	5	5
Eri Mardianto	5	4	2	5	5
Sriyani	5	4	1	5	5

Deskripsi Penilaian Perusahaan :

1. *Grade* dengan Nilai
  - a. Tenaga Pembantu (1)
  - b. Tenaga Keamanan (2)
  - c. Karyawan Grade 4-7 (3)
  - d. Karyawan Grade 8-11 (4)
  - e. Karyawan Grade 12-15 (5)
2. Kesesuaian Program kerja Corpu dengan Nilai  
Nilai rata-rata dari penilaian *grade*, *tema*, *sharing*, dan keikutsertaan
3. Keikutsertaan dengan Nilai
  - a. (1-5) Berdasarkan banyaknya keikutsertaan peserta dalam jangka waktu setahun
4. Tema dengan Nilai
  - a. Keterkaitan dengan bagian tersebut Grade 4-7 (1)
  - b. Keterkaitan dengan bagian tersebut Grade 8-11 (2)
  - c. Keterkaitan dengan *jobdesk* dan seminar Grade 4-7 (3)
  - d. Keterkaitan dengan *jobdesk* dan seminar Grade 8-11 (4)
  - e. Keterkaitan dengan *jobdesk* dan seminar Grade 12-15 (5)



5. *Sharing* (Membagikan Materi Seminar di KMS) dengan Nilai

- a. Grade 1 (1)
- b. Grade 2-3 (2)
- c. Grade 4-7 (3)
- d. Grade 8-11 (4)
- e. Grade 12-15 (5)

3. Pemberian bobot pada masing-masing peserta, kemudian pencarian nilai dengan rumus sebagai berikut :

$$|X_n| = \sqrt{\sum_{i=1}^m x_{ij}^2}$$

$$|X_1| = \sqrt{(5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2} = 15,0000$$

$$|X_2| = \sqrt{(4)^2 + (5)^2 + (4)^2 + (4)^2 + (4)^2 + (5)^2 + (4)^2 + (4)^2 + (4)^2} = 12,7279$$

$$|X_3| = \sqrt{(1)^2 + (3)^2 + (1)^2 + (1)^2 + (1)^2 + (4)^2 + (2)^2 + (2)^2 + (1)^2} = 6,1644$$

$$|X_4| = \sqrt{(5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2} = 15,0000$$

$$|X_5| = \sqrt{(5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2} = 15,0000$$

1. Setelah rumus  $X_n$  maka selanjutnya menggunakan rumus  $r_{ij}$

:

$$r_{ij} = \frac{x^{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}$$

$R_{10} = 5/15,0000 = 0,333$	$R_{20} = 4/12,7279 = 0,314$	$R_{30} = 1/6,1644 = 0,162$
$R_{11} = 5/15,0000 = 0,333$	$R_{21} = 5/12,7279 = 0,393$	$R_{31} = 3/6,1644 = 0,487$
$R_{12} = 5/15,0000 = 0,333$	$R_{22} = 4/12,7279 = 0,314$	$R_{32} = 1/6,1644 = 0,162$
$R_{13} = 5/15,0000 = 0,333$	$R_{23} = 4/12,7279 = 0,314$	$R_{33} = 1/6,1644 = 0,162$
$R_{14} = 5/15,0000 = 0,333$	$R_{24} = 4/12,7279 = 0,314$	$R_{34} = 1/6,1644 = 0,162$
$R_{15} = 5/15,0000 = 0,333$	$R_{25} = 5/12,7279 = 0,393$	$R_{35} = 4/6,1644 = 0,649$
$R_{16} = 5/15,0000 = 0,333$	$R_{26} = 4/12,7279 = 0,314$	$R_{36} = 2/6,1644 = 0,324$
$R_{17} = 5/15,0000 = 0,333$	$R_{27} = 4/12,7279 = 0,314$	$R_{37} = 2/6,1644 = 0,324$
$R_{18} = 5/15,0000 = 0,333$	$R_{28} = 4/12,7279 = 0,314$	$R_{38} = 1/6,1644 = 0,162$
$R_{40} = 5/15,0000 = 0,333$	$R_{50} = 5/15,0000 = 0,333$	
$R_{41} = 5/15,0000 = 0,333$	$R_{51} = 5/15,0000 = 0,333$	
$R_{42} = 5/15,0000 = 0,333$	$R_{52} = 5/15,0000 = 0,333$	
$R_{43} = 5/15,0000 = 0,333$	$R_{53} = 5/15,0000 = 0,333$	
$R_{44} = 5/15,0000 = 0,333$	$R_{54} = 5/15,0000 = 0,333$	
$R_{45} = 5/15,0000 = 0,333$	$R_{55} = 5/15,0000 = 0,333$	

$R_{46} = 5/15,0000 = 0,333$	$R_{56} = 5/15,0000 = 0,333$
$R_{47} = 5/15,0000 = 0,333$	$R_{57} = 5/15,0000 = 0,333$
$R_{48} = 5/15,0000 = 0,333$	$R_{58} = 5/15,0000 = 0,333$

2. Menghitung matriks ternormalisasi yang sudah terbobot dengan bobot yang sudah ditentukan ( $W$ ) = [4,5,3,5,2], dengan rumus :

$$y_{ij} = w_i * r_{ij}$$

$R_{10} = 20 * 0,333 = 6,667$	$R_{20} = 20 * 0,314 = 6,285$	$R_{30} = 15 * 0,162 = 2,433$
$R_{11} = 20 * 0,333 = 6,667$	$R_{21} = 20 * 0,393 = 7,857$	$R_{31} = 15 * 0,487 = 7,300$
$R_{12} = 20 * 0,333 = 6,667$	$R_{22} = 20 * 0,314 = 6,285$	$R_{32} = 15 * 0,162 = 2,433$
$R_{13} = 20 * 0,333 = 6,667$	$R_{23} = 20 * 0,314 = 6,285$	$R_{33} = 15 * 0,162 = 2,433$
$R_{14} = 20 * 0,333 = 6,667$	$R_{24} = 20 * 0,314 = 6,285$	$R_{34} = 15 * 0,162 = 2,433$
$R_{15} = 20 * 0,333 = 6,667$	$R_{25} = 20 * 0,393 = 7,857$	$R_{35} = 15 * 0,649 = 9,733$
$R_{16} = 20 * 0,333 = 6,667$	$R_{26} = 20 * 0,314 = 6,285$	$R_{36} = 15 * 0,324 = 4,867$
$R_{17} = 20 * 0,333 = 6,667$	$R_{27} = 20 * 0,314 = 6,285$	$R_{37} = 15 * 0,324 = 4,867$
$R_{18} = 20 * 0,333 = 6,667$	$R_{28} = 20 * 0,314 = 6,285$	$R_{38} = 15 * 0,162 = 2,433$
$R_{41} = 25 * 0,333 = 8,333$	$R_{51} = 20 * 0,333 = 6,667$	
$R_{42} = 25 * 0,333 = 8,333$	$R_{52} = 20 * 0,333 = 6,667$	
$R_{43} = 25 * 0,333 = 8,333$	$R_{53} = 20 * 0,333 = 6,667$	
$R_{44} = 25 * 0,333 = 8,333$	$R_{54} = 20 * 0,333 = 6,667$	

$R_{45} = 25 * 0,333 = 8,333$	$R_{55} = 20 * 0,333 = 6,667$
$R_{46} = 25 * 0,333 = 8,333$	$R_{56} = 20 * 0,333 = 6,667$
$R_{47} = 25 * 0,333 = 8,333$	$R_{57} = 20 * 0,333 = 6,667$
$R_{48} = 25 * 0,333 = 8,333$	$R_{58} = 20 * 0,333 = 6,667$

### 3. Menentukan Solusi Ideal Matriks ( $A^+$ )

$$A^+ = \max(y_1^+, y_2^+, \dots, y_n^+)$$

$$Y_1 = \max(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

$$Y_2 = \max(6.285, 7.857, 6.285, 6.285, 6.285, 7.857, 6.285, 6.285, 6.285) = 7.857$$

$$Y_3 = \max(2.433, 7.300, 2.433, 2.433, 2.433, 9.733, 4.867, 4.867, 2.433) = 9.733$$

$$Y_4 = \max(8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333) = 8.333$$

$$Y_5 = \max(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

### 4. Menentukan Solusi Ideal Matriks ( $A^-$ )

$$A^- = \min(y_1^-, y_2^-, \dots, y_n^-)$$

$$Y_1 = \min(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

$$Y_2 = \min(6.285, 7.857, 6.285, 6.285, 6.285, 7.857, 6.285, 6.285, 6.285) = 6.285$$

$$Y_3 = \min(2.433, 7.300, 2.433, 2.433, 2.433, 9.733, 4.867, 4.867, 2.433) = 2.433$$

$$Y_4 = \min(8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333) = 8.333$$

$$Y_5 = \min(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

##### 5. Menghitung Jarak Solusi Ideal Positif ( $D^+$ )

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2}$$

$$\begin{aligned} D_1^+ &= \sqrt{(6.667 - 6.667)^2 + (6.285 - 7.857)^2 + (2.433 - 9.733)^2 + (8.333 - 8.333)^2 + (6.667 - 6.667)^2} \\ &= 7.467 \end{aligned}$$

$$\begin{aligned} D_2^+ &= \sqrt{(6.667 - 6.667)^2 + (7.857 - 7.857)^2 + (7.300 - 9.733)^2 + (8.333 - 8.333)^2 + (6.667 - 6.667)^2} \\ &= 2.433 \end{aligned}$$

$$\begin{aligned} D_3^+ &= \sqrt{(6.667 - 6.667)^2 + (6.285 - 7.857)^2 + (2.433 - 9.733)^2 + (8.333 - 8.333)^2 + (6.667 - 6.667)^2} \\ &= 7.467 \end{aligned}$$

$$\begin{aligned} D_4^+ &= \sqrt{(6.667 - 6.667)^2 + (6.285 - 7.857)^2 + (2.433 - 9.733)^2 + (8.333 - 8.333)^2 + (6.667 - 6.667)^2} \\ &= 7.467 \end{aligned}$$

$$\begin{aligned} D_5^+ &= \sqrt{(6.667 - 6.667)^2 + (6.285 - 7.857)^2 + (2.433 - 9.733)^2 + (8.333 - 8.333)^2 + (6.667 - 6.667)^2} \\ &= 7.467 \end{aligned}$$

$$\begin{aligned} D_6^+ &= \sqrt{(6.667 - 6.667)^2 + (7.857 - 7.857)^2 + (9.733 - 9.733)^2 + (8.333 - 8.333)^2 + (6.667 - 6.667)^2} \\ &= 0.000 \end{aligned}$$

$$D_7^+$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (4,867 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 5,114$$

$$D_8^+$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (4,867 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 5,114$$

$$D_9^+$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (2,433 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 7,467$$

## 6. Menghitung Jarak Solusi Ideal Negatif ( $D^-$ )

$$D_i^- = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2}$$

$$D_1^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 0,000$$

$$D_2^- =$$

$$\sqrt{(6,667 - 6,667)^2 + (7,857 - 6,285)^2 + (7,300 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} =$$

$$5,114$$

$$D_3^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 0,000$$

$$D_4^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 0,000$$

$$D_5^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 0,000$$

$$D_6^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (7,857 - 6,285)^2 + (9,733 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 7,467$$

$$D_7^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (4,867 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 2,433$$

$$D_8^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (4,867 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 2,433$$

$$D_9^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 0,000$$

## 7. Menghitung Nilai Preferensi

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}$$

$$V_1 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_2 = \frac{5,114}{5,114} + 2,433 = 0,6776$$

$$V_3 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_4 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_5 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_6 = \frac{7,467}{7,467} + 0,000 = 1,0000$$

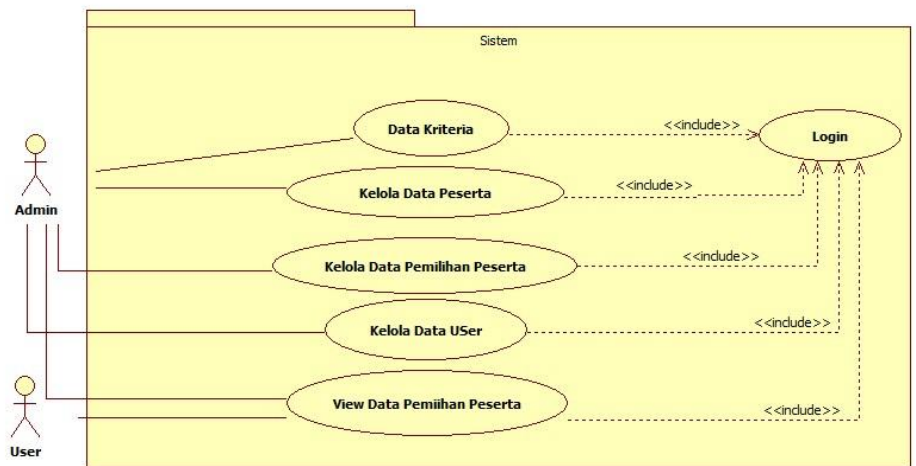
$$V_7 = \frac{2,433}{2,433} + 5,114 = 0,3224$$

$$V_8 = \frac{2,433}{2,433} + 5,114 = 0,3224$$

$$V_9 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

#### 5.1.2.4 UML (Unified Modelling Language)

##### 5.1.2.4.1 Usecase Diagram



Gambar 5.2 Usecase Diagram

#### Definisi Usecase

Tabel 5.3 Definisi Usecase

No	Use Case	Deskripsi
1.	Login	Melakukan proses login
2.	Data Kriteria	Merupakan serangkaian data kriteria yang akan berlangsung untuk akademik karyawan.



3.	Kelola Data Peserta	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola data peserta yang mengikuti kegiatan akademik.
4.	Kelola Data Pemilihan Peserta	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola data untuk pemilihan peserta.
5.	Kelola Data User	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola user atau pengguna.
6.	View Data Pemilihan Peserta	Melihat hasil dari data pemilihan peserta.

#### 5.1.2.4.1.2 Usecase Diagram Skenario

*Use Case* Skenario mendeskripsikan urutan langkah-langkah dalam proses sistem, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berikut ini penjelasan dari masing-masing skenario tersebut:

##### 5.1.2.4.1.2.1 Skenario Usecase Login Admin

Tabel 5. 4 Skenario *Usecase Login Admin*

Identifikasi	
Nomor	UC1
Nama	<i>Login admin</i>
Tujuan	Memberikan hak akses admin terhadap sistem dengan melakukan validasi terhadap <i>username</i> dan <i>password</i> yang dimasukan oleh admin
Deskripsi	

<b>Aktor</b>	Admin
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>Form Login</i> sudah tersedia
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Memasukkan <i>username</i> dan <i>password</i>	2. <i>Form Login</i> akan menampilkan <i>textbox</i> <i>username</i> , <i>password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ‘•’ pada la untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap <i>username</i> , <i>password</i> yang telah dimasukkan dengan menekan tombol <i>Ok</i> .	4. Aplikasi melakukan validasi terhadap <i>username</i> , <i>password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengece pada basis data.
<b>Kondisi Akhir</b>	Jika pada akhir interaksi <i>username</i> , <i>passw</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi se hak aksesnya.
<b>Alur Alternatif kondisi akhir</b> a. Jika tidak <i>valid</i> maka Aplikasi akan kembali ke tampilan awal login.	

#### 5.1.2.4.1.2.2 Skenario *Usecase Diagram Login User*

Tabel 5.5 Skenario *Use Case Login User*

Identifikasi	
Nomor	UC2
Nama	<i>Login User</i>
Tujuan	Memberikan hak akses peserta terhadap sistem dengan melakukan validasi terhadap <i>username</i> dan <i>password</i> yang dimasukan oleh <i>user</i> .
Deskripsi	
Aktor	<i>User</i>
Skenario Utama	
Kondisi Awal	<i>Form Login</i> sudah tersedia
Aksi Aktor	Reaksi Sistem
1. Memasukkan <i>username</i> dan <i>password</i>	2. <i>Form Login</i> akan menampilkan <i>textbox</i> <i>username</i> , <i>password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ‘•’ pada label untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap <i>username</i> , <i>password</i> yang telah dimasukkan dengan menekan tombol <i>Ok</i> .	4. Aplikasi melakukan validasi terhadap <i>username</i> , <i>password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengecekan pada basis data.
Kondisi Akhir	Jika pada akhir interaksi <i>username</i> , <i>password</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman

	utama dan dapat menggunakan aplikasi sesuai hak aksesnya.
--	---

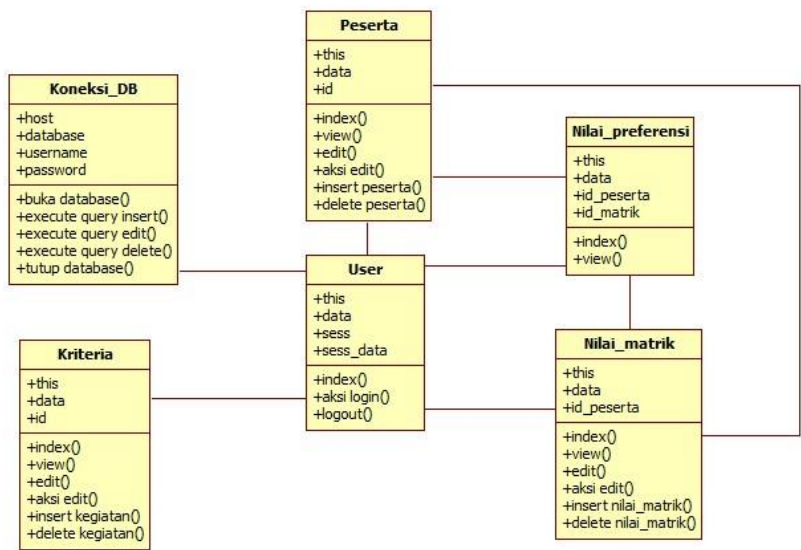
**Alur Alternatif kondisi akhir**

a. Jika tidak *valid* maka Aplikasi akan kembali ke tampilan awal *login*.

5.1.2.4.2 *Class Diagram*

*Class Diagram* menggambarkan *class* dan hubungan antar *class* didalam sistem. *Class* digambarkan dengan sebuah kotak dibagi menjadi tiga bagian. Bagian paling atas diisikan nama *class*, bagian tengah diisikan variabel yang dimiliki *class*, dan bagian bawah diisikan *method-method* dari *class*.

Dalam proses pembuatan aplikasi ini *class diagram* yang akan digunakan adalah sebagai berikut :



### *Gambar 5.3 Class Diagram*

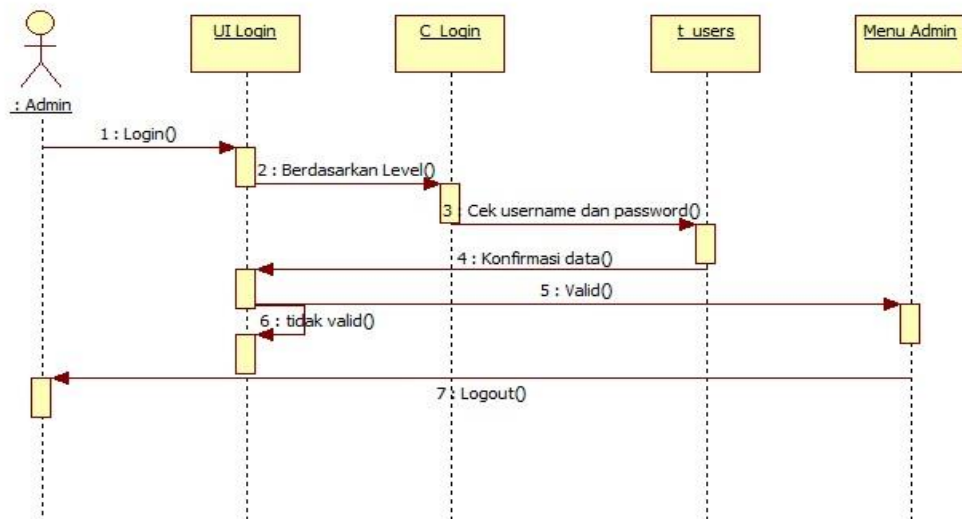
#### 5.1.2.4.3 Interaction Diagram

*Interaction diagram* adalah diagram yang menggambarkan interaksi dalam sistem, antara aplikasi, *user* dan *database*. *Interaction diagram* didefinisikan dengan dua tipe diagram yaitu *sequence diagram* dan *collaboration diagram*.

#### 5.1.2.4.4 Sequence Diagram

*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara objek-objek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh objek-objek yang melakukan suatu tugas atau aksi tertentu. Objek-objek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram, pada diagram ini, dimensi vertikal merepresentasikan waktu.

#### 5.1.2.4.4.1 Sequence Diagram Login Admin

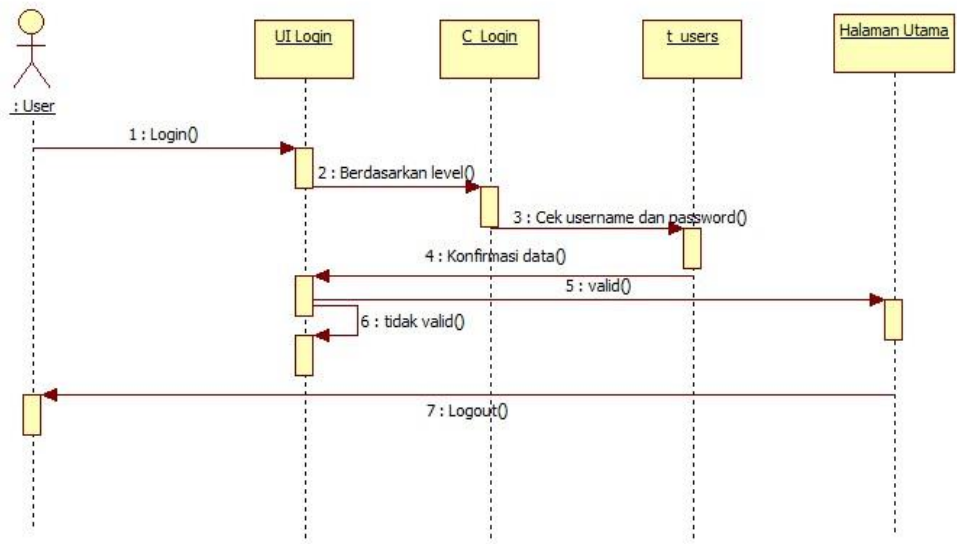


Gambar 5.4 Sequence Diagram Login Admin

Penjelasan :

*Sequence diagram* ini menjelaskan proses *login*. *Admin* mulai menjalankan aplikasi akan tampil *form login*. Kemudian *admin* memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *user*. Jika sesuai maka akan mendapatkan tampilan halaman utama *admin*.

#### 5.1.2.4.4.2 Sequence Diagram Login User

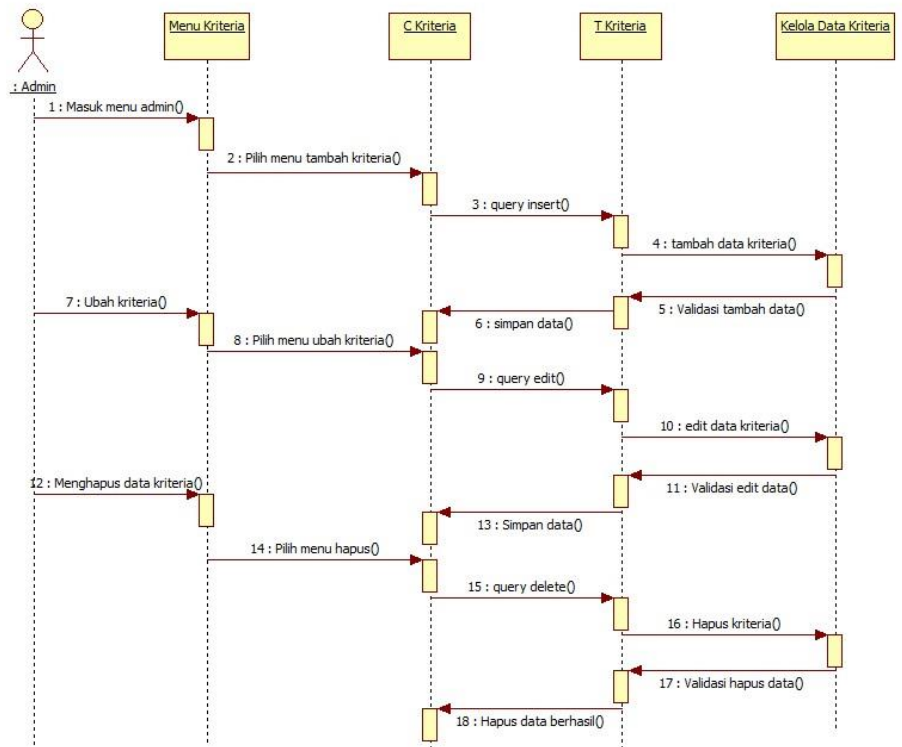


Gambar 5.5 Sequence Diagram Login User

Penjelasan :

*Sequence diagram* ini menjelaskan proses *login*. *User* mulai menjalankan aplikasi akan tampil *form login*. Kemudian *user* memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *user*. Jika sesuai maka *user* dapat melakukan pendaftaran kegiatan akademik.

#### 5.1.2.4.4.3 Sequence Diagram Kelola Data Kriteria



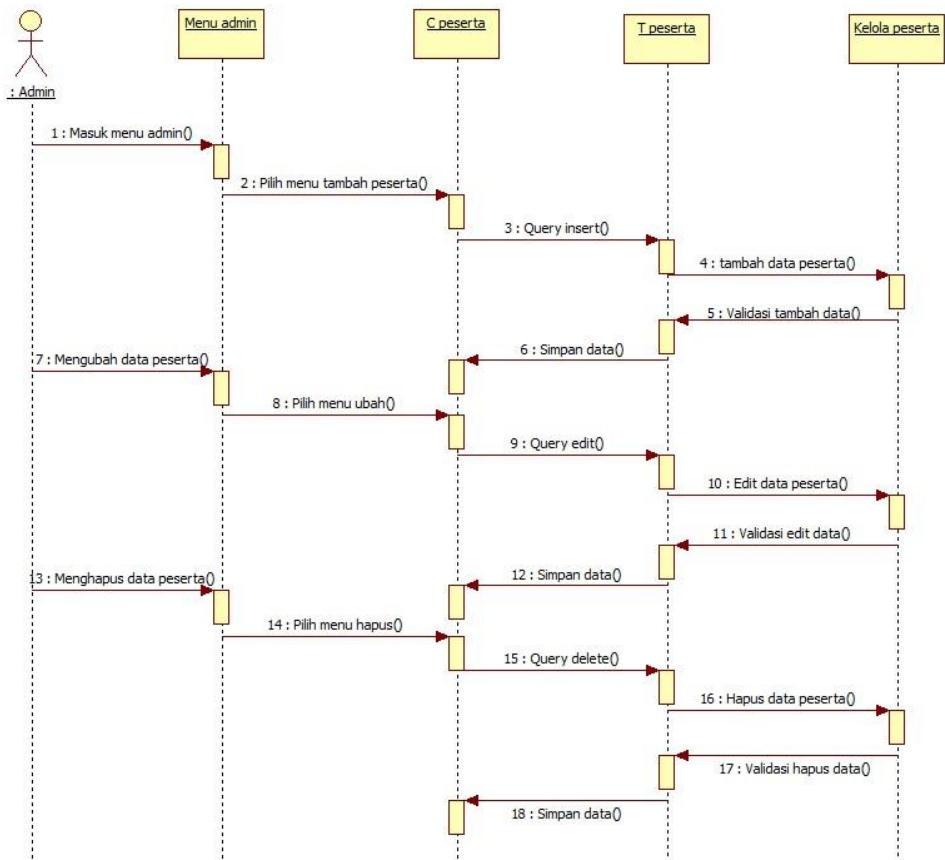
Gambar 5.6 Sequence Diagram Kelola Data Kriteria

Penjelasan :

*Sequence diagram* ini menjelaskan proses kelola data kriteria. Admin mulai menjalankan aplikasi akan tampil *form login*. Kemudian admin memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *database*. Jika sesuai maka admin dapat melakukan pengelolaan data kegiatan dengan *meng input*, ubah dan hapus data pada form tersebut.



#### 5.1.2.4.4 Sequence Diagram Kelola Data Peserta



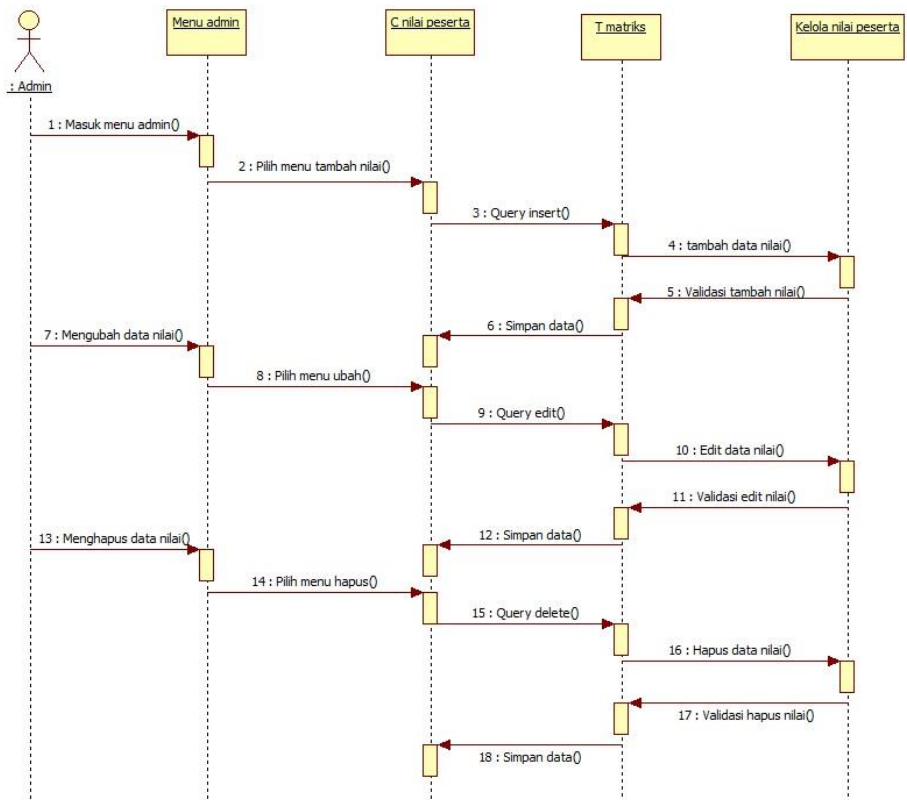
Gambar 5.7 Sequence Diagram Kelola Data Peserta

Penjelasan :

*Sequence diagram* ini menjelaskan proses kelola data peserta. Aktor memilih menu kelola data peserta. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Maka sistem memuat tampilan data

peserta, pada tampilan ini aktor dapat melakukan *input*, ubah dan hapus.

#### 5.1.2.4.4.5 Sequence Diagram Kelola Pemilihan Peserta



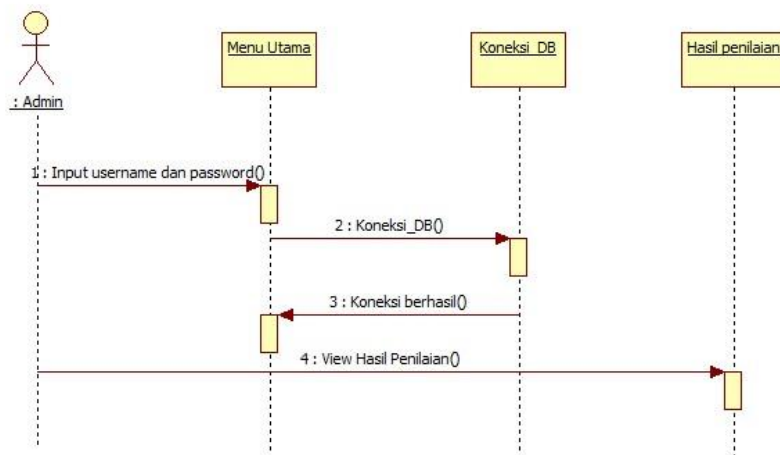
Gambar 5.8 Sequence Diagram Kelola Data Pemilihan Peserta

Penjelasan :

*Sequence diagram* ini menjelaskan proses kelola data pemilihan peserta. Aktor memilih menu kelola data pemilihan peserta. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi

*database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data pemilihan peserta, pada tampilan ini aktor dapat melakukan *input* nilai peserta, ubah dan hapus.

#### 5.1.2.4.4.6 *Sequence Diagram* Admin Hasil Penilaian

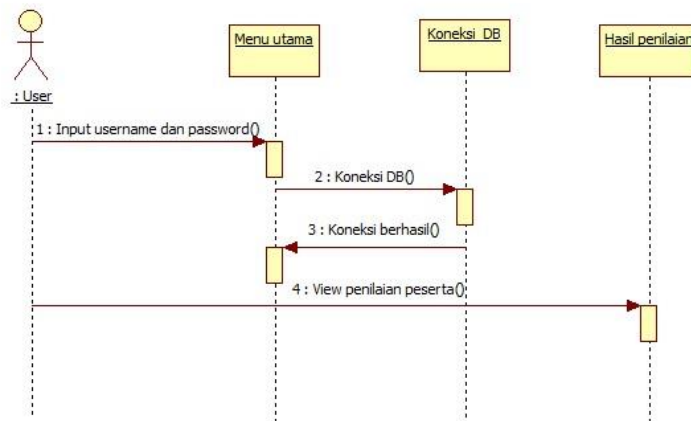


Gambar 5.9 *Sequence Diagram* Admin Hasil Penilaian

Penjelasan :

*Sequence diagram* ini menjelaskan proses hasil penilaian peserta. Aktor memilih menu hasil penilaian. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data hasil penilaian peserta, pada tampilan ini aktor dapat melihat hasil penilaian dari perhitungan sebelumnya.

#### 5.1.2.4.4.7 Sequence Diagram User Hasil Penilaian

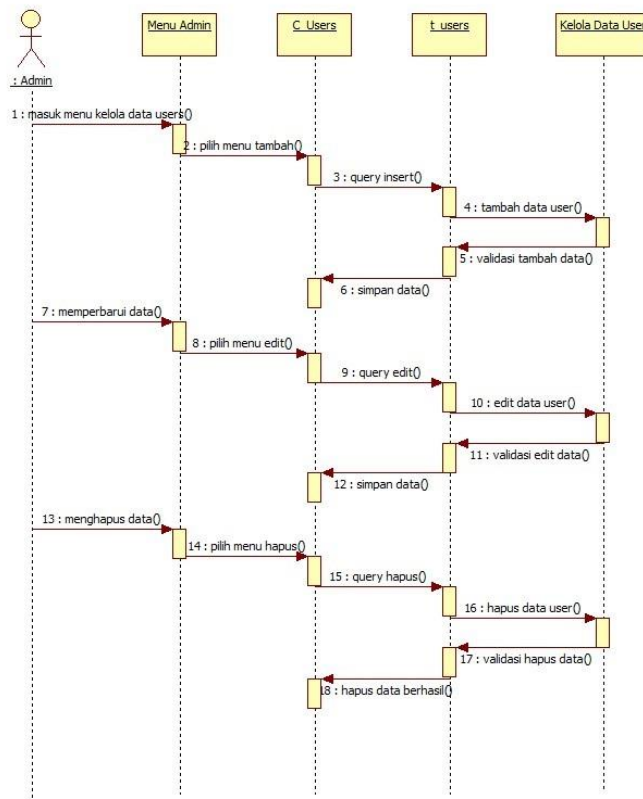


Gambar 5.10 Sequence Diagram Admin Hasil Penilaian

Penjelasan :

*Sequence diagram* ini menjelaskan proses hasil penilaian peserta. Aktor memilih menu hasil penilaian. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data hasil penilaian peserta, pada tampilan ini aktor dapat melihat hasil penilaian dari perhitungan sebelumnya.

#### 5.1.2.4.4.7 Sequence Diagram User Hasil Penilaian



Gambar 5. 11 Sequence Diagram Kelola Data User

Penjelasan :

*Sequence diagram* ini menjelaskan proses pengelolaan *user*. Aktor memilih menu *user*. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data *user*, pada tampilan ini aktor dapat menambah, mengubah dan menghapus data *user*.

## **Perancangan *User Interface***

### **5.2.1 *User Interface Login***

ACCOUNT LOGIN

Username	Password
<b>SIGN IN</b>	
KEMBALI	

Gambar 5.12 *User Interface Login*

Keterangan Gambar :

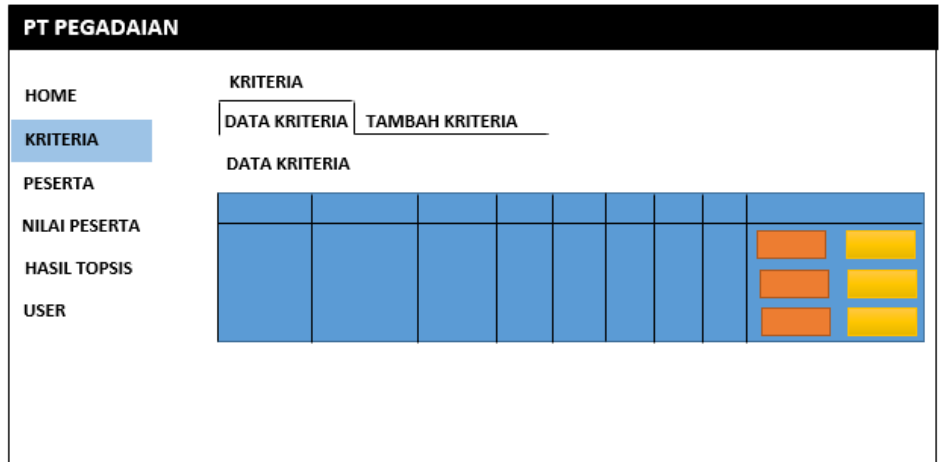
*Textbox 1* : *Input Username*

*Textbox 2* : *Input Password*

*Command Button 1* : *Sign In*

*Command Button 1* : *Kembali*

### 5.2.2 User Interface Kelola Data Kriteria



Gambar 5.13 User Interface Data Kegiatan

Keterangan Gambar :

Label 1 : Kegiatan

Label 2 : Data Kegiatan

Label 3 : Tambah Kegiatan

Table 1 : Tabel Data Kegiatan

### 5.2.3 User Interface Tambah Data Kegiatan

The screenshot displays a web application interface for PT PEGADAIAN. On the left is a vertical sidebar menu with the following items: HOME, KRITERIA (highlighted in blue), PESERTA, NILAI PESERTA, HASIL TOPSIS, and USER. The main content area is titled 'KRITERIA' and contains two sub-sections: 'DATA KRITERIA' and 'TAMBAH KRITERIA'. The 'TAMBAH KRITERIA' section includes several input fields: 'KODE', 'KRITERIA', 'JUDUL KEGIATAN', 'BOBOT', 'POINT 1', 'PONIT 2', 'POINT 3', 'POINT 4', 'POINT 5', and 'SIFAT KRITERIA'. A blue button is located at the bottom of this section.

Gambar 5.14 User Interface Tambah Data Kriteria

Keterangan Gambar :

Label 1 : Kriteria

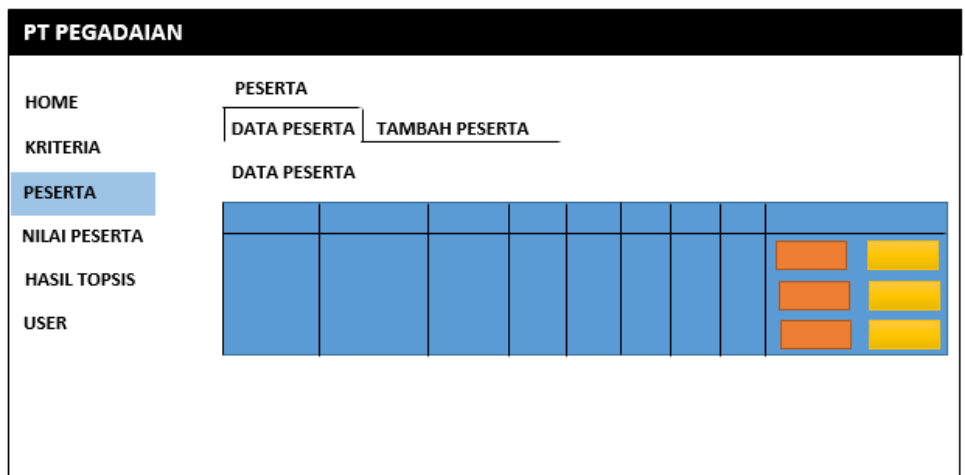
Label 2 : Data Kriteria

Label 3 : Tambah Kriteria



*Textbox 1* : Tambah Data Kriteria  
*Textbox 2* : Tambah Data Kriteria  
*Textbox 3* : Tambah Data Kriteria  
*Textbox 4* : Tambah Data Kriteria  
*Textbox 5* : Tambah Data Kriteria  
*Button 1* : Simpan

#### 5.2.4 User Interface Data Peserta



Gambar 5.15 User Interface Data Peserta

Keterangan Gambar :

*Label 1* : Peserta  
*Label 2* : Data Peserta  
*Label 3* : Tambah Peserta  
*Tabel 1* : Tabel Data Peserta

### 5.2.5 User Interface Tambah Data Peserta

The screenshot shows a web application interface for PT PEGADAIAN. On the left is a vertical navigation menu with the following items: HOME, KRITERIA, PESERTA (highlighted in blue), NILAI PESERTA, HASIL TOPSIS, and USER. The main content area is titled 'PESERTA' and contains two sub-sections: 'DATA PESERTA' and 'TAMBAH PESERTA'. The 'TAMBAH PESERTA' section includes four text input fields labeled 'ID', 'NAMA', 'JUDUL KEGIATAN', and 'TANGGAL'. Below these fields is a blue button, likely for saving the data.

Gambar 5. 16 User Interface Tambah Data Peserta

Keterangan Gambar :

*Label 1* : Peserta

*Label 2* : Data Peserta

*Label 3* : Tambah Peserta

*Textbox* : Data Peserta

*Button 1* : Simpan

### 5.2.6 User Interface Data Penilaian

**PT PEGADAIAN**

HOME  
**KRITERIA**  
PESERTA  
NILAI PESERTA  
HASIL TOPSIS  
USER

**NILAI PESERTA**

ISI NILAI PESERTA

TAMBAH NILAI PESERTA

PESERTA

GRADE	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
SERTIFIKASI	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
KEIKUTSERTAAN	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TEMA	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
KEDISIPLINAN	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**Simpan**

*Gambar 5.17 User Interface Data Penilaian*

Keterangan Gambar :

*Label 1* : Penilaian

*Label 2* : Data Nilai

*Alternatif Box 1* : ID Peserta

*Textbox* : Data Peserta

*Radio Button* : Angka Nilai

*Button 1* : Simpan

### 5.2.7 User Interface Hasil TOPSIS Penilaian

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

USER

HASIL TOPSIS

NILAI PESERTA

NILAI TERNORMALISASI NILAI BOBOT TERNORMALISASI

MATRIKS IDEAL POSITIF/NEGATIF

JARAK SOLUSI IDEAL POSITIF/NEGATIF

NILAI PREFERENSI

NILAI PESERTA


Gambar 5.18 User Interface Hasil Topsis Penilaian

Keterangan Gambar :

Label 1 : Penilaian

Tabel 1 : Tabel Penilaian

### 5.2.8 User Interface Hasil Data Penilaian Ternormalisasi

PT PEGADAIAN						
HOME	HASIL TOPSIS					
KRITERIA	NILAI PESERTA	NILAI TERNORMALISASI			NILAI BOBOT TERNORMALISASI	
PESERTA	MATRIKS IDEAL POSITIF/NEGATIF    JARAK SOLUSI IDEAL POSITIF/NEGATIF    NILAI PREFERENSI					
	NILAI TERNORMALISASI					
NILAI PESERTA						
HASIL TOPSIS						
USER						

Gambar 5.19 User Interface Hasil Data Penilaian Ternormalisasi

Keterangan Gambar :

*Label 1* : Data Penilaian Ternormalisasi

*Tabel 1* : Tabel Penilaian Ternormalisasi

### 5.2.9 User Interface Hasil Data Penilaian Bobot

Ternormalisasi

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

USER

HASIL TOPSIS

NILAI PESERTA

NILAI TERNORMALISASI

NILAI BOBOT TERNORMALISASI

MATRIKS IDEAL POSITIF/NEGATIF

JARAK SOLUSI IDEAL POSITIF/NEGATIF

NILAI PREFERENSI

NILAI BOBOT TERNORMALISASI


*Gambar 5.20 User Interface Hasil Data Penilaian Bobot Ternormalisasi*

Keterangan Gambar :

*Label 1* : Data Penilaian Bobot Ternormalisasi

*Tabel 1* : Tabel Penilaian Bobot Ternormalisasi

### 5.2.10 User Interface Nilai Ideal Positif/Negatif

PT PEGADAIAN			
HOME	HASIL TOPSIS		
KRITERIA	NILAI PESERTA	NILAI TERNORMALISASI	NILAI BOBOT TERNORMALISASI
PESERTA	MATRIKS IDEAL POSITIF/NEGATIF	JARAK SOLUSI IDEAL POSITIF/NEGATIF	NILAI PREFERENSI
NILAI PESERTA	MATRIKS IDEAL POSITIF A <sup>+</sup>		
HASIL TOPSIS			
USER			
	MATRIKS IDEAL NEGATIF A <sup>-</sup>		

Gambar 5.21 User Interface Nilai Ideal Positif/Negatif

Keterangan Gambar :

Label 1 : Nilai Ideal Positif/Negatif

Label 2 : Nilai Ideal Positif A<sup>+</sup>

Label 1 : Nilai Ideal Negatif A<sup>-</sup>

Tabel 1 : Tabel Penilaian Ternormalisasi

Tabel 2 : Tabel Penilaian Ternormalisasi

### 5.2.11 User Interface Jarak Solusi Ideal Positif/Negatif

PT PEGADAIAN

HOME
KRITERIA
PESERTA
NILAI PESERTA
HASIL TOPSIS
USER

HASIL TOPSIS
NILAI PESERTA
NILAI TERNORMALISASI
NILAI BOBOT TERNORMALISASI
Matriks Ideal Positif/Negatif
Jarak Solusi Ideal Positif/Negatif
Nilai Preferensi

Jarak Solusi Ideal Positif D<sup>+</sup>

Jarak Solusi Ideal Negatif D<sup>-</sup>

Gambar 5.29 User Interface Jarak Solusi Ideal Positif/Negatif

Keterangan Gambar :

Label 1 : Jarak Solusi Ideal Positif/Negatif

Label 2 : Jarak Solusi Ideal Positif D<sup>+</sup>

Label 1 : Jarak Solusi Ideal Negatif D<sup>-</sup>

Tabel 1 : Tabel Penilaian Ternormalisasi

Tabel 2 : Tabel Penilaian Ternormalisasi

### 5.2.12 User Interface Nilai Preferensi

PT PEGADAIAN

HOME

KRITERIA

PESERTA

**NILAI PESERTA**

HASIL TOPSIS

USER

HASIL TOPSIS

NILAI PESERTA

NILAI TERNORMALISASI

NILAI BOBOT TERNORMALISASI

MATRIKS IDEAL POSITIF/NEGATIF

JARAK SOLUSI IDEAL POSITIF/NEGATIF

NILAI PREFERENSI

NILAI PREFERENSI


Gambar 5.22 User Interface Nilai Preferensi

Keterangan Gambar :

Label 1 : Nilai Preferensi

Tabel 1 : Tabel Nilai Preferensi

### 5.2.13 User Interface Kelola Data User

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

**USER**

USER

DATA USER

TAMBAH USER

DATA USER

			<div></div> <div></div>
			<div></div> <div></div>
			<div></div> <div></div>

Gambar 5.23 User Interface Kelola Data User



Keterangan Gambar :

*Label 1* : User

*Label 2* : Data User

*Label 3* : Tambah User

*Tabel 1* : Tabel Data User

#### 5.2.14 User Interface Tambah Data User

The screenshot displays the PT PEGADAIAN user interface. On the left is a sidebar menu with options: HOME, KRITERIA, PESERTA, NILAI PESERTA, HASIL TOPSIS, and USER (which is highlighted in blue). The main content area has a header 'PT PEGADAIAN' and a sub-header 'USER'. Below the sub-header are two tabs: 'DATA USER' and 'TAMBAH USER'. The 'TAMBAH USER' tab is selected. Under this tab, there is a section titled 'TAMBAH USER' containing three input fields labeled 'NAMA', 'USERNAME', and 'PASSWORD'. A blue button is positioned below the 'PASSWORD' field.

*Gambar 5.24 User Interface Tambah Data User*

Keterangan Gambar :

*Label 1* : User

*Label 2* : Data User

*Label 3* : Tambah User

*Tabel 1* : Tabel Tambah User

## 5.3 Perancangan Arsitektur Sistem

### 5.3.1 Perancangan Arsitektur Perangkat Lunak Sistem

Perangkat Lunak yang digunakan dalam Sistem Implementasi Pemilihan Karyawan Pada Penjadwalan Akademik di PT Pegadaian (Persero) adalah sebagai berikut:

Tabel 5.6 Arsitektur Perangkat Lunak Sistem

No.	Tools/Software	Fungsi	Keterangan
1	<i>Windows 8 (min)</i>	Sistem Operasi	-
2	<i>Bizagi Modeller</i>	Pembuatan Flow Map	-
3	<i>Google Chrome</i>	Aplikasi web browser	-
4	<i>Star UML</i>	Pembuatan Diagram UML	-
5	<i>MySQL</i>	DBMS	-
6	<i>Microsoft Word</i>	Document	-
7	<i>Visual Studio Code</i>	Text Editor	-

### 5.3.2 Perancangan Arsitektur Perangkat Keras Sistem

Perangkat keras yang digunakan dalam Sistem Implementasi Pemilihan Peserta Pada Penjadwalan Kegiatan Akademik di PT Pegadaian ini adalah sebagai berikut :

Tabel 5.7 Arsitektur Perangkat Keras Sistem

No	Nama Perangkat	Spesifikasi	Keterangan
1	<i>Memory</i>	4 GB	<i>Memory System</i> yang digunakan
2	<i>Processor</i>	<i>Intel® core™ CPU</i>	Untuk kecepatan <i>transfer</i> data dari sistem yang sangat bergantung pada kecepatan <i>processor</i> komputer
3	<i>System Type</i>	64-bit <i>Operating System</i> , <i>x64-Based Processor</i>	

## 5.4 Pemetaan Struktur *Diagram User/Aktor Sistem*

Tabel 5. 8 Struktur *Diagram User/Aktor Sistem*

<b>N o</b>	<b>Aktor</b>	<b>Keterangan</b>
1	Admin	Melakukan kelola data kegiatan, , kelola data peserta, dan kelola data pemilihan peserta
2	User	Melakukan login dan View Data Pemilihan Peserta

## 5.5 Pengujian Metode

Tabel 5.1 Data Kriteria Karyawan

Kriteria	Bobot	Point 1	Point 2	Point 3	Point 4	Point 5	Sifat Kriteria
<i>Grade</i>	20	1	2	3	4	5	Benefit
Kesesuaian Proker Corpu	20	1	2	3	4	5	Benefit
Keikutsertaan	15	1	2	3	4	5	Cost
Tema	25	1	2	3	4	5	Benefit
Sharing	20	1	2	3	4	5	Benefit

Tabel 5.2 Nilai Peserta

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin</i>
Eka Pebriansyah	5	4	1	5	5
Supriyanto	5	5	3	5	5
Endang Pertiwi	5	4	1	5	5
Muh Ariyadi	5	4	1	5	5
Hertin Maulida	5	4	1	5	5
Elvi Rofiqatul Hidayah	5	5	4	5	5
Teguh Budi Ismanto	5	4	2	5	5
Eri Mardianto	5	4	2	5	5

Sriyani	5	4	1	5	5
---------	---	---	---	---	---

Tabel 5.3 Nilai Peserta Ternormalisasi

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin g</i>
Eka Pebriansyah	0.333	0.314	0.162	0.333	0.333
Supriyanto	0.333	0.393	0.487	0.333	0.333
Endang Pertiwi	0.333	0.314	0.162	0.333	0.333
Muh Ariyadi	0.333	0.314	0.162	0.333	0.333
Hertin Maulida	0.333	0.393	0.649	0.333	0.333
Elvi Rofiqatul Hidayah	0.333	0.314	0.324	0.333	0.333
Teguh Budi Ismanto	0.333	0.314	0.324	0.333	0.333

Eri Mardianto	0.33 3	0.314	0.162	0.33 3	0.333
Sriyani	0.33 3	0.314	0.162	0.33 3	0.333

Tabel 5.4 Nilai Peserta bobot Ternormalisasi

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tem a	<i>Sharin g</i>
Eka Pebriansyah	6.66 7	4	1	8.33 3	6.667
Supriyanto	6.66 7	6.285	2.433	8.33 3	6.667
Endang Pertiwi	6.66 7	7.857	7.3	8.33 3	6.667
Muh Ariyadi	6.66 7	6.285	2.433	8.33 3	6.667
Hertin Maulida	6.66 7	6.285	2.433	8.33 3	6.667
Elvi Rofiqatul Hidayah	6.66 7	7.857	9.733	8.33 3	6.667

Teguh Budi Ismanto	6.66 7	6.285	4.867	8.33 3	6.667
Eri Mardianto	6.66 7	6.285	4.867	8.33 3	6.667
Sriyani	6.66 7	6.285	4.867	8.33 3	6.667

Tabel 5.5 Nilai Ideal Positif

<i>Alternatif</i>	A+
Grade	6.667
Kesesuaian dengan Proker	7.857
Keikutsertaan	9.733
Tema	8.333
Sharing	6.667



Tabel 5.6 Nilai Ideal Negatif

<i>Alternatif</i>	D+
Grade	6.667
Kesesuaian dengan Proker	6.285
Keikutsertaan	2.433
Tema	8.333
Sharing	6.667

Tabel 5.7 Jarak Solusi Ideal Positif

<i>Alternatif</i>	D+
Eka Pebriansyah	7.467
Supriyanto	2.433
Endang Pertiwi	7.467
Muh Ariyadi	7.467
Hertin Maulida	7.467

Elvi Rofiqatul Hidayah	0
Teguh Budi Ismanto	5.114
Eri Mardianto	5.114
Sriyani	7.467

Tabel 5.8 Jarak Solusi Ideal Negatif

<i>Alternatif</i>	D+
Eka Pebriansyah	0
Supriyanto	5.115
Endang Pertiwi	0
Muh Ariyadi	0
Hertin Maulida	0
Elvi Rofiqatul Hidayah	7.467
Teguh Budi Ismanto	2.434

Eri Mardianto	2.434
Sriyani	0

Tabel 5. 9 Nilai Preferensi

Nama	Preferensi
Eka Pebriansyah	0
Supriyanto	0.6777
Endang Pertiwi	0
Muh Ariyadi	0
Hertin Maulida	0
Elvi Rofiqatul Hidayah	1
Teguh Budi Ismanto	0.3225
Eri Mardianto	0.3225
Sriyani	0

Pada tahap pengujian metode ini, dilakukan pengujian metode untuk mengetahui apakah metode Topsis dapat menghasilkan pemilihan alternatif terbaik untuk karyawan dengan proses pengolahan data menggunakan bahasa pemrograman PHP.

Berikut adalah hasil pengujian metode Topsis :

Nilai peserta

Nilai Berdasarkan Tanggal : <input type="text" value="dd/mm/yyyy"/>		<input type="button" value="Search"/>		Nilai Berdasarkan Kegiatan : <input type="text" value="- Pilih -"/>		<input type="button" value="Search"/>	
No	Nama	kriteria					
		C1	C2	C3	C4	C5	
1	Eka Pebriansyah	5	4	1	5	5	
2	Supriyanto	5	5	3	5	5	
3	Endang Pertiwi	5	4	1	5	5	
4	Muh Ariyadi	5	4	1	5	5	
5	Hertin Maulida	5	4	1	5	5	
6	Elvi Rofiqatul Hidayah	5	5	4	5	5	
7	Teguh Budi Ismanto	5	4	2	5	5	
8	Eri Mardianto	5	4	2	5	5	
9	Sriyani	5	4	1	5	5	

Gambar 6.1 Nilai Peserta

Nilai peserta ternormalisasi

Nilai Berdasarkan Tanggal : dd/mm/yyyy		Search		Nilai Berdasarkan Kegiatan : - Pilih -		Search	
No	Nama	kriteria					
		C1	C2	C3	C4	C5	
1	Eka Pebriansyah	0.333	0.314	0.162	0.333	0.333	
2	Supriyanto	0.333	0.393	0.487	0.333	0.333	
3	Endang Pertiwi	0.333	0.314	0.162	0.333	0.333	
4	Muh Ariyadi	0.333	0.314	0.162	0.333	0.333	
5	Hertin Maulida	0.333	0.314	0.162	0.333	0.333	
6	Elvi Rofiqatul Hidayah	0.333	0.393	0.649	0.333	0.333	
7	Teguh Budi Ismanto	0.333	0.314	0.324	0.333	0.333	
8	Eri Mardianto	0.333	0.314	0.324	0.333	0.333	
9	Sriyani	0.333	0.314	0.162	0.333	0.333	

Gambar 6.2 Nilai Peserta Ternormalisasi

Nilai Bobot Ternormalisasi

Nilai Berdasarkan Tanggal : <input type="text" value="dd/mm/yyyy"/>		<input type="text" value="Search"/>		Nilai Berdasarkan Kegiatan : <input type="text" value="- Pilih -"/>		<input type="text" value="Search"/>	
No	Nama	kriteria					
		C1	C2	C3	C4	C5	
1	Eka Pebriansyah	6.667	6.285	2.433	8.333	6.667	
2	Supriyanto	6.667	7.857	7.3	8.333	6.667	
3	Endang Pertiwi	6.667	6.285	2.433	8.333	6.667	
4	Muh Ariyadi	6.667	6.285	2.433	8.333	6.667	
5	Hertin Maulida	6.667	6.285	2.433	8.333	6.667	
6	Elvi Rofiqatul Hidayah	6.667	7.857	9.733	8.333	6.667	
7	Teguh Budi Ismanto	6.667	6.285	4.867	8.333	6.667	
8	Eri Mardianto	6.667	6.285	4.867	8.333	6.667	
9	Sriyani	6.667	6.285	2.433	8.333	6.667	

Gambar 6.3 Nilai Peserta Terbobot Ternormalisasi

Matriks Ideal Positif ( $A^+$ )

Nilai Berdasarkan Tanggal :   Nilai Berdasarkan Kegiatan :

kriteria				
kr002	kr003	kr004	kr005	kr006
$y_1^+$	$y_2^+$	$y_3^+$	$y_4^+$	$y_5^+$
6.667	7.857	9.733	8.333	6.667

Gambar 6.4 Nilai Ideal Positif

Matriks Ideal Positif ( $A^-$ )

kriteria				
kr002	kr003	kr004	kr005	kr006
$y_1^-$	$y_2^-$	$y_3^-$	$y_4^-$	$y_5^-$
6.667	6.285	2.433	8.333	6.667

Gambar 6.5 Nilai Ideal Negatif

Jarak Solusi Ideal Positif ( $D^+$ )

Nilai Berdasarkan Tanggal :   Nilai Berdasarkan Kegiatan :

Nomor	Nama	$D^+$
1	Eka Pebriansyah	7.467
2	Supriyanto	2.433
3	Endang Pertiwi	7.467
4	Muh Ariyadi	7.467
5	Hertin Maulida	7.467
6	Elvi Rofiqatul Hidayah	0
7	Teguh Budi Ismanto	5.114
8	Eri Mardianto	5.114
9	Sriyani	7.467

Gambar 6.6 Nilai Jarak Solusi Ideal Positif

Jarak Solusi Ideal Negatif ( $D^-$ )

Nomor	Nama	$D^-$
1	Eka Pebriansyah	0
2	Supriyanto	5.115
3	Endang Pertiwi	0
4	Muh Ariyadi	0
5	Hertin Maulida	0
6	Elvi Rofiqatul Hidayah	7.467
7	Teguh Budi Ismanto	2.434
8	Eri Mardianto	2.434
9	Sriyani	0

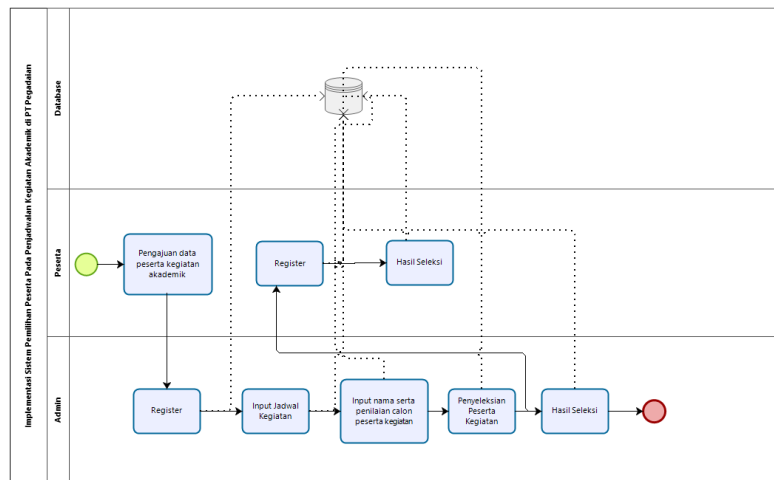
Gambar 6.7 Nilai Jarak Solusi Ideal Negatif

Nilai Preferensi

Nomor	Nama	$V_i$
1	Elvi Rofiqatul Hidayah	1
2	Supriyanto	0.6777
3	Teguh Budi Ismanto	0.3225
4	Eri Mardianto	0.3225
5	Eka Pebriansyah	0
6	Endang Pertiwi	0
7	Muh Ariyadi	0
8	Hertin Maulida	0
9	Sriyani	0

Gambar 6.8 Nilai Preferensi

Dalam pembuatan suatu sistem diperlukan adanya sebuah perancangan sistem yang akan dibuat, berikut merupakan salah satu perancangan sistem yang akan dibangun :



Gambar 4.25 Analisis Prosedur Yang Akan Dibangun

Prosedur yang akan dibangun :

7. Peserta melakukan pengajuan data peserta kegiatan akademik kepada admin.
8. Admin melakukan register untuk menginputkan jadwal kegiatan peserta yang akan mengikuti kegiatan akademik.
9. Setelah menginputkan jadwal admin akan menginputkan nama calon peserta kegiatan.
10. Admin melakukan penyeleksian peserta kegiatan berdasarkan nilai-nilai yang telah didapatkan dari kriteria perusahaan yang diinputkan.

11. Setelah proses penyeleksian maka akan keluar hasil berdasarkan nilai yang diinputkan.
12. Peserta akan melakukan register untuk melihat hasil pemilihan peserta kegiatan.

#### 4.3 Analisis Dokumen Yang Akan Dibangun

Di dalam sistem yang saat ini berjalan terdapat beberapa dokumen yang digunakan sebagai arsip untuk menyimpan data di pos market. Adapun dokumen yang dimaksud adalah sebagai berikut :

Tabel 4.9 Dokumen Admin

<b>Dibuat Oleh</b>	Admin
<b>Dibuat Untuk</b>	Admin
<b>Isi</b>	Berupa jadwal kegiatan akademik dan kuota peserta, pemilihan karyawan dalam kegiatan akademik
<b>Frekuensi</b>	Dibuat form pemilihan peserta akademik
<b>Tujuan</b>	Pendataan peserta, jumlah peserta dan nama peserta kegiatan

Tabel 4.10 Dokumen *User*

<b>Dibuat oleh</b>	Admin
--------------------	-------



<b>Dibuat untuk</b>	User
<b>Isi</b>	Berupa form hasil penilaian peserta kegiatan akademik
<b>Frekuensi</b>	Dibuat form untuk melihat hasil penilaian
<b>Tujuan</b>	Melihat penilaian peserta kegiatan akademik

#### 4.1.2.3 Analisis Metode TOPSIS

Metode ini merupakan metode kompromi yang banyak digunakan untuk menangani berbagai kriteria masalah pengambilan keputusan. Dasar pemikiran metode TOPSIS adalah bahwa alternatif yang dipilih harus memiliki jarak terdekat solusi positif ideal dan jarak terjauh dari solusi *negative* ideal. Bobot kriteria dalam metode TOPSIS diberikan apriori. Selain itu, solusi positif-ideal dan solusi *negative* ideal langsung dihasilkan dari matriks keputusan.

Langkah-langkah metode TOPSIS dapat dilakukan seperti yang sudah dijelaskan pada BAB II . Berikut merupakan tahap penerapan metode TOPSIS :

## 2. Data Kriteria dan Bobot

No	Kriteria	Atribut	Bobot
----	----------	---------	-------

1	Grade	Benefit	20
2	Kesesuaian Program kerja Corpu	Benefit	20
3	Keikutsertaan	Cost	15
4	Tema (Jobdesk)	Benefit	25
5	Sharing	Benefit	20

Deskripsi Bobot beserta Nilai :

- f. Grade (20) = Untuk meningkatkan *performance* mengenai kesesuaian dengan tema
- g. Kesesuaian Proker Corpu (20) = Untuk mendukung program kerja dari Coporate University
- h. Keikutsertaan (15) = Mengenai keadilan keikutsertaan karyawan dalam kegiatan pertahun
- i. Tema (Jobdesk) (25) = Untuk meningkatkan kompetensi karyawan sehingga diharapkan dapat meningkatkan *Performance* karyawan dalam bekerja.
- j. Sharing (20) = Materi yang telah diseminarkan harus dibagikan untuk menghidupkan sistem KMS (*Knowledge Management System*) PT Pegadaian (Persero).

## 2. Data Matriks Berdasarkan Kriteria

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin</i> g
Eka Pebriansyah	5	4	1	5	5
Supriyanto	5	5	3	5	5
Endang Pertiwi	5	4	1	5	5
Muh Ariyadi	5	4	1	5	5
Hertin Maulida	5	4	1	5	5
Elvi Rofiqatul Hidayah	5	5	4	5	5
Teguh Budi Ismanto	5	4	2	5	5
Eri Mardianto	5	4	2	5	5
Sriyani	5	4	1	5	5

Deskripsi Penilaian Perusahaan :

6. *Grade* dengan Nilai
  - f. Tenaga Pembantu (1)
  - g. Tenaga Keamanan (2)
  - h. Karyawan Grade 4-7 (3)
  - i. Karyawan Grade 8-11 (4)
  - j. Karyawan Grade 12-15 (5)
7. Kesesuaian Program kerja Corpu dengan Nilai  
Nilai rata-rata dari penilaian grade,tema,*sharing*, dan keikutsertaan
8. Keikutsertaan dengan Nilai
  - b. (1-5) Berdasarkan banyaknya keikutsertaan peserta dalam jangka waktu setahun
9. Tema dengan Nilai
  - f. Keterkaitan dengan bagian tersebut Grade 4-7 (1)
  - g. Keterkaitan dengan bagian tersebut Grade 8-11 (2)
  - h. Keterkaitan dengan jobdesk dan seminar Grade 4-7 (3)
  - i. Keterkaitan dengan jobdesk dan seminar Grade 8-11 (4)
  - j. Keterkaitan dengan jobdesk dan seminar Grade 12-15 (5)
10. *Sharing* (Membagikan Materi Seminar di KMS) dengan Nilai

- f. Grade 1 (1)
- g. Grade 2-3 (2)
- h. Grade 4-7 (3)
- i. Grade 8-11 (4)
- j. Grade 12-15 (5)

3. Pemberian bobot pada masing-masing peserta, kemudian pencarian nilai dengan rumus sebagai berikut :

$$|X_n| = \sqrt{\sum_{i=1}^m x_{ij}^2}$$

$$|X_1| = \sqrt{(5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2} = 15,0000$$

$$|X_2| = \sqrt{(4)^2 + (5)^2 + (4)^2 + (4)^2 + (4)^2 + (5)^2 + (4)^2 + (4)^2} = 12,7279$$

$$|X_3| = \sqrt{(1)^2 + (3)^2 + (1)^2 + (1)^2 + (1)^2 + (4)^2 + (2)^2 + (2)^2 + (1)^2} = 6,1644$$

$$|X_4| = \sqrt{(5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2} = 15,0000$$

$$|X_5| = \sqrt{(5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2 + (5)^2} = 15,0000$$

8. Setelah rumus  $X_n$  maka selanjutnya menggunakan rumus rij

:

$$r_{ij} = \frac{x^{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}$$

R <sub>10</sub> = 5/15,0000 = 0,333	R <sub>20</sub> = 4/12,7279 = 0,314	R <sub>30</sub> = 1/6,1644 = 0,162
R <sub>11</sub> = 5/15,0000 = 0,333	R <sub>21</sub> = 5/12,7279 = 0,393	R <sub>31</sub> = 3/6,1644 = 0,487
R <sub>12</sub> = 5/15,0000 = 0,333	R <sub>22</sub> = 4/12,7279 = 0,314	R <sub>32</sub> = 1/6,1644 = 0,162
R <sub>13</sub> = 5/15,0000 = 0,333	R <sub>23</sub> = 4/12,7279 = 0,314	R <sub>33</sub> = 1/6,1644 = 0,162
R <sub>14</sub> = 5/15,0000 = 0,333	R <sub>24</sub> = 4/12,7279 = 0,314	R <sub>34</sub> = 1/6,1644 = 0,162
R <sub>15</sub> = 5/15,0000 = 0,333	R <sub>25</sub> = 5/12,7279 = 0,393	R <sub>35</sub> = 4/6,1644 = 0,649
R <sub>16</sub> = 5/15,0000 = 0,333	R <sub>26</sub> = 4/12,7279 = 0,314	R <sub>36</sub> = 2/6,1644 = 0,324
R <sub>17</sub> = 5/15,0000 = 0,333	R <sub>27</sub> = 4/12,7279 = 0,314	R <sub>37</sub> = 2/6,1644 = 0,324
R <sub>18</sub> = 5/15,0000 = 0,333	R <sub>28</sub> = 4/12,7279 = 0,314	R <sub>38</sub> = 1/6,1644 = 0,162
R <sub>40</sub> = 5/15,0000 = 0,333	R <sub>50</sub> = 5/15,0000 = 0,333	
R <sub>41</sub> = 5/15,0000 = 0,333	R <sub>51</sub> = 5/15,0000 = 0,333	
R <sub>42</sub> = 5/15,0000 = 0,333	R <sub>52</sub> = 5/15,0000 = 0,333	
R <sub>43</sub> = 5/15,0000 = 0,333	R <sub>53</sub> = 5/15,0000 = 0,333	
R <sub>44</sub> = 5/15,0000 = 0,333	R <sub>54</sub> = 5/15,0000 = 0,333	
R <sub>45</sub> = 5/15,0000 = 0,333	R <sub>55</sub> = 5/15,0000 = 0,333	
R <sub>46</sub> = 5/15,0000 = 0,333	R <sub>56</sub> = 5/15,0000 = 0,333	
R <sub>47</sub> = 5/15,0000 = 0,333	R <sub>57</sub> = 5/15,0000 = 0,333	

$R_{48} = 5/15,0000 = 0,333$	$R_{58} = 5/15,0000 = 0,333$
------------------------------	------------------------------

9. Menghitung matriks ternormalisasi yang sudah terbobot dengan bobot yang sudah ditentukan ( $W$ ) = [4,5,3,5,2], dengan rumus :

$$y_{ij} = w_i * r_{ij}$$

$R_{10} = 20 * 0,333 = 6,667$	$R_{20} = 20 * 0,314 = 6,285$	$R_{30} = 15 * 0,162 = 2,433$
$R_{11} = 20 * 0,333 = 6,667$	$R_{21} = 20 * 0,393 = 7,857$	$R_{31} = 15 * 0,487 = 7,300$
$R_{12} = 20 * 0,333 = 6,667$	$R_{22} = 20 * 0,314 = 6,285$	$R_{32} = 15 * 0,162 = 2,433$
$R_{13} = 20 * 0,333 = 6,667$	$R_{23} = 20 * 0,314 = 6,285$	$R_{33} = 15 * 0,162 = 2,433$
$R_{14} = 20 * 0,333 = 6,667$	$R_{24} = 20 * 0,314 = 6,285$	$R_{34} = 15 * 0,162 = 2,433$
$R_{15} = 20 * 0,333 = 6,667$	$R_{25} = 20 * 0,393 = 7,857$	$R_{35} = 15 * 0,649 = 9,733$
$R_{16} = 20 * 0,333 = 6,667$	$R_{26} = 20 * 0,314 = 6,285$	$R_{36} = 15 * 0,324 = 4,867$
$R_{17} = 20 * 0,333 = 6,667$	$R_{27} = 20 * 0,314 = 6,285$	$R_{37} = 15 * 0,324 = 4,867$
$R_{18} = 20 * 0,333 = 6,667$	$R_{28} = 20 * 0,314 = 6,285$	$R_{38} = 15 * 0,162 = 2,433$
$R_{41} = 25 * 0,333 = 8,333$	$R_{51} = 20 * 0,333 = 6,667$	
$R_{42} = 25 * 0,333 = 8,333$	$R_{52} = 20 * 0,333 = 6,667$	
$R_{43} = 25 * 0,333 = 8,333$	$R_{53} = 20 * 0,333 = 6,667$	
$R_{44} = 25 * 0,333 = 8,333$	$R_{54} = 20 * 0,333 = 6,667$	
$R_{45} = 25 * 0,333 = 8,333$	$R_{55} = 20 * 0,333 = 6,667$	

$R_{46} = 25 * 0,333 = 8,333$	$R_{56} = 20 * 0,333 = 6,667$
$R_{47} = 25 * 0,333 = 8,333$	$R_{57} = 20 * 0,333 = 6,667$
$R_{48} = 25 * 0,333 = 8,333$	$R_{58} = 20 * 0,333 = 6,667$

10. Menentukan Solusi Ideal Matriks ( $A^+$ )

$$A^+ = \max(y_1, y_2, \dots, y_n)$$

$$Y_1 = \max(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

$$Y_2 = \max(6.285, 7.857, 6.285, 6.285, 6.285, 7.857, 6.285, 6.285, 6.285) = 7.857$$

$$Y_3 = \max(2.433, 7.300, 2.433, 2.433, 2.433, 9.733, 4.867, 4.867, 2.433) = 9.733$$

$$Y_4 = \max(8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333) = 8.333$$

$$Y_5 = \max(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

11. Menentukan Solusi Ideal Matriks ( $A^-$ )

$$A^- = \min(y_1, y_2, \dots, y_n)$$

$$Y_1 = \min(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.667$$

$$Y_2 = \min(6.285, 7.857, 6.285, 6.285, 6.285, 7.857, 6.285, 6.285, 6.285) = 6.285$$



$$Y_3 = \min(2.433, 7.300, 2.433, 2.433, 2.433, 9.733, 4.867, 4.867, 2.433) = 2.433$$

$$Y_4 = \min(8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333, 8.333) = 8.333$$

$$Y_5 = \min(6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667, 6.667) = 6.66$$

12. Menghitung Jarak Solusi Ideal Positif ( $D^+$ )

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2}$$

$$\begin{aligned} D_1^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (2,433 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 7,467 \end{aligned}$$

$$\begin{aligned} D_2^+ &= \sqrt{(6,667 - 6,667)^2 + (7,857 - 7,857)^2 + (7,300 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 2,433 \end{aligned}$$

$$\begin{aligned} D_3^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (2,433 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 7,467 \end{aligned}$$

$$\begin{aligned} D_4^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (2,433 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 7,467 \end{aligned}$$

$$\begin{aligned} D_5^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (2,433 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 7,467 \end{aligned}$$

$$\begin{aligned} D_6^+ &= \sqrt{(6,667 - 6,667)^2 + (7,857 - 7,857)^2 + (9,733 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 0,000 \end{aligned}$$

$$\begin{aligned} D_7^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (4,867 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\ &= 5,114 \end{aligned}$$

$$\begin{aligned}
D_8^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (4,867 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 5,114 \\
D_9^+ &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 7,857)^2 + (2,433 - 9,733)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 7,467
\end{aligned}$$

### 13. Menghitung Jarak Solusi Ideal Negatif ( $D^-$ )

$$D_i^- = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2}$$

$$\begin{aligned}
D_1^- &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 0,000 \\
D_2^- &= \sqrt{(6,667 - 6,667)^2 + (7,857 - 6,285)^2 + (7,300 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 5,114 \\
D_3^- &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 0,000 \\
D_4^- &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 0,000 \\
D_5^- &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 0,000 \\
D_6^- &= \sqrt{(6,667 - 6,667)^2 + (7,857 - 6,285)^2 + (9,733 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 7,467 \\
D_7^- &= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (4,867 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2} \\
&= 2,433
\end{aligned}$$

$$D_8^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (4,867 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 2,433$$

$$D_9^-$$

$$= \sqrt{(6,667 - 6,667)^2 + (6,285 - 6,285)^2 + (2,433 - 2,433)^2 + (8,333 - 8,333)^2 + (6,667 - 6,667)^2}$$

$$= 0,000$$

#### 14. Menghitung Nilai Preferensi

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}$$

$$V_1 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_2 = \frac{5,114}{5,114} + 2,433 = 0,6776$$

$$V_3 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_4 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_5 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

$$V_6 = \frac{7,467}{7,467} + 0,000 = 1,0000$$

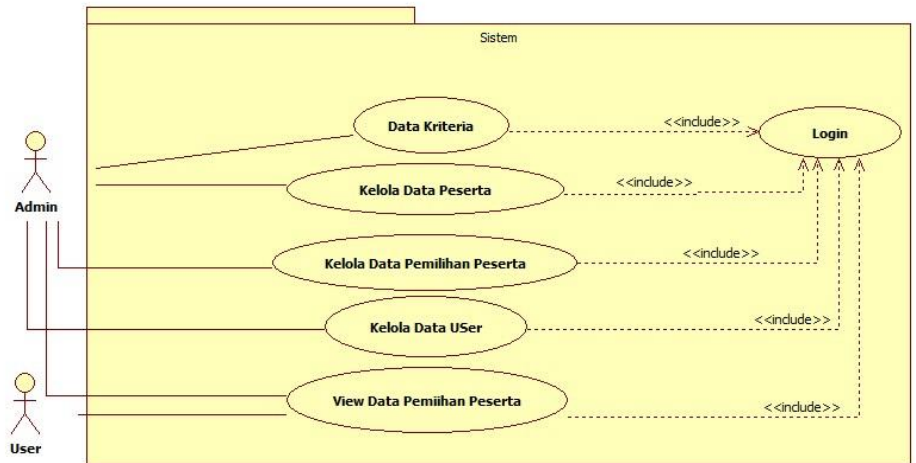
$$V_7 = \frac{2,433}{2,433} + 5,114 = 0,3224$$

$$V_8 = \frac{2,433}{2,433} + 5,114 = 0,3224$$

$$V_9 = \frac{0,000}{0,000} + 7,467 = 0,0000$$

### 5.1.2.4 UML (Unified Modelling Language)

#### 5.1.2.4.1 Usecase Diagram



Gambar 5.26 Usecase Diagram

#### Definisi Usecase

Tabel 5.11 Definisi Usecase

No	Use Case	Deskripsi
1.	Login	Melakukan proses login
2.	Data Kriteria	Merupakan serangkaian data kriteria yang akan berlangsung untuk akademik karyawan.
3.	Kelola Data Peserta	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola data peserta yang mengikuti kegiatan akademik.

4.	Kelola Data Pemilihan Peserta	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola data untuk pemilihan peserta.
5.	Kelola Data User	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola user atau pengguna.
6.	View Data Pemilihan Peserta	Melihat hasil dari data pemilihan peserta.

#### 5.1.2.4.1.2 Usecase Diagram Skenario

*Use Case* Skenario mendeskripsikan urutan langkah-langkah dalam proses sistem, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berikut ini penjelasan dari masing-masing skenario tersebut:

##### 5.1.2.4.1.2.1 Skenario Usecase Login Admin

Tabel 5. 12 Skenario *Usecase Login Admin*

Identifikasi	
Nomor	UC1
Nama	<i>Login admin</i>
Tujuan	Memberikan hak akses admin terhadap sistem dengan melakukan validasi terhadap <i>username</i> dan <i>password</i> yang dimasukan oleh admin
Deskripsi	
Aktor	Admin
Skenario Utama	

<b>Kondisi Awal</b>	<i>Form Login</i> sudah tersedia
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Memasukkan <i>username</i> dan <i>password</i>	2. <i>Form Login</i> akan menampilkan <i>textbox</i> <i>username</i> , <i>password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ‘•’ pada la untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap <i>username</i> , <i>password</i> yang telah dimasukkan dengan menekan tombol <i>Ok</i> .	4. Aplikasi melakukan validasi terhadap <i>username</i> , <i>password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengece pada basis data.
<b>Kondisi Akhir</b>	Jika pada akhir interaksi <i>username</i> , <i>passw</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi se hak aksesnya.
<b>Alur Alternatif kondisi akhir</b> a. Jika tidak <i>valid</i> maka Aplikasi akan kembali ke tampilan awal login.	

#### 5.1.2.4.1.2.2 Skenario *Usecase Diagram Login User*

Tabel 5.13 Skenario *Use Case Login User*

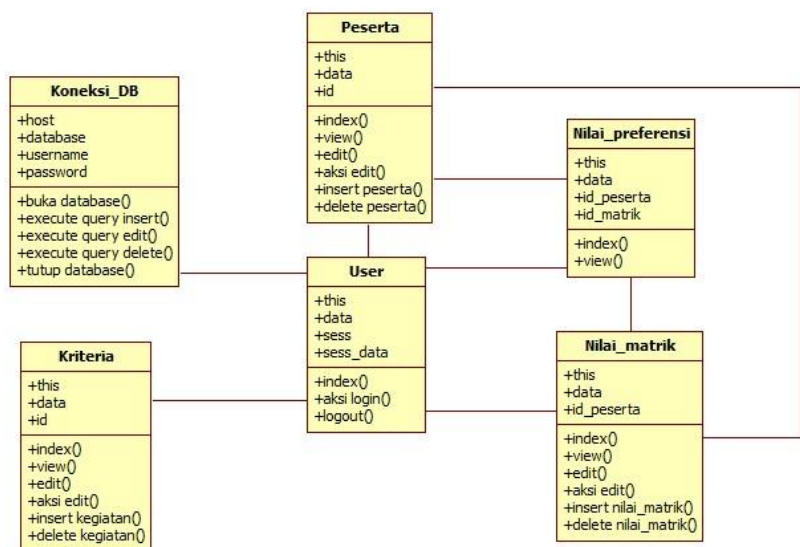
Identifikasi	
Nomor	UC2

<b>Nama</b>	<i>Login User</i>
<b>Tujuan</b>	Memberikan hak akses peserta terhadap sistem dengan melakukan validasi terhadap <i>username</i> dan <i>password</i> yang dimasukan oleh <i>user</i> .
<b>Deskripsi</b>	
<b>Aktor</b>	<i>User</i>
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>Form Login</i> sudah tersedia
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Memasukkan <i>username</i> dan <i>password</i>	2. <i>Form Login</i> akan menampilkan <i>textbox</i> <i>username</i> , <i>password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ‘•’ pada lajur untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap <i>username</i> , <i>password</i> yang telah dimasukkan dengan menekan tombol <i>Ok</i> .	4. Aplikasi melakukan validasi terhadap <i>username</i> , <i>password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengecekan pada basis data.
<b>Kondisi Akhir</b>	Jika pada akhir interaksi <i>username</i> , <i>password</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi sesuai hak aksesnya.
<b>Alur Alternatif kondisi akhir</b>	
a. Jika tidak <i>valid</i> maka Aplikasi akan kembali ke tampilan awal <i>login</i> .	

#### 5.1.2.4.2 Class Diagram

*Class Diagram* menggambarkan *class* dan hubungan antar *class* didalam sistem. *Class* digambarkan dengan sebuah kotak dibagi menjadi tiga bagian. Bagian paling atas diisikan nama *class*, bagian tengah diisikan variabel yang dimiliki *class*, dan bagian bawah diisikan *method-method* dari *class*.

Dalam proses pembuatan aplikasi ini *class diagram* yang akan digunakan adalah sebagai berikut :



Gambar 5.27 Class Diagram

#### 5.1.2.4.3 Interaction Diagram

*Interaction diagram* adalah diagram yang menggambarkan interaksi dalam sistem, antara aplikasi, *user*

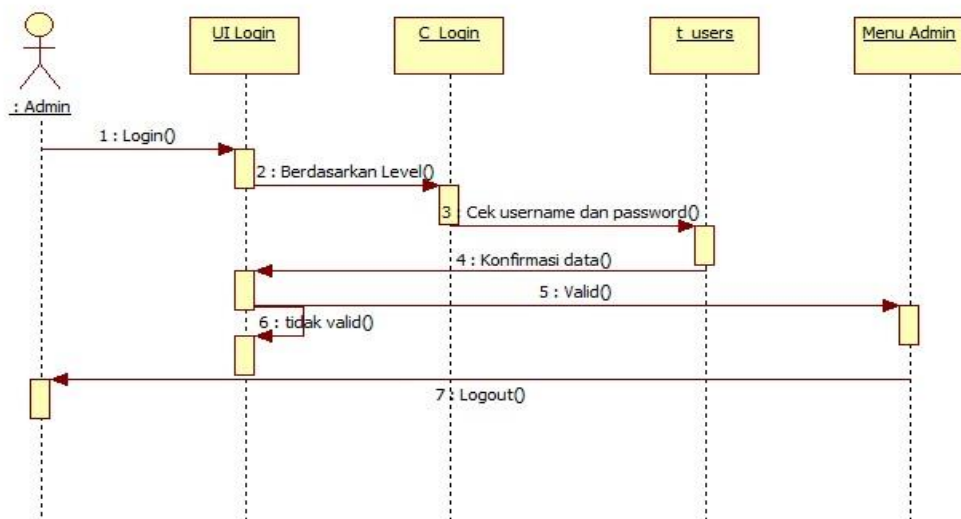


dan *database*. *Interaction diagram* didefinisikan dengan dua tipe diagram yaitu *sequence diagram* dan *collaboration diagram*.

#### 5.1.2.4.4 Sequence Diagram

*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara objek-objek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh objek-objek yang melakukan suatu tugas atau aksi tertentu. Objek-objek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram, pada diagram ini, dimensi vertikal merepresentasikan waktu.

##### 5.1.2.4.4.1 Sequence Diagram Login Admin

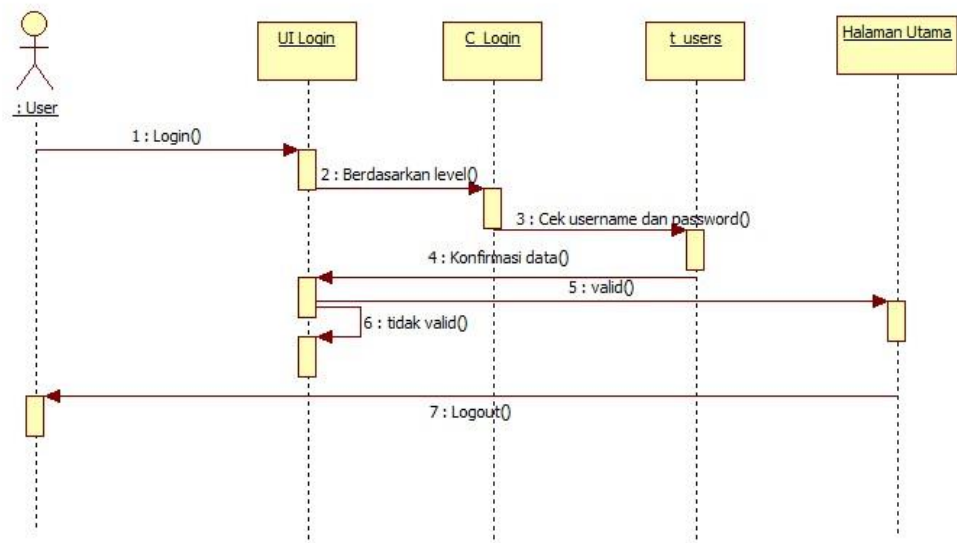


Gambar 5.28 Sequence Diagram Login Admin

Penjelasan :

*Sequence diagram* ini menjelaskan proses *login*. *Admin* mulai menjalankan aplikasi akan tampil *form login*. Kemudian *admin* memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *user*. Jika sesuai maka akan mendapatkan tampilan halaman utama *admin*.

#### 5.1.2.4.4.2 Sequence Diagram Login User

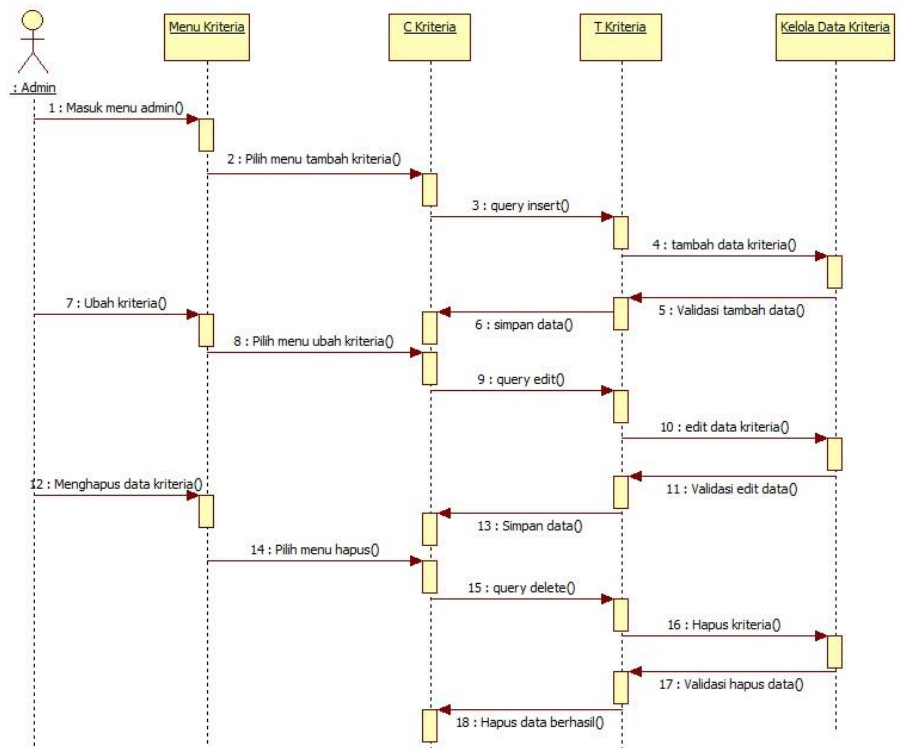


Gambar 5.29 Sequence Diagram Login User

Penjelasan :

*Sequence diagram* ini menjelaskan proses *login*. *User* mulai menjalankan aplikasi akan tampil *form login*. Kemudian *user* memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *user*. Jika sesuai maka user dapat melakukan pendaftaran kegiatan akademik.

#### 5.1.2.4.4.3 Sequence Diagram Kelola Data Kriteria

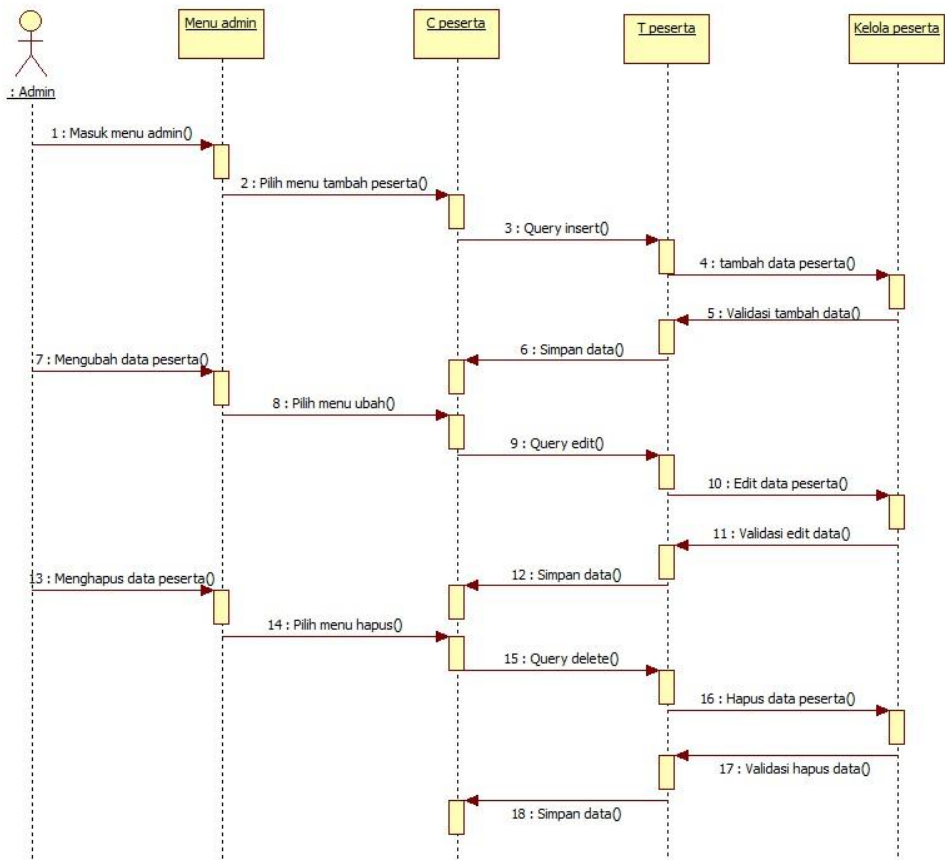


Gambar 5.30 Sequence Diagram Kelola Data Kriteria

Penjelasan :

*Sequence diagram* ini menjelaskan proses kelola data kriteria. Admin mulai menjalankan aplikasi akan tampil *form login*. Kemudian admin memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *database*. Jika sesuai maka admin dapat melakukan pengelolaan data kegiatan dengan *meng input*, ubah dan hapus data pada form tersebut.

#### 5.1.2.4.4 Sequence Diagram Kelola Data Peserta



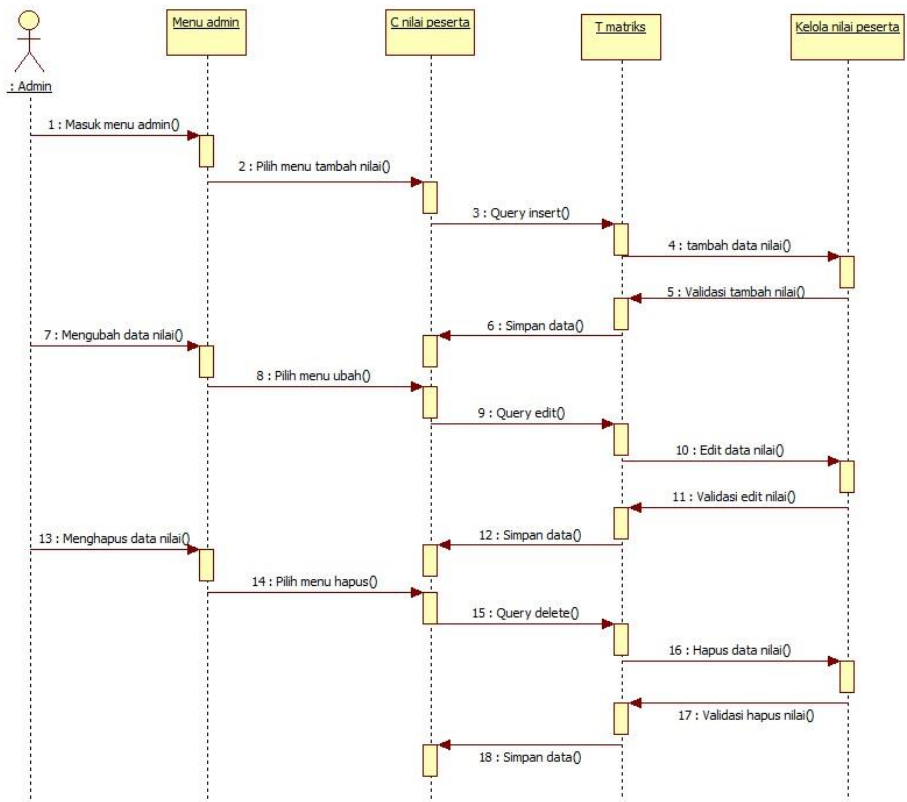
Gambar 5.31 Sequence Diagram Kelola Data Peserta

Penjelasan :

*Sequence diagram* ini menjelaskan proses kelola data peserta. Aktor memilih menu kelola data peserta. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Maka sistem memuat tampilan data

peserta, pada tampilan ini aktor dapat melakukan *input*, ubah dan hapus.

#### 5.1.2.4.4.5 Sequence Diagram Kelola Pemilihan Peserta



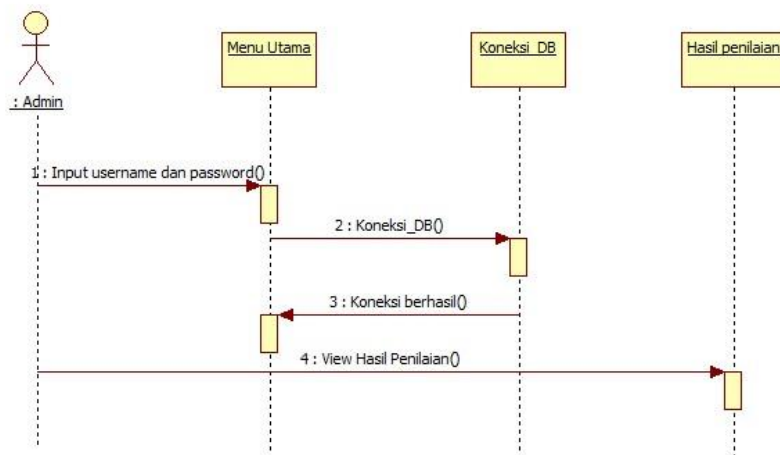
Gambar 5.32 Sequence Diagram Kelola Data Pemilihan Peseta

Penjelasan :

*Sequence diagram* ini menjelaskan proses kelola data pemilihan peserta. Aktor memilih menu kelola data pemilihan peserta. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi

*database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data pemilihan peserta, pada tampilan ini aktor dapat melakukan *input* nilai peserta, ubah dan hapus.

#### 5.1.2.4.4.6 *Sequence Diagram* Admin Hasil Penilaian

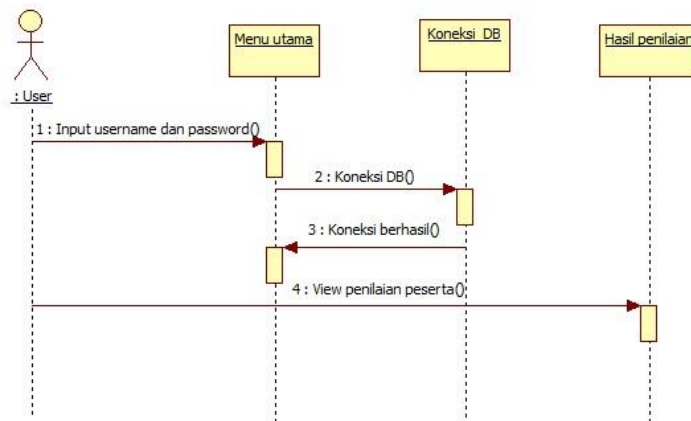


Gambar 5.33 *Sequence Diagram* Admin Hasil Penilaian

Penjelasan :

*Sequence diagram* ini menjelaskan proses hasil penilaian peserta. Aktor memilih menu hasil penilaian. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data hasil penilaian peserta, pada tampilan ini aktor dapat melihat hasil penilaian dari perhitungan sebelumnya.

#### 5.1.2.4.4.7 Sequence Diagram User Hasil Penilaian

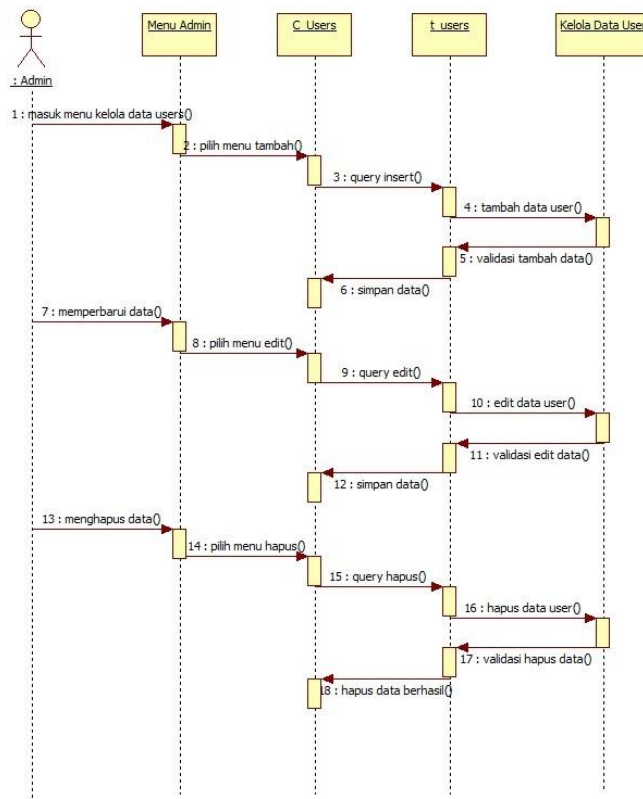


Gambar 5.34 Sequence Diagram Admin Hasil Penilaian

Penjelasan :

*Sequence diagram* ini menjelaskan proses hasil penilaian peserta. Aktor memilih menu hasil penilaian. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data hasil penilaian peserta, pada tampilan ini aktor dapat melihat hasil penilaian dari perhitungan sebelumnya.

#### 5.1.2.4.4.7 Sequence Diagram User Hasil Penilaian



Gambar 5. 35 Sequence Diagram Kelola Data User

Penjelasan :

*Sequence diagram* ini menjelaskan proses pengelolaan *user*. Aktor memilih menu *user*. Namun sebelumnya aktor harus melakukan *login* terlebih dahulu. Maka sistem akan melakukan koneksi *database*. Selanjutnya sistem akan memvalidasi. Pada saat ini sistem memuat tampilan data *user*, pada tampilan ini aktor dapat menambah, mengubah dan menghapus data *user*.



## **Perancangan *User Interface***

### **5.2.1 *User Interface Login***

ACCOUNT LOGIN

Username	Password
<b>SIGN IN</b>	
KEMBALI	

Gambar 5.36 *User Interface Login*

Keterangan Gambar :

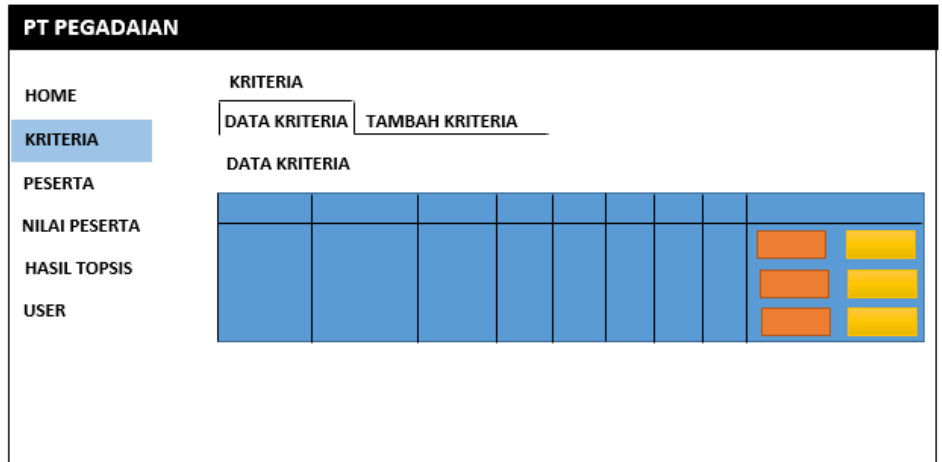
*Textbox 1* : *Input Username*

*Textbox 2* : *Input Password*

*Command Button 1* : *Sign In*

*Command Button 1* : *Kembali*

### 5.2.2 User Interface Kelola Data Kriteria



Gambar 5.37 User Interface Data Kegiatan

Keterangan Gambar :

Label 1 : Kegiatan

Label 2 : Data Kegiatan

Label 3 : Tambah Kegiatan

Table 1 : Tabel Data Kegiatan

### 5.2.3 User Interface Tambah Data Kegiatan

**PT PEGADAIAN**

HOME  
**KRITERIA**  
PESERTA  
NILAI PESERTA  
HASIL TOPSIS  
USER

**KRITERIA**

DATA KRITERIA | **TAMBAH KRITERIA**

TAMBAH KRITERIA

KODE

KRITERIA

JUDUL KEGIATAN

BOBOT

POINT 1

PONT 2

POINT 3

POINT 4

POINT 5

SIFAT KRITERIA

Gambar 5.38 User Interface Tambah Data Kriteria

Keterangan Gambar :

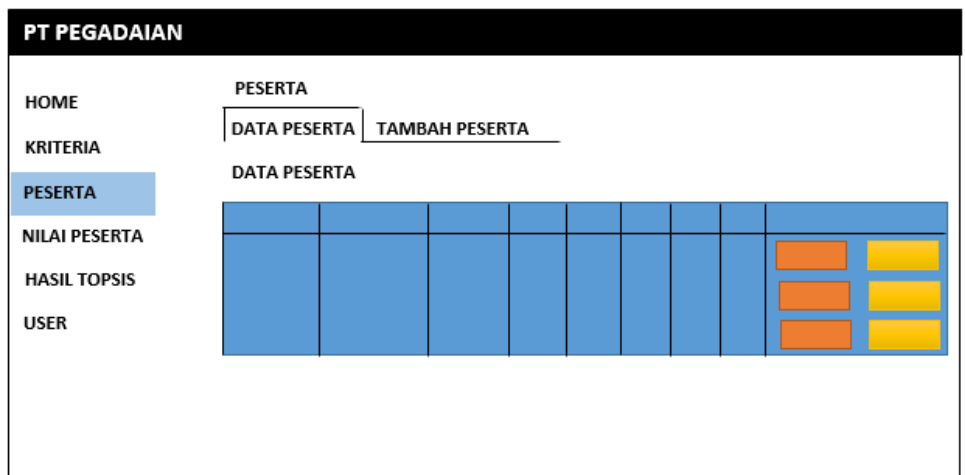
Label 1 : Kriteria

Label 2 : Data Kriteria

Label 3 : Tambah Kriteria

*Textbox 1* : Tambah Data Kriteria  
*Textbox 2* : Tambah Data Kriteria  
*Textbox 3* : Tambah Data Kriteria  
*Textbox 4* : Tambah Data Kriteria  
*Textbox 5* : Tambah Data Kriteria  
*Button 1* : Simpan

#### 5.2.4 User Interface Data Peserta



Gambar 5.39 User Interface Data Peserta

Keterangan Gambar :

*Label 1* : Peserta  
*Label 2* : Data Peserta  
*Label 3* : Tambah Peserta  
*Tabel 1* : Tabel Data Peserta

### 5.2.5 User Interface Tambah Data Peserta

The screenshot shows a web application interface for PT PEGADAIAN. On the left is a vertical navigation menu with the following items: HOME, KRITERIA, PESERTA (highlighted in blue), NILAI PESERTA, HASIL TOPSIS, and USER. The main content area is titled 'PESERTA' and contains two sub-sections: 'DATA PESERTA' and 'TAMBAH PESERTA'. The 'TAMBAH PESERTA' section includes four text input fields labeled 'ID', 'NAMA', 'JUDUL KEGIATAN', and 'TANGGAL'. Below these fields is a blue button. The 'DATA PESERTA' section is currently empty.

Gambar 5. 40 User Interface Tambah Data Peserta

Keterangan Gambar :

*Label 1* : Peserta

*Label 2* : Data Peserta

*Label 3* : Tambah Peserta

*Textbox* : Data Peserta

*Button 1* : Simpan

### 5.2.6 User Interface Data Penilaian

**PT PEGADAIAN**

HOME  
**KRITERIA**  
PESERTA  
NILAI PESERTA  
HASIL TOPSIS  
USER

**NILAI PESERTA**

ISI NILAI PESERTA

TAMBAH NILAI PESERTA

PESERTA

GRADE	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
SERTIFIKASI	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
KEIKUTSERTAAN	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
TEMA	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
KEDISIPLINAN	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

*Gambar 5.41 User Interface Data Penilaian*

Keterangan Gambar :

*Label 1* : Penilaian

*Label 2* : Data Nilai

*Alternatif Box 1* : ID Peserta

*Textbox* : Data Peserta

*Radio Button* : Angka Nilai

*Button 1* : Simpan

### 5.2.7 User Interface Hasil TOPSIS Penilaian

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

USER

HASIL TOPSIS

NILAI PESERTA

NILAI TERNORMALISASI NILAI BOBOT TERNORMALISASI

MATRIKS IDEAL POSITIF/NEGATIF   JARAK SOLUSI IDEAL POSITIF/NEGATIF   NILAI PREFERENSI

NILAI PESERTA


Gambar 5.42 User Interface Hasil Topsis Penilaian

Keterangan Gambar :

Label 1 : Penilaian

Tabel 1 : Tabel Penilaian

### 5.2.8 User Interface Hasil Data Penilaian Ternormalisasi

PT PEGADAIAN						
HOME	HASIL TOPSIS					
KRITERIA	NILAI PESERTA	NILAI TERNORMALISASI			NILAI BOBOT TERNORMALISASI	
PESERTA	MATRIKS IDEAL POSITIF/NEGATIF    JARAK SOLUSI IDEAL POSITIF/NEGATIF    NILAI PREFERENSI					
NILAI PESERTA	NILAI TERNORMALISASI					
HASIL TOPSIS						
USER						

Gambar 5.43 User Interface Hasil Data Penilaian Ternormalisasi

Keterangan Gambar :

*Label 1* : Data Penilaian Ternormalisasi

*Tabel 1* : Tabel Penilaian Ternormalisasi

### 5.2.9 User Interface Hasil Data Penilaian Bobot Ternormalisasi

Ternormalisasi

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

USER

HASIL TOPSIS

NILAI PESERTA

NILAI TERNORMALISASI

NILAI BOBOT TERNORMALISASI

MATRIKS IDEAL POSITIF/NEGATIF

JARAK SOLUSI IDEAL POSITIF/NEGATIF

NILAI PREFERENSI

NILAI BOBOT TERNORMALISASI


*Gambar 5.44 User Interface Hasil Data Penilaian Bobot Ternormalisasi*

Keterangan Gambar :

*Label 1* : Data Penilaian Bobot Ternormalisasi

*Tabel 1* : Tabel Penilaian Bobot Ternormalisasi



### 5.2.10 User Interface Nilai Ideal Positif/Negatif

PT PEGADAIAN											
HOME  KRITERIA  PESERTA  NILAI PESERTA  HASIL TOPSIS  USER	HASIL TOPSIS										
	NILAI PESERTA	NILAI TERNORMALISASI	NILAI BOBOT TERNORMALISASI								
	MATRIKS IDEAL POSITIF/NEGATIF	JARAK SOLUSI IDEAL POSITIF/NEGATIF	NILAI PREFERENSI								
	MATRIKS IDEAL POSITIF A <sup>+</sup>										
<table border="1"> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>											
MATRIKS IDEAL NEGATIF A <sup>-</sup>											
<table border="1"> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>											

Gambar 5.45 User Interface Nilai Ideal Positif/Negatif

Keterangan Gambar :

Label 1 : Nilai Ideal Positif/Negatif

Label 2 : Nilai Ideal Positif A<sup>+</sup>

Label 1 : Nilai Ideal Negatif A<sup>-</sup>

Tabel 1 : Tabel Penilaian Ternormalisasi

Tabel 2 : Tabel Penilaian Ternormalisasi

### 5.2.11 User Interface Jarak Solusi Ideal Positif/Negatif

PT PEGADAIAN

HOME
KRITERIA
PESERTA
NILAI PESERTA
HASIL TOPSIS
USER

HASIL TOPSIS
NILAI PESERTA
NILAI TERNORMALISASI
NILAI BOBOT TERNORMALISASI
Matriks Ideal Positif/Negatif
Jarak Solusi Ideal Positif/Negatif
Nilai Preferensi

Jarak Solusi Ideal Positif D<sup>+</sup>

Jarak Solusi Ideal Negatif D<sup>-</sup>

Gambar 5.29 User Interface Jarak Solusi Ideal Positif/Negatif

Keterangan Gambar :

Label 1 : Jarak Solusi Ideal Positif/Negatif

Label 2 : Jarak Solusi Ideal Positif D<sup>+</sup>

Label 1 : Jarak Solusi Ideal Negatif D<sup>-</sup>

Tabel 1 : Tabel Penilaian Ternormalisasi

Tabel 2 : Tabel Penilaian Ternormalisasi

### 5.2.12 User Interface Nilai Preferensi

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

USER

HASIL TOPSIS

NILAI PESERTA

NILAI TERNORMALISASI

NILAI BOBOT TERNORMALISASI

MATRIKS IDEAL POSITIF/NEGATIF

JARAK SOLUSI IDEAL POSITIF/NEGATIF

NILAI PREFERENSI

NILAI PREFERENSI


Gambar 5.46 User Interface Nilai Preferensi

Keterangan Gambar :

Label 1 : Nilai Preferensi

Tabel 1 : Tabel Nilai Preferensi

### 5.2.13 User Interface Kelola Data User

PT PEGADAIAN

HOME

KRITERIA

PESERTA

NILAI PESERTA

HASIL TOPSIS

USER

USER

DATA USER

TAMBAH USER

DATA USER

			<div></div> <div></div>
			<div></div> <div></div>
			<div></div> <div></div>

Gambar 5.47 User Interface Kelola Data User

Keterangan Gambar :

*Label 1* : User

*Label 2* : Data User

*Label 3* : Tambah User

*Tabel 1* : Tabel Data User

#### 5.2.14 User Interface Tambah Data User

The screenshot displays the PT PEGADAIAN user interface. On the left is a sidebar menu with options: HOME, KRITERIA, PESERTA, NILAI PESERTA, HASIL TOPSIS, and USER (which is highlighted in blue). The main content area is titled 'USER' and contains two sub-sections: 'DATA USER' and 'TAMBAH USER'. The 'TAMBAH USER' section includes three input fields labeled 'NAMA', 'USERNAME', and 'PASSWORD', followed by a blue button labeled 'TAMBAH USER'.

*Gambar 5.48 User Interface Tambah Data User*

Keterangan Gambar :

*Label 1* : User

*Label 2* : Data User

*Label 3* : Tambah User

*Tabel 1* : Tabel Tambah User

## 5.3 Perancangan Arsitektur Sistem

### 5.3.1 Perancangan Arsitektur Perangkat Lunak Sistem

Perangkat Lunak yang digunakan dalam Sistem Implementasi Pemilihan Karyawan Pada Penjadwalan Akademik di PT Pegadaian (Persero) adalah sebagai berikut:

Tabel 5.14 Arsitektur Perangkat Lunak Sistem

No.	Tools/Software	Fungsi	Keterangan
8	<i>Windows 8 (min)</i>	Sistem Operasi	-
9	<i>Bizagi Modeller</i>	Pembuatan Flow Map	-
1	<i>Google Chrome</i>	Aplikasi web browser	-
1	<i>Star UML</i>	Pembuatan Diagram UML	-
1	<i>MySQL</i>	DBMS	-
1	<i>Microsoft Word</i>	Document	-
1	<i>Visual Studio Code</i>	Text Editor	-

### 5.3.2 Perancangan Arsitektur Perangkat Keras Sistem

Perangkat keras yang digunakan dalam Sistem Implementasi Pemilihan Peserta Pada Penjadwalan Kegiatan Akademik di PT Pegadaian ini adalah sebagai berikut :

Tabel 5.15 Arsitektur Perangkat Keras Sistem

No	Nama Perangkat	Spesifikasi	Keterangan
1	<i>Memory</i>	4 GB	<i>Memory System</i> yang digunakan
2	<i>Processor</i>	<i>Intel® core™ CPU</i>	Untuk kecepatan <i>transfer</i> data dari sistem yang sangat bergantung pada kecepatan <i>processor</i> komputer
3	<i>System Type</i>	64-bit <i>Operating System</i> , <i>x64-Based Processor</i>	

## 5.4 Pemetaan Struktur *Diagram User/Aktor Sistem*

Tabel 5. 16 Struktur *Diagram User/Aktor Sistem*

N o	Aktor	Keterangan
1	Admin	Melakukan kelola data kegiatan, , kelola data peserta, dan kelola data pemilihan peserta
2	User	Melakukan login dan View Data Pemilihan Peserta

## 10.5 Pengujian Metode

Tabel 5.10 Data Kriteria Karyawan

Kriteria	Bobot	Point 1	Point 2	Point 3	Point 4	Point 5	Sifat Kriteria
<i>Grade</i>	20	1	2	3	4	5	Benefit
Kesesuaian Proker Corpu	20	1	2	3	4	5	Benefit
Keikutsertaan	15	1	2	3	4	5	Cost
Tema	25	1	2	3	4	5	Benefit
Sharing	20	1	2	3	4	5	Benefit

Tabel 5.11 Nilai Peserta

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin</i>
Eka Pebriansyah	5	4	1	5	5
Supriyanto	5	5	3	5	5
Endang Pertiwi	5	4	1	5	5
Muh Ariyadi	5	4	1	5	5
Hertin Maulida	5	4	1	5	5
Elvi Rofiqatul Hidayah	5	5	4	5	5
Teguh Budi Ismanto	5	4	2	5	5
Eri Mardianto	5	4	2	5	5



Sriyani	5	4	1	5	5
---------	---	---	---	---	---

Tabel 5.12 Nilai Peserta Ternormalisasi

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin g</i>
Eka Pebriansyah	0.333	0.314	0.162	0.333	0.333
Supriyanto	0.333	0.393	0.487	0.333	0.333
Endang Pertiwi	0.333	0.314	0.162	0.333	0.333
Muh Ariyadi	0.333	0.314	0.162	0.333	0.333
Hertin Maulida	0.333	0.393	0.649	0.333	0.333
Elvi Rofiqatul Hidayah	0.333	0.314	0.324	0.333	0.333
Teguh Budi Ismanto	0.333	0.314	0.324	0.333	0.333

Eri Mardianto	0.33 3	0.314	0.162	0.33 3	0.333
Sriyani	0.33 3	0.314	0.162	0.33 3	0.333

Tabel 5.13 Nilai Peserta bobot Ternormalisasi

Nama	<i>Grade</i>	Kesesuaian Proker	Keikutsertaan	Tempa	<i>Sharin g</i>
Eka Pebriansyah	6.66 7	4	1	8.33 3	6.667
Supriyanto	6.66 7	6.285	2.433	8.33 3	6.667
Endang Pertiwi	6.66 7	7.857	7.3	8.33 3	6.667
Muh Ariyadi	6.66 7	6.285	2.433	8.33 3	6.667
Hertin Maulida	6.66 7	6.285	2.433	8.33 3	6.667
Elvi Rofiqatul Hidayah	6.66 7	7.857	9.733	8.33 3	6.667

Teguh Budi Ismanto	6.66 7	6.285	4.867	8.33 3	6.667
Eri Mardianto	6.66 7	6.285	4.867	8.33 3	6.667
Sriyani	6.66 7	6.285	4.867	8.33 3	6.667

Tabel 5.14 Nilai Ideal Positif

<i>Alternatif</i>	A+
Grade	6.667
Kesesuaian dengan Proker	7.857
Keikutsertaan	9.733
Tema	8.333
Sharing	6.667

Tabel 5.15 Nilai Ideal Negatif

<i>Alternatif</i>	D+
Grade	6.667
Kesesuaian dengan Proker	6.285
Keikutsertaan	2.433
Tema	8.333
Sharing	6.667

Tabel 5.16 Jarak Solusi Ideal Positif

<i>Alternatif</i>	D+
Eka Pebriansyah	7.467
Supriyanto	2.433
Endang Pertiwi	7.467
Muh Ariyadi	7.467
Hertin Maulida	7.467

Elvi Rofiqatul Hidayah	0
Teguh Budi Ismanto	5.114
Eri Mardianto	5.114
Sriyani	7.467

Tabel 5.17 Jarak Solusi Ideal Negatif

<i>Alternatif</i>	D+
Eka Pebriansyah	0
Supriyanto	5.115
Endang Pertiwi	0
Muh Ariyadi	0
Hertin Maulida	0
Elvi Rofiqatul Hidayah	7.467
Teguh Budi Ismanto	2.434

Eri Mardianto	2.434
Sriyani	0

Tabel 5. 18 Nilai Preferensi

Nama	Preferensi
Eka Pebriansyah	0
Supriyanto	0.6777
Endang Pertiwi	0
Muh Ariyadi	0
Hertin Maulida	0
Elvi Rofiqatul Hidayah	1
Teguh Budi Ismanto	0.3225
Eri Mardianto	0.3225
Sriyani	0

Pada tahap pengujian metode ini, dilakukan pengujian metode untuk mengetahui apakah metode Topsis dapat menghasilkan pemilihan alternatif terbaik untuk karyawan dengan proses pengolahan data menggunakan bahasa pemrograman PHP.

Berikut adalah hasil pengujian metode Topsis :

Nilai peserta

Nilai Berdasarkan Tanggal : dd/mm/yyyy		Search		Nilai Berdasarkan Kegiatan : - Pilih -		Search	
No	Nama	kriteria					
		C1	C2	C3	C4	C5	
1	Eka Pebriansyah	5	4	1	5	5	
2	Supriyanto	5	5	3	5	5	
3	Endang Pertiwi	5	4	1	5	5	
4	Muh Ariyadi	5	4	1	5	5	
5	Hertin Maulida	5	4	1	5	5	
6	Elvi Rofiqatul Hidayah	5	5	4	5	5	
7	Teguh Budi Ismanto	5	4	2	5	5	
8	Eri Mardianto	5	4	2	5	5	
9	Sriyani	5	4	1	5	5	

Gambar 6.9 Nilai Peserta

Nilai peserta ternormalisasi

Nilai Berdasarkan Tanggal : <input type="text" value="dd/mm/yyyy"/>		Search		Nilai Berdasarkan Kegiatan : <input type="text" value="- Pilih -"/>		Search	
No	Nama	kriteria					
		C1	C2	C3	C4	C5	
1	Eka Pebriansyah	0.333	0.314	0.162	0.333	0.333	
2	Supriyanto	0.333	0.393	0.487	0.333	0.333	
3	Endang Pertiwi	0.333	0.314	0.162	0.333	0.333	
4	Muh Ariyadi	0.333	0.314	0.162	0.333	0.333	
5	Hertin Maulida	0.333	0.314	0.162	0.333	0.333	
6	Elvi Rofiqatul Hidayah	0.333	0.393	0.649	0.333	0.333	
7	Teguh Budi Ismanto	0.333	0.314	0.324	0.333	0.333	
8	Eri Mardianto	0.333	0.314	0.324	0.333	0.333	
9	Sriyani	0.333	0.314	0.162	0.333	0.333	

Gambar 6.10 Nilai Peserta Ternormalisasi

Nilai Bobot Ternormalisasi

Nilai Berdasarkan Tanggal : <input type="text" value="dd/mm/yyyy"/>		Search		Nilai Berdasarkan Kegiatan : <input type="text" value="- Pilih -"/>		Search	
No	Nama	kriteria					
		C1	C2	C3	C4	C5	
1	Eka Pebriansyah	6.667	6.285	2.433	8.333	6.667	
2	Supriyanto	6.667	7.857	7.3	8.333	6.667	
3	Endang Pertiwi	6.667	6.285	2.433	8.333	6.667	
4	Muh Ariyadi	6.667	6.285	2.433	8.333	6.667	
5	Hertlin Maulida	6.667	6.285	2.433	8.333	6.667	
6	Elvi Rofiqatul Hidayah	6.667	7.857	9.733	8.333	6.667	
7	Teguh Budi Ismanto	6.667	6.285	4.867	8.333	6.667	
8	Eri Mardianto	6.667	6.285	4.867	8.333	6.667	
9	Sriyani	6.667	6.285	2.433	8.333	6.667	

Gambar 6.11 Nilai Peserta Terbobot Ternormalisasi

Matriks Ideal Positif ( $A^+$ )

Nilai Berdasarkan Tanggal :  
 Nilai Berdasarkan Kegiatan :

kriteria				
kr002	kr003	kr004	kr005	kr006
$y_1^+$	$y_2^+$	$y_3^+$	$y_4^+$	$y_5^+$
6.667	7.857	9.733	8.333	6.667

Gambar 6.12 Nilai Ideal Positif

Matriks Ideal Positif ( $A^-$ )

kriteria				
kr002	kr003	kr004	kr005	kr006
$y_1^-$	$y_2^-$	$y_3^-$	$y_4^-$	$y_5^-$
6.667	6.285	2.433	8.333	6.667

Gambar 6.13 Nilai Ideal Negatif

Jarak Solusi Ideal Positif ( $D^+$ )

Nilai Berdasarkan Tanggal :  
 Nilai Berdasarkan Kegiatan :

Nomor	Nama	$D^+$
1	Eka Pebriansyah	7.467
2	Supriyanto	2.433
3	Endang Pertiwi	7.467
4	Muh Ariyadi	7.467
5	Hertin Maulida	7.467
6	Elvi Rofiqatul Hidayah	0
7	Teguh Budi Ismanto	5.114
8	Eri Mardianto	5.114
9	Sriyani	7.467

Gambar 6.14 Nilai Jarak Solusi Ideal Positif



Jarak Solusi Ideal Negatif ( $D^-$ )

Nomor	Nama	$D^-$
1	Eka Pebriansyah	0
2	Supriyanto	5.115
3	Endang Pertiwi	0
4	Muh Ariyadi	0
5	Hertin Maulida	0
6	Elvi Rofiqatul Hidayah	7.467
7	Teguh Budi Ismanto	2.434
8	Eri Mardianto	2.434
9	Sriyani	0

Gambar 6.15 Nilai Jarak Solusi Ideal Negatif

Nilai Preferensi

Nomor	Nama	$V_i$
1	Elvi Rofiqatul Hidayah	1
2	Supriyanto	0.6777
3	Teguh Budi Ismanto	0.3225
4	Eri Mardianto	0.3225
5	Eka Pebriansyah	0
6	Endang Pertiwi	0
7	Muh Ariyadi	0
8	Hertin Maulida	0
9	Sriyani	0

Gambar 6.16 Nilai Preferensi