BAB IV
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Bagian implementasi ini akan memperlihatkan hasil dari perancangan modul yang telah dibuat dan dijelaskan pada bagian *design.* Pada bagian ini antara lain akan dijelaskan mengenai kode program untuk tiap modul dari sistem ini.

4.1.1 Perangkat Lunak Pembangun

Perangkat lunak yang digunakan pada sistem komputer dalam membangun Sistem Pendukukung Pengambilan Keputusan Mutasi Pegawai Dengan Menggunakan Metode *Simple Additive Weighting* Pada Badan Kepegawai Daerah Kota Bandung ini, adalah sebagai berikut:

1. Sistem Operasi *Windows XP Profesional SP 2*.
2. *Ruby on Rails*
3. *Database* menggunakan *mysql.*
4. *Mozilla Firefox* sebagai *browser*.
5. *Xampp* untuk penyimpanan *database*,*web server*.

4.1.2 Perangkat Keras

Perangkat keras yang dibutuhkan untuk menjalankan sistem informasi managemen ini seperti yang sudah dijelaskan pada BAB III Analisa.

4.1.3 Hasil Implementasi

Dari batasan implementasi yang telah ditentukan, elemen-elemen yang telah terimplementasi adalah sebagai berikut:

1. Tampilan untuk *login*
2. Tampilan untuk menu utama visi & misi Kota Bandung.
3. Tampilan untuk menu lihat pegawai
4. Tampilan untuk menu isi pegawai baru.
5. Pemilihan menu import pegawai
6. Pemilihan menu pengelolaan *user*
7. Pemilihan menu pembuatan *user*
8. Pemilihan menu pengajuan mutasi
9. Pemilihan menu laporan pengajuan mutasi
10. Pemilihan menu lihat mutasi
11. Pemilihan menu verifikasi mutasi.
12. Pemilihan menu pengisian DP3, Pengisian nilai ujian dinas, seleksi jabatan.
13. Pemilihan menu laporan verifikasi pegawai.
14. Pemilihan menu kriteria penilaian *SAW*.
15. Pemilihan menu perhitungan *SAW*.
16. Pemilihan menu keputusan mutasi
17. Pemilihan menu laporan persetujuan mutasi

4.1.4 Implementasi *Database*

Deskripsi tabel dari *database* tersebut yaitu :

Tabel 4.1 Tabel *Users*

Tabel *Users*
CREATE TABLE `users` (`user_id` int(11) NOT NULL AUTO_INCREMENT, `login` varchar(40) COLLATE utf8_unicode_ci DEFAULT NULL, `name` varchar(100) COLLATE utf8_unicode_ci DEFAULT '', `email` varchar(100) COLLATE utf8_unicode_ci DEFAULT NULL, `crypted_password` varchar(40) COLLATE utf8_unicode_ci DEFAULT NULL, `salt` varchar(40) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `remember_token` varchar(40) COLLATE utf8_unicode_ci DEFAULT NULL, `remember_token_expires_at` datetime DEFAULT NULL, PRIMARY KEY (`user_id`), UNIQUE KEY `index_users_on_login` (`login`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.2 Tabel *Employees*

Tabel *Employees*
CREATE TABLE `employees` (`employee_id` int(11) NOT NULL AUTO_INCREMENT, `nip` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `gelar` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `tempat_lahir` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `tgl_lahir` date DEFAULT NULL, `tmt_cpns` date DEFAULT NULL, `jenis_kelamin` int(11) DEFAULT NULL, `status_nikah` int(11) DEFAULT NULL, `agama` int(11) DEFAULT NULL, `golongan` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `tmt_gol` date DEFAULT NULL, `jabatan_struktural` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `eselon` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `tmt_jbt_struktural` date DEFAULT NULL, `jabatan_fungsional` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `tmt_jafung` date DEFAULT NULL, `pendidikan` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `sekolah` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `jurusan` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `tahun_lulus` int(11) DEFAULT NULL, `ket` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `unit_kerja` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `kode_skpd` int(11) DEFAULT NULL, `skpd` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` datetime DEFAULT NULL,

```
  `updated_at` datetime DEFAULT NULL,
  PRIMARY KEY (`employee_id`)
 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```

Tabel 4.3 Tabel *Roles*

Tabel *Roles*
CREATE TABLE `roles` (`role_id` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, PRIMARY KEY (`role_id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.4 Tabel *Roles Users*

Tabel *Role Users*
CREATE TABLE `roles_users` (`role_id` int(11) DEFAULT NULL, `user_id` int(11) DEFAULT NULL, KEY `role_id` (`role_id`), KEY `user_id` (`user_id`), CONSTRAINT `roles_users_ibfk_1` FOREIGN KEY (`role_id`) REFERENCES `roles` (`role_id`) ON UPDATE CASCADE, CONSTRAINT `roles_users_ibfk_2` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.5 Tabel *Judment_values*

Tabel *Judment_values*
CREATE TABLE `judment_values` (` judment_value_id` int(11) NOT NULL AUTO_INCREMENT, `kesetiaan` float DEFAULT NULL, `prestasi_kerja` float DEFAULT NULL, `tanggung_jawab` float DEFAULT NULL, `ketaatan` float DEFAULT NULL, `kejujuran` float DEFAULT NULL, `kerja_sama` float DEFAULT NULL, `prakarsa` float DEFAULT NULL, `kepemimpinan` float DEFAULT NULL, `mutation_id` int(11) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `average_value` float DEFAULT NULL, PRIMARY KEY (` judment_value_id`), KEY `mutation_id` (`mutation_id`), CONSTRAINT `judment_values_ibfk_1` FOREIGN KEY (`mutation_id`) REFERENCES `mutations` (`mutation_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.6 Tabel *Mutations*

Tabel *Mutations*
CREATE TABLE `mutations` (`mutation_id` int(11) NOT NULL AUTO_INCREMENT, `user_id` int(11) DEFAULT NULL, `employee_id` int(11) DEFAULT NULL, `mutation_type` int(11) DEFAULT NULL, `status` int(11) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `dss_application_id` int(11) DEFAULT NULL, PRIMARY KEY (`mutation_id`), KEY `user_id` (`user_id`), KEY `employee_id` (`employee_id`), KEY `dss_application_id` (`dss_application_id`), CONSTRAINT `mutations_ibfk_1` FOREIGN KEY (`dss_application_id`) REFERENCES `dss_applications` (`dss_application_id`) ON UPDATE CASCADE, CONSTRAINT `mutations_ibfk_2` FOREIGN KEY (`employee_id`) REFERENCES `employees` (`employee_id`) ON UPDATE CASCADE, CONSTRAINT `mutations_ibfk_3` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.7 Tabel *Rank_selections*

Tabel *Rank_selection*
CREATE TABLE `rank_selections` (` rank_selection_id` int(11) NOT NULL AUTO_INCREMENT, `mutation_id` int(11) DEFAULT NULL, `value` float DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, PRIMARY KEY (` rank_selection_id`), KEY `mutation_id` (`mutation_id`), CONSTRAINT `rank_selections_ibfk_1` FOREIGN KEY (`mutation_id`) REFERENCES `mutations` (`mutation_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.8 Tabel *Suitablities*

Tabel *Suitabilities*
CREATE TABLE `suitabilities` (`suitability_id` int(11) NOT NULL AUTO_INCREMENT, `label` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `value` float DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `dss_application_id` int(11) DEFAULT NULL, PRIMARY KEY (`suitability_id`),

```
 KEY `dss_application_id` (`dss_application_id`),
 CONSTRAINT `suitabilities_ibfk_1` FOREIGN KEY (`dss_application_id`)
REFERENCES `dss_applications` (`dss_application_id`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci CONSTRAINT `service_exams_ibfk_1` FOREIGN
KEY (`mutation_id`) REFERENCES `mutations` (`mutation_id`) ON UPDATE
CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```

Tabel 4.9 Tabel *Service_Exams*

Tabel *Service_Exams*
CREATE TABLE `service_exams` (`service_exam_id` int(11) NOT NULL AUTO_INCREMENT, `mutation_id` int(11) DEFAULT NULL, `value` float DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, PRIMARY KEY (`service_exam_id`), KEY `mutation_id` (`mutation_id`), CONSTRAINT `service_exams_ibfk_1` FOREIGN KEY (`mutation_id`) REFERENCES `mutations` (`mutation_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.10 Tabel *Attribute Values*

Tabel *Attribute_Values*
CREATE TABLE `attribute_values` (`attribute_value_id` int(11) NOT NULL AUTO_INCREMENT, `alternative_id` int(11) DEFAULT NULL, `attribute_id` int(11) DEFAULT NULL, `value` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, PRIMARY KEY (`attribute_value_id`), KEY `alternative_id` (`alternative_id`), KEY `attribute_id` (`attribute_id`), CONSTRAINT `attribute_values_ibfk_1` FOREIGN KEY (`attribute_id`) REFERENCES `attributes` (`attribute_id`) ON UPDATE CASCADE, CONSTRAINT `attribute_values_ibfk_2` FOREIGN KEY (`alternative_id`) REFERENCES `alternatives` (`alternative_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.11 Tabel *Alternatives*

Tabel *Alternatives*
CREATE TABLE `alternatives` (`alternative_id` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `dss_application_id` int(11) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `final_score` double DEFAULT '0', `rank` int(11) DEFAULT NULL, `mutation_id` int(11) DEFAULT NULL, PRIMARY KEY (`alternative_id`), KEY `dss_application_id` (`dss_application_id`), KEY `mutation_id` (`mutation_id`), CONSTRAINT `alternatives_ibfk_1` FOREIGN KEY (`dss_application_id`) REFERENCES `dss_applications` (`dss_application_id`) ON UPDATE CASCADE, CONSTRAINT `alternatives_ibfk_2` FOREIGN KEY (`mutation_id`) REFERENCES `mutations` (`mutation_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.12 Tabel *Suitablity_Values*

Tabel *Suitability_Values*
CREATE TABLE `suitability_values` (`suitability_value_id` int(11) NOT NULL AUTO_INCREMENT, `alternative_id` int(11) DEFAULT NULL, `criteria_id` int(11) DEFAULT NULL, `suitability_id` int(11) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, PRIMARY KEY (`suitability_value_id`), KEY `alternative_id` (`alternative_id`), KEY `criteria_id` (`criteria_id`), KEY `suitability_id` (`suitability_id`), CONSTRAINT `suitability_values_ibfk_1` FOREIGN KEY (`criteria_id`) REFERENCES `criterias` (`criteria_id`) ON UPDATE CASCADE, CONSTRAINT `suitability_values_ibfk_2` FOREIGN KEY (`alternative_id`) REFERENCES `alternatives` (`alternative_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.13 Tabel *Attributes*

Tabel *Attributes*
CREATE TABLE `attributes` (`attribute_id` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `attr_type` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `dss_application_id` int(11) DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL,

```
 PRIMARY KEY (`attribute_id`),
 KEY `dss_application_id` (`dss_application_id`),
 CONSTRAINT `attributes_ibfk_1` FOREIGN KEY (`dss_application_id`)
REFERENCES `dss_applications` (`dss_application_id`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
```

Tabel 4.14 Tabel DSS_Aplications

Tabel Dss_aplications
CREATE TABLE `dss_applications` (`dss_application_id` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `user_id` int(11) DEFAULT NULL, `alternative_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, PRIMARY KEY (`dss_application_id`), KEY `user_id` (`user_id`), CONSTRAINT `dss_applications_ibfk_1` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.15 Tabel *Scores*

Tabel *Scores*
CREATE TABLE `scores` (`score_id` int(11) NOT NULL AUTO_INCREMENT, `alternative_id` int(11) DEFAULT NULL, `criteria_id` int(11) DEFAULT NULL, `value` double DEFAULT '0', `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, PRIMARY KEY (`score_id`), KEY `alternative_id` (`alternative_id`), KEY `criteria_id` (`criteria_id`), CONSTRAINT `scores_ibfk_1` FOREIGN KEY (`alternative_id`) REFERENCES `alternatives` (`alternative_id`) ON UPDATE CASCADE, CONSTRAINT `scores_ibfk_2` FOREIGN KEY (`criteria_id`) REFERENCES `criterias` (`criteria_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci CONSTRAINT `suitability_values_ibfk_1` FOREIGN KEY (`criteria_id`) REFERENCES `criterias` (`criteria_id`) ON UPDATE CASCADE, CONSTRAINT `suitability_values_ibfk_2` FOREIGN KEY (`alternative_id`) REFERENCES `alternatives` (`alternative_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

Tabel 4.16 Tabel *Criterias*

Tabel *Criterias*
CREATE TABLE `criterias` (`criteria_id` int(11) NOT NULL AUTO_INCREMENT, `description` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `is_benefit` tinyint(1) DEFAULT NULL, `preference_weight` float DEFAULT NULL, `created_at` datetime DEFAULT NULL, `updated_at` datetime DEFAULT NULL, `dss_application_id` int(11) DEFAULT NULL, PRIMARY KEY (`criteria_id`), KEY `dss_application_id` (`dss_application_id`), CONSTRAINT `criterias_ibfk_1` FOREIGN KEY (`dss_application_id`) REFERENCES `dss_applications` (`dss_application_id`) ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

4.1.5 Implementasi Class pada Program

Implementasi semua kode program dilakukan dengan menggunakan bahasa pemrograman *Ruby on Rails*. Kelas – kelas yang terdapat pada *class diagram* telah di implementasi, berbagai kode kelas tersebut yaitu :

1. Class User

Tabel 4.17 Tabel *Class User*

Nama Class	User
Jenis Class	Boundary
Nama File	ClassUser

```
class User < ActiveRecord::Base


    # ----------------------------------------
  # The following code has been generated by
role_requirement.
  # You may wish to modify it to suit your need
  has_and_belongs_to_many :roles

  # has_role? simply needs to return true or false whether
a user has a role or not.
  # It may be a good idea to have "admin" roles return
true always
  def has_role?(role_in_question)
    @_list ||= self.roles.collect(&:name)
#    return true if @_list.include?("admin")
```

```
    (@_list.include?(role_in_question.to_s) )
  end
  # ----------------------------------------

  include Authentication
  include Authentication::ByPassword
  include Authentication::ByCookieToken

  validates_presence_of     :login
  validates_length_of       :login,    :within => 3..40
  validates_uniqueness_of   :login
  validates_format_of       :login,    :with =>
Authentication.login_regex, :message =>
Authentication.bad_login_message

  validates_format_of       :name,     :with =>
Authentication.name_regex,  :message =>
Authentication.bad_name_message, :allow_nil => true
  validates_length_of       :name,     :maximum => 100

  validates_presence_of     :email
  validates_length_of       :email,    :within => 6..100
#r@a.wk
  validates_uniqueness_of   :email
  validates_format_of       :email,    :with =>
Authentication.email_regex, :message =>
Authentication.bad_email_message


  # HACK HACK HACK -- how to do attr_accessible from here?
  # prevents a user from submitting a crafted form that
bypasses activation
  # anything else you want your user to change should be
added here.
  attr_accessible :login, :email, :name, :password,
:password_confirmation


  # Authenticates a user by their login name and
unencrypted password.  Returns the user or nil.
  #
  # uff.  this is really an authorization, not
authentication routine.
  # We really need a Dispatch Chain here or something.
  # This will also let us return a human error message.
  #
  def self.authenticate(login, password)
    return nil if login.blank? || password.blank?
    u = find_by_login(login.downcase) # need to get the
salt
    u && u.authenticated?(password) ? u : nil
```

```
  end

  def login=(value)
    write_attribute :login, (value ? value.downcase : nil)
  end

  def email=(value)
    write_attribute :email, (value ? value.downcase : nil)
  end

  protected

end
```

2. Class FormUser

Tabel 4.18 Tabel *Class FormUser*

Nama Class	FormUser
Jenis Class	View
Nama File	classFormUser

```
<section id="dashtabs" class="grid_12">
  <div class="box-header">
    Daftar User
    <ul class="controls">
      <li><a title="Buat user baru" class="tooltip"
href="/users/new"><img
src="/img/icons/16/add.png"></a></li>
    </ul>
  </div>
  <div class="box-content no-padding" id="dashtabs-pages">
    <table class="table no-border">
      <thead>
        <tr>
          <td>Username</td>
          <td>Nama</td>
          <td>Email</td>
          <td>Akses</td>
          <td class="last">Opsi</td>
        </tr>
      </thead>
      <tbody>
        <% @users.each do |user| %>
          <tr class="last">
            <td><%= user.login %></td>
            <td><%= user.name %></td>
            <td><%= user.email %></td>
            <td><%= user.roles.map(&:name) %></td>
            <td class="last">
```

```
              <%= link_to
image_tag("/img/icons/16/page_edit.png", :alt => "edit"),
edit_user_path(user), :class => "tooltip", :title => "ubah
user" %>
              <%= link_to
image_tag("/img/icons/16/page_delete.png", :alt =>
"delete"), user, :confirm => 'Are you sure?', :method =>
:delete,:class => "tooltip", :title => "hapus user" %>
            </td>
          </tr>
        <% end %>
      </tbody>
    </table>
  </div>
</section>
```

3. Class Employee

Tabel 4.19 Tabel *Class Employee*

Nama Class	*Employee*
Jenis Class	*Boundary*
Nama File	*ClassEmployee*

```
class Employee < ActiveRecord::Base
  require 'fastercsv'

  has_many :mutations, :dependent => :destroy
  has_many :judment_values, :dependent => :destroy
  has_many :judment_values, :dependent => :destroy
  has_many :judment_values, :dependent => :destroy

  validates_presence_of :nip, :name
  validates_uniqueness_of :nip


  def self.load_from_xls(path)
    FasterCSV.foreach(path) do |row|
      tgl_lahir = "#{row[5].split("/").last}-
#{row[5].split("/").first}-#{row[5].split("/")[1]}" rescue
nil
      tmt_cpns = "#{row[6].split("/").last}-
#{row[6].split("/").first}-#{row[6].split("/")[1]}" rescue
nil
      tmt_gol = "#{row[11].split("/").last}-
#{row[11].split("/").first}-#{row[11].split("/")[1]}"
rescue nil
      tmt_jbt_struktural = "#{row[14].split("/").last}-
#{row[14].split("/").first}-#{row[14].split("/")[1]}"
rescue nil
```

```
      e = Employee.new(
         :nip => row[1],
         :name => row[2],
         :gelar => row[3],
         :tempat_lahir => row[4],
         :tgl_lahir => tgl_lahir,
         :tmt_cpns => tmt_cpns,
         :jenis_kelamin => row[7],
         :status_nikah =>  row[8],
         :agama => row[9],
         :golongan => row[10],
         :tmt_gol => tmt_gol,
         :jabatan_struktural => row[12],
         :eselon => row[13],
         :tmt_jbt_struktural => tmt_jbt_struktural,
         :jabatan_fungsional => row[15],
         :tmt_jafung => row[16],
         :pendidikan => row[17],
         :sekolah => row[18],
         :jurusan => row[19],
         :tahun_lulus => row[20],
         :ket => row[21],
         :unit_kerja => row[22],
         :kode_skpd => row[23]
       )
       e.save
     end

  end
end
```

4. Class FormPegawai

Tabel 4.20 Tabel *Class FormEmployee*

Nama Class	FormPegawai
Jenis Class	View
Nama File	classFormPegawai

```
<section id="dashtabs" class="grid_12">
  <div class="box-header">
    Daftar Pegawai
  </div>
  <div class="box-content no-padding" id="dashtabs-pages">
    <table class="table">
      <thead>
        <tr>
          <th>Nip</th>
          <th>Nama</th>
          <th>Tmt cpns</th>
```

```
          <th>Golongan</th>
          <th>Jabatan Struktural</th>
          <th>Jabatan Eselon</th>
          <th>Jabatan fungsional</th>
          <th>Unit kerja</th>
          <th>Skpd</th>
          <% if current_user.has_role?("admin") %>
            <th></th>
          <% end %>
        </tr>
      </thead>
      <tbody>
        <% @employees.each do |employee| %>
          <tr>
            <td><%=h employee.nip %></td>
            <td><%=h employee.name %></td>
            <td><%=h employee.tmt_cpns %></td>
            <td><%=h employee.golongan %></td>
            <td><%=h employee.jabatan_struktural %></td>
            <td><%=h employee.eselon %></td>
            <td><%=h employee.jabatan_fungsional %></td>
            <td><%=h employee.unit_kerja %></td>
            <td><%=h employee.skpd %></td>
            <% if current_user.has_role?("admin") %>
              <td>
                <%= link_to
image_tag("/img/icons/16/page_edit.png", :alt => "edit"),
edit_employee_path(employee) %>
                <%= link_to
image_tag("/img/icons/16/page_delete.png", :alt =>
"delete"), employee_path(employee), :method => :delete,
:confirm => "Anda Yakin akan menghapus data?" %>
              </td>
            <% end %>
          </tr>
        <% end %>
      </tbody>
    </table>
  </div>
  <div class="box-footer">     
    <%= will_paginate @employees %>
  </div>
</section>
```

5. Class Role

Tabel 4.21 Tabel *Class Role*

Nama Class	Role
Jenis Class	Boundary
Nama File	classRole

```
class Role < ActiveRecord::Base

end
```

6. Class FormRole

Tabel 4.22 Tabel *Class FormRole*

Nama Class	FormRole
Jenis Class	View
Nama File	classFormRole

```
<section id="dashtabs" class="grid_12">
  <div class="box-header">
    List Roles
  </div>
  <div class="box-content no-padding" id="dashtabs-pages">
    <table class="table no-border">
      <thead>
        <tr>
          <td>Name</td>
        </tr>
      </thead>
      <tbody>
        <% @roles.each do |role| %>
          <tr class="last">
            <td><%= role.name %></td>
          </tr>
        <% end %>
      </tbody>
    </table>
  </div>
</section>
```

7. Class DP3

Tabel 4.23 Tabel *Class DP3*

Nama Class	DP3
Jenis Class	Boundary
Nama File	classDP3

```
class JudmentValue < ActiveRecord::Base

  belongs_to :mutation
  belongs_to :employee

  before_save :set_average_value, :set_value_to_score

  validates_numericality_of :kesetiaan,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of  :prestasi_kerja,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of :tanggung_jawab,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of :ketaatan,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of :kejujuran,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of :kerja_sama,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of :prakarsa,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_numericality_of :kepemimpinan,
:less_than_or_equal_to => 100, :greater_than => 0
  validates_uniqueness_of :mutation_id
  validates_presence_of :mutation_id

  def set_average_value
    self.average_value = (self.kesetiaan.to_f +
self.prestasi_kerja.to_f + self.tanggung_jawab.to_f +
        self.ketaatan.to_f + self.kejujuran.to_f +
self.kerja_sama.to_f +
        self.prakarsa.to_f + self.kepemimpinan.to_f)/8
  end

  def set_value_to_score

self.mutation.alternative.scores.first.update_attributes(:v
alue => self.average_value)
  end


end
```

8. Class FormDP3

Tabel 4.24 Tabel *Class FormDP3*

Nama Class	FormDP3
Jenis Class	View
Nama File	classFormDP3

```
<section id="dashtabs" class="grid_12">
  <div class="box-header">
    DP3
    <ul class="controls">
      <li><a title="Isi DP3" class="tooltip"
href="/judment_values/new"><img
src="/img/icons/16/add.png"></a></li>
    </ul>
  </div>
  <div class="box-content no-padding" id="dashtabs-pages">
    <table class="table">
      <thead>
        <tr>
          <th>Pegawai</th>
          <th>Mutation</th>
          <th>Tanggal Pengajuan</th>
          <th>Kesetiaan</th>
          <th>Prestasi kerja</th>
          <th>Tanggung jawab</th>
          <th>Ketaatan</th>
          <th>Kejujuran</th>
          <th>Kerja sama</th>
          <th>Prakarsa</th>
          <th>Kepemimpinan</th>
          <th>Nilai Rata-rata</th>
        </tr>
      </thead>
      <tbody>
        <% @judment_values.each do |judment_value| %>
          <tr>
            <td><%=h
"#{judment_value.mutation.employee.nip} -
#{judment_value.mutation.employee.name}" %></td>
            <td><%=h
judment_value.mutation.mutation_type.name %></td>
            <td><%=h to_tanggal
judment_value.mutation.created_at %></td>
            <td><%=h judment_value.kesetiaan %></td>
            <td><%=h judment_value.prestasi_kerja %></td>
            <td><%=h judment_value.tanggung_jawab %></td>
            <td><%=h judment_value.ketaatan %></td>
            <td><%=h judment_value.kejujuran %></td>
            <td><%=h judment_value.kerja_sama %></td>
            <td><%=h judment_value.prakarsa %></td>
```

```
            <td><%=h judment_value.kepemimpinan %></td>
            <td><%=h judment_value.average_value %></td>
          </tr>
        <% end %>
      </tbody>
    </table>
  </div>
</section>
```

9. Class Mutasi

Tabel 4.25 Tabel *Class Mutasi*

Nama Class	Mutasi
Jenis Class	Boundary
Nama File	classMutasi

```
class Mutation < ActiveRecord::Base

  belongs_to :employee
  belongs_to :user
  belongs_to :mutation_type, :class_name =>
"DssApplication", :foreign_key => "dss_application_id"
  has_one :judment_value
  has_one :service_exam
  has_one :rank_selection
  has_one :alternative

  after_create :create_alternative_and_default_score
  before_create :set_default_status

  def status_name
    case self.status
    when 1
      "Belum di verifikasi"
    when 2
      "Disetujui Bagian mutasi"
    when 3
      "Ditolak Bagian mutasi"
    when 4
      "Disetujui Sekda/Mutasi"
    when 5
      "Ditolak Sekda/Mutasi"
    else
      "Belum di verifikasi"
    end
  end

  def self.list_status
    ["", "Belum di verifikasi", "Disetujui Bagian mutasi",
```

```
"Ditolak Bagian mutasi"  "Disetujui Sekda/Mutasi",
"Ditolak Sekda/Mutasi"]
  end

  def self.report_title_name(status)
    case status.to_i
    when 1
      "NOMINATIF PENGAJUAN"
    when 2
      "NOMINATIF VERIFIKASI"
    when 3
      "NOMINATIF TIDAK TERVERIFIKASI"
    when 4
      "NOMINATIF PERSETUJUAN"
    when 5
      "NOMINATIF TIDAK DISETUJUI"
    else
      "NOMINATIF PENGAJUAN"
    end
  end

  def validate
    if self.new_record?
      errors.add("Pegawai", "sedang dalam proses pengajuan
mutasi.") unless is_valid_mutation?
    end
  end

  def set_default_status
    self.status = 1
  end

  def create_alternative_and_default_score
    alt = Alternative.create(:mutation_id => self.id,
:dss_application_id => self.mutation_type.id)
    criterias = self.mutation_type.criterias
    criterias.each do |cr|
      Score.create(:alternative_id => alt.id, :criteria_id
=> cr.id, :value => 0)
    end
  end

  def is_valid_mutation?
    mutations = Mutation.find(:all, :conditions =>
["employee_id = ? AND status NOT IN (4,5)",
self.employee.id]) rescue [Employee.first]
    return mutations.empty?
  end

end
```

10. Class FormMutasi

Tabel 4.26 Tabel *Class FormMutasi*

Nama Class	FormMutasi
Jenis Class	View
Nama File	classFormMutasi

```
<% form_tag({:action => :index}, :method => :get, :class =>
"form", :id => "search-form") do %>
  <fieldset>
    <legend>Cari Mutasi</legend>
    <%= label_tag "Jenis Mutasi" %>
    <%= select_tag :mutation_type,
options_for_select(DssApplication.all.collect{|d| [d.name,
d.id]}.unshift(["Semua Type",""]), params[:mutation_type])
%>
    <br />
    <br />
    <%= label_tag "Tanggal Pengajuan" %> <%= text_field_tag
:date_from, params[:date_from], :class => "date-pick" %>
    <div style="float:left;margin:5px;">Sampai</div> <%=
text_field_tag :date_until, params[:date_until], :class =>
"date-pick" %>
    <br class="cl" /> <br />
    <div style="display:<%= params[:laporan] == "1" ?
"none" : "" %>">
      <%= label_tag "Status" %>
      <%= select_tag :status,
options_for_select(status_mutation_schoices.unshift(["Semua
Status", ""]), params[:status]) %>
      <br class="cl"/>
    </div>
    <%= hidden_field_tag :laporan, params[:laporan] %>
    <br />
    <br />
    <button class="button small" type="submit">
      Cari
    </button>
  </fieldset>
<% end %>
<section id="dashtabs" class="grid_12">
  <div class="box-header">
    Daftar Mutasi
    <% if current_user.has_role?("admin") ||
current_user.has_role?("kepala unit") %>
      <ul class="controls">
        <li><a title="Ajukan Mutasi" class="tooltip"
href="/mutations/new"><img
src="/img/icons/16/add.png"></a></li>
        <li>
          <%= link_to "<img
```

```
src='/img/icons/button/search.png'>", {
            :action => :print_previews,
            :mutation_type => params[:mutation_type],
            :status => params[:status],
            :date_from => params[:date_from],
            :date_until => params[:date_until]
          }
        %>
        </li>
      </ul>
    <% end %>
  </div>
  <div class="box-content no-padding" id="dashtabs-pages">
    <table class="table">
      <thead>
        <tr>
          <th>Nama Pegawai</th>
          <th>NIP Pegawai</th>
          <th>Jenis Mutasi</th>
          <th>Tanggal Pengajuan</th>
          <th>Status</th>
        </tr>
      </thead>
      <tbody>
        <% @mutations.each do |mutation| %>
          <tr>
            <td><%=h mutation.employee.name %></td>
            <td><%=h mutation.employee.nip %></td>
            <td><%=h mutation.mutation_type.name rescue nil
%></td>
            <td><%=h to_tanggal mutation.created_at %></td>
            <td><%=h mutation.status_name %></td>
          </tr>
        <% end %>
      </tbody>
    </table>

  </div>
</section>
```

11. *Class* Pengambilan Keputusan

Tabel 4.27 Tabel *Class* Pengambilan Keputusan

Nama Class	Pengambilan Keputusan
Jenis Class	Boundary
Nama File	classPengambilan Keputusan

```
class DssApplication < ActiveRecord::Base
  belongs_to :user
  has_many :criterias, :dependent => :destroy
  has_many :alternatives, :dependent => :destroy
  has_many :suitabilities, :dependent => :destroy
  has_many :app_attributes, :class_name => "Attribute",
:dependent => :destroy
  has_many :mutations
  accepts_nested_attributes_for :alternatives
  validates_presence_of :name#, :alternative_name

  def generate_final_score
    alternatives.each do |alternative|
      final_score = 0
      criterias.each do |criteria|
        score =
Score.find_by_alternative_id_and_criteria_id(alternative.id
, criteria.id)
        if criteria.is_benefit
          max_score = criteria.scores.max_by { |max|
max.value }
          normalization = score.value / max_score.value
          result = criteria.preference_weight *
normalization
        else
          min_score = criteria.scores.min_by { |min|
min.value }
          normalization = min_score.value / score.value
          result = criteria.preference_weight *
normalization
        end
        final_score += result
      end
      alternative.update_attribute("final_score",
final_score)
    end

    alternatives.find(:all, :order => "final_score
DESC").each_with_index do |alternative, i|
      alternative.update_attribute("rank", i+1)
    end
  end
end
```

12. *Class Form* Pengambilan Keputusan

Tabel 4.28 Tabel *Class Form* Pengambilan Keputusan

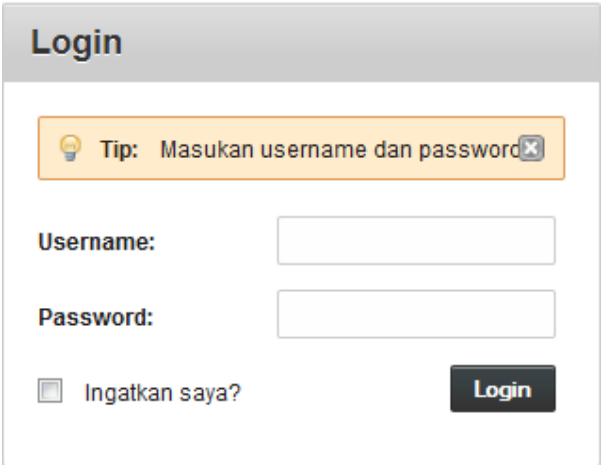
Nama Class	FormPengambilan Keputusan
Jenis Class	View
Nama File	classFormPengambilan Keputusan

```
<div class="box-header">
  Tipe Mutasi
  <ul class="controls">
    <li><a title="Buat Tipe Mutasi" class="tooltip"
href="<%= new_dss_application_path %>"><img
src="/img/icons/16/add.png"></a></li>
  </ul>
  <br class="cl"/>
</div>
<div class="box-content">
  <table class="table">
    <thead>
      <tr>
        <th>Name</th>
      </tr>
    </thead>
    <tbody>
      <% @dss_applications.each do |dss_application| %>
        <tr <%= cycle "class='odd'","" %>>
          <td><%=h dss_application.name %></td>
        </tr>
      <% end %>
    </tbody>
  </table>
</div>
```

4.1.6 User Interface

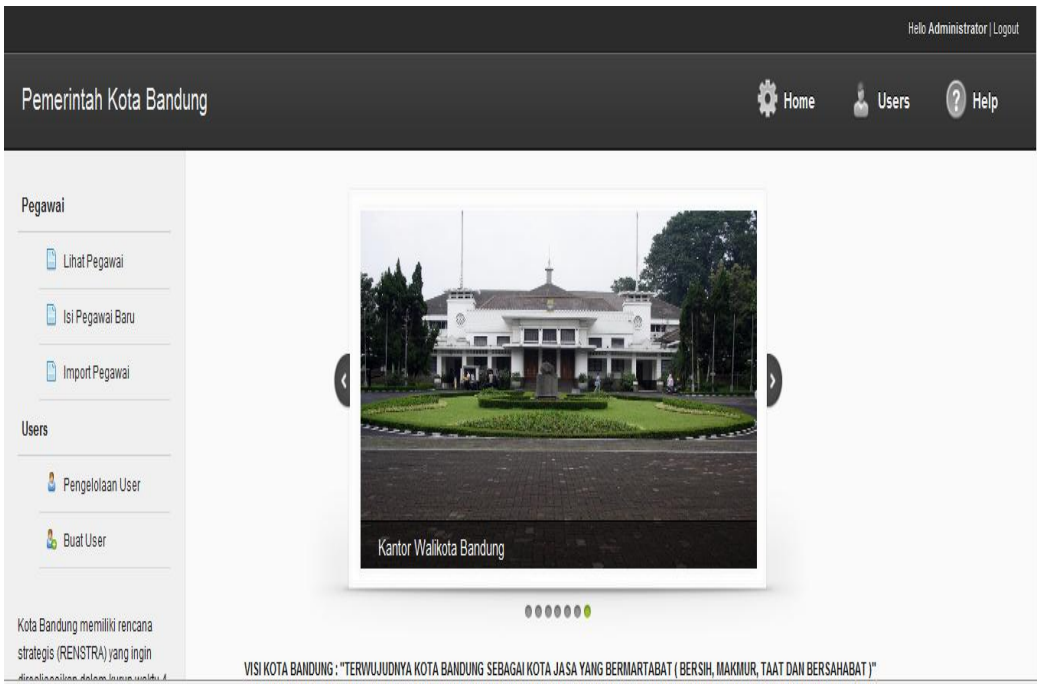
Untuk user interface pada aplikasi ini terdiri dari beberapa bagian halaman yang akan dijelaskan dibawah ini, antara lain :

i. Login



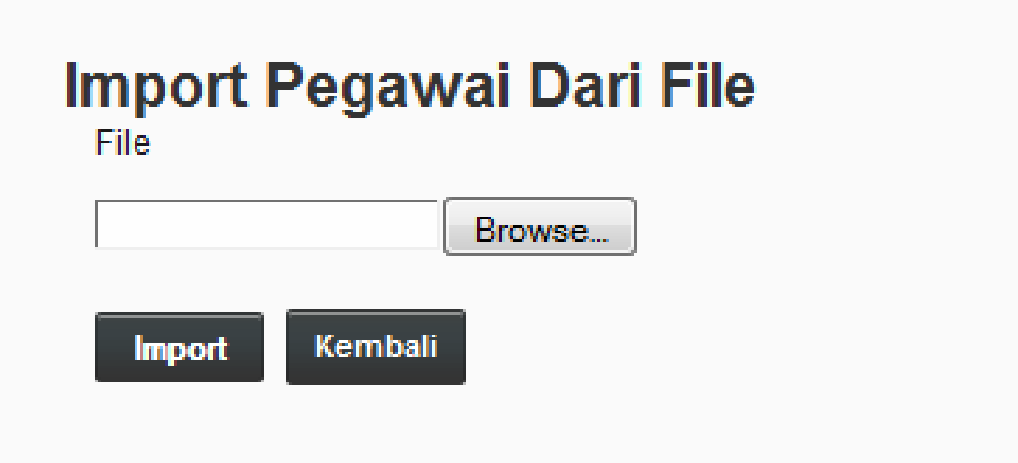
Gambar 4.1 Halaman Login

1. Halaman Utama



Gambar 4.2. Halaman Utama

2. Halaman *Import* Pegawai



Gambar 4.3 Halaman *Import* Pegawai

4.2 Pengujian Sistem

Tahap pengujian (*testing*) merupakan tahap yang menentukan apakah hasil akhir sebuah sistem memenuhi *requirement* yang telah ditentukan. Pengujian dibuat berdasarkan analisa dan perancangan yang telah dilakukan sebelumnya. Dalam tahap Pengujian ini akan ditentukan apakah sistem telah memenuhi *requirement-requirement* yang dijadikan sebagai tolok ukur dalam menentukan keberhasilan dalam pengerjaan sebuah sistem perangkat lunak. Untuk membuat Sistem Pendukung Pengambilan Keputusan Mutasi Pegawai Dengan Menggunakan Metode *Simple Additive Weighting* Pada Badan Kepegawaian Daerah Kota Bandung ini, akan ditentukan proses pengujian.

Pengujian yang dilakukan yaitu pengujian secara fungsional *(alpha)* dan beta. Metode yang digunakan dalam pengujian ini adalah pengujian *blackbox* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

4.2.1 Rencana Pengujian

Rencana pengujian adalah pengujian terhadap fungsi-fungsi yang ada di dalam sistem, apakah fungsi tersebut berfungsi sesuai yang diharapkan atau tidak. Berikut ini tabel rencana pengujian dari sistem yang dibangun :

Tabel 4.29 Tabel Rencana Pengujian

Pengujian Sistem Pendukung Pengambilan Keputusan Mutasi Pegawai dengan Menggunakan Metode Simple Additive Weighting pada Badan Kepegawaian Daerah Kota Bandung		
Item Uji	**Detail Uji**	**Jenis Uji**
Login	Login	*Black box*
Pengelolaan User	Tambah User	*Black box*
	Hapus User	*Black box*
	Edit User	*Black box*
	Ganti Password	*Black box*
	Konfirmasi Password	*Black box*
Import Pegawai	Pilih File	*Black box*
Pengajuan Mutasi	Pilih Pegawai	*Black box*
	Pilih Jenis Mutasi	*Black box*
	Pilih Status	*Black box*
Verifikasi Mutasi	Verifikasi	*Black box*
	Isi Nilai DP3	*Black box*
	Isi Nilai Ujian Dinas	*Black box*
	Isi Nilai Seleksi Jabatan	*Black box*
Pengambilan Keputusan	Kriteria Penilaian SAW	*Black box*
	Perhitungan SAW	*Black box*
	Keputusan Mutasi	*Black box*

4.2.2 Kasus dan Hasil Pengujian Alpha

Dari rencana pengujian diatas, maka dapat dilakukan pengujian *Alpha* pada sistem sebagai berikut:

Tabel 4.30 Tabel Pengujian *Login*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin Password : password	Dapat masuk ke halaman utama sesuai dengan hak aksesnya.	Dapat mengisi nama *nama* dan *password* Sesuai yang diharapakan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username :adnan Password : 12345	Tidak dapat masuk ke halaman utama dan tampilan pesan "user atau Password salah !"	User tidak dapat login	[√] diterima [ ] ditolak

Tabel 4.31Tabel Pengujian Buat *User*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
User Name : Hari Nama : Hari Suherlan Email : nicko_indy@yahoo.com Password : 13456 Confirmasi Password : 123456 Akses : admin	Dapat disimpan sebagai user name baru	Dapat mengisi seluruh isian sesuai yang diharapakan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
User Name : Hari Nama : Hari Suherlan Email : nicko_indy@yahoo.com Password : 13456 Confirmasi Password : 654321 Akses : admin	Tidak dapat disimpan sebagai user name baru	Pembuatan uset tidak dapat disimpan	[ ] diterima [√] ditolak

Tabel 4.32 Tabel Pengujian Ubah *User*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : nicko_indy@yahoo.com Password : 123456 Confirmasi password : 123456	Dapat disimpan sesuai dengan data yang di ubah	Dapat mengisi seluruh isian sesuai yang diharapakan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : nicko_indy@yahoo.com Password : 123456 Confirmasi password : 654321	Tidak dapat disimpan karena pengisian data salah	Perubahan user tidak sesuai	[ ] diterima [√] ditolak

Tabel 4.33 Tabel Pengujian *Import* Pegawai

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pegawai.xlsx	Dapat di import	Dapat di import sesuai data yang dipilih.	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pegawai.docx	Tidak dapat di import	Tidak dapat di import karena data yang di import beda type.	[ ] diterima [√] ditolak

Tabel 4.34 Tabel Pengujian Pengajuan Mutasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih Pegawai : Dadang Achadiat Pilih Jenis Mutasi : Jabatan	Data yang dipilih masuk	Data dapat di simpan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Pilih Pegawai : Null Pilih Jenis Mutasi : Jabatan	Data yang dipilih tidak dapat masuk ke pengajuan mutasi	Data tidak dapat disimpan	[ ] diterima [√] ditolak

Tabel 4.35 Tabel Pengujian Verifikasi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih : Verifikasi dengan pengisian tiga nilai sesuai dengan isiannya	Data yang dipilih masuk dan dapat disimpan	Data dapat di simpan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih: verifikasi dengan data isian null	Data yang dipilih tidak dapat masuk ke verifikasi	Data tidak dapat disimpan	[ ] diterima [√] ditolak

Tabel 4.36 Tabel Pengujian Kriteria *SAW*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi nilai kriteri dengan nilai numerik, contoh : 35	Data yang di input masuk kedalam databse	Data dapat di simpan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi nilai kriteri dengan nilai huruf, contoh : abc	Data yang di input tidak dapat masuk kedalam databse	Data tidak dapat disimpan	[ ] diterima [√] ditolak

Tabel 4.37 Tabel Pengujian Penghitungan *SAW*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Meng-genarate score dengan pegawai terisi semua nilai.	Pegawai dapat dilihat dengan score berurutan.	Data dapat dilihat dengan rangking berurutan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Meng-genarate score dengan pegawai tidak terisi semua nilai.	Pegawai masih belum bisa dilihat rankingnya	Data tidak berurutan	[ ] diterima [√] ditolak

Tabel 4.38 Tabel Pengujian Keputusan *SAW*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pegawai yang sudah di verifikasi dan di hitung nilainya bisa di update	Pegawai dapat diputuskan diterima atau di tolak.	Data dapat dilihat dengan rangking berurutan	[√] diterima [ ] ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pegawai yang belum di verifikasi	Pegawai masih belum dapat dilihat rankingnya/diverifikasi	Data tidak dapat diproses	[ ] diterima [√] ditolak

4.2.3 Kesimpulan Pengujian Alpha

Berdasarkan hasil pengujian alpha yang telah dilakukan dapat ditarik kesimpulan bahwa pada aplikasi ini sudah berjalan cukup baik, tetapi tidak menutup kemungkinan dapat terjadi kesalahan pada saat aplikasi digunakan sehingga membutuhkan proses maintenance untuk lebih mengetahui kekurangan dari aplikasi ini.

4.2.4 Pengujian Beta

Pengujian *beta* merupakan pengujian langsung kepada *user* untuk mencoba Sistem Pendukung Pengambilan Keputusan Mutasi Pegawai Dengan Menggunakan Metode *Simple Additive Weighting* Pada Badan Kepegawaian Daerah Kota Bandung dengan melakukan wawancara mengenai kepuasan *user*. Dari hasil wawancara tersebut dapat mengambil kesimpulan terhadap penilaian dari aplikasi yang telah dibuat. (Lembaran wawancara dilampirkan).

Berikut ini adalah hasil wawancara penulis dengan para pengguna aplikasi diantaranya sebagai berikut :

A. Admin

1. Nama : Arif M. Rizal
2. NIP : 198004072006041010
3. Jabatan: PELAKSANA SUB. BAGIAN UMUM

Pertanyaan :

a. Apakah bapak sebagai pelaksana pada bagian yang ini merasa terbantu dengan aplikasi ini…?

   Jawaban : Ya, cukup terbantu dengan adanya aplikasi ini, karena memudahkan dalam pendistribusian data.

b. Apakah aplikasi ini mudah untuk digunakan…?

   Jawaban : Ya, cukup mudah, karena langkah/caranya tidak terlalu rumit.

c. Apakah bapak sebagai user administrator sudah merasa nyaman dalam menggunakan aplikasi ini…?

   Jawaban : Ya, karena dalam pengelolaannya mudah untuk di operasikan.

d. Apakah dalam tampilan aplikasi ini mudah dipahami…?

Jawaban : Ya, karena tampilan dari aplikasi ini cukup sederhana.

e. Apakah setelah menggunakan aplikasi ini bapak berencana untuk memodifikasi/memperbaharui aplikasi ini…?

Jawaban : Insya Allah…, tergantung kebutuhan.

Berdasarkan hasil wawancara dengan admin, dapat disimpulkan bahwa aplikasi ini cukup membantu dalam pelaksanaan tugas dan fungsi pada bagian ini.

B. Kepala Unit

1. Nama : Amir Hamzah, S. SIP
2. NIP : 196911111998032012
3. Jabatan: PELAKSANA SUB. BIDANG PENGEMBANGAN KARIER

Pertanyaan :

a. Apakah aplikasi ini sudah membantu secara teknis pada pekerjaan bapak..?

Jawaban : Ya, cukup membantu dalam pelaksanaan keseharian tugas pokok saya.

b. Apa yang bapak rasakan setelah mencoba menggunakan aplikasi ini…?

Jawaban : merasa cukup bermanfaat dengan menggunakan aplikasi ini.

c. Apakah aplikasi ini nyaman di gunakan…?

Jawaban : Cukup nyaman karena aplikasi ini pengoperasiannya tidak rumit.

d. Apakah dalam tampilan aplikasi ini mudah dipahami…?

Jawaban : Ya, karena tampilan dari aplikasi ini cukup sederhana.

e. Apakah kecepatan pengiriman informasi data sudah optimal…?

Jawaban: Ya, sangat cepat dan tersetruktur

Berdasarkan hasil wawancara dengan admin, dapat disimpulkan bahwa aplikasi ini cukup membantu dalam pelaksanaan tugas dan fungsi pada bagian ini.

C. Bagian Mutasi

1. Nama : Rita Pujiani, S. SIP
2. NIP : 197011251998032004
3. Jabatan: PELAKSANA SUB. BIDANG MUTASI PEGAWAI

Pertanyaan :

a. Apakah aplikasi ini sudah membantu secara teknis pada pekerjaan ibu.?

Jawaban : Ya, cukup membantu dalam pelaksanaan keseharian tugas pokok saya.

b. Apakah dari uji coba aplikasi ini ibu mengalami hambatan…?

Jawaban : Tidak, karena aplikasi ini cukup familier.

c. Apakah aplikasi ini nyaman di gunakan…?

Jawaban : Cukup nyaman karena aplikasi ini pengoperasiannya tidak rumit.

d. Apakah dalam tampilan aplikasi ini mudah dipahami…?

Jawaban : Ya, karena tampilan dari aplikasi ini cukup sederhana.

e. Apakah data yang tersimpan di database sudah aman…?

Jawaban: Ya, cukup aman.

Berdasarkan hasil wawancara dengan admin, dapat disimpulkan bahwa aplikasi ini cukup membantu dalam pelaksanaan tugas dan fungsi pada bagian ini.

D. SEKDA

1. Nama : Dedi Sodikin
2. NIP : 195922111993031004
3. Jabatan: PELAKSANA SEKRETARIAT

Pertanyaan :

a. Apakah aplikasi ini sudah membantu secara teknis pada pekerjaan bapak..?

Jawaban : Ya, cukup membantu dalam pelaksanaan keseharian tugas pokok saya.

b. Apa yang bapak rasakan setelah mencoba menggunakan aplikasi ini…?

Jawaban : merasa cukup bermanfaat dengan menggunakan aplikasi ini.

c. Dalam hak akses bapak, apakah bapak mengalami kesulitan dalam aplikasi ini…?

Jawaban : Tidak, karena dalam pelaksanannya aplikasi ini cukup nyama untuk digunakan.

d. Apakah dalam tampilan aplikasi ini mudah dipahami…?

Jawaban : Ya, karena tampilan dari aplikasi ini cukup sederhana.

e. Apakah kecepatan pengiriman informasi data sudah optimal…?

Jawaban: Ya, sangat cepat.

Berdasarkan hasil wawancara dengan admin, dapat disimpulkan bahwa aplikasi ini cukup membantu dalam pelaksanaan tugas dan fungsi pada bagian ini.

4.2.5 Kesimpulan Pengujian Beta

Berdasarkan hasil wawacara dengan ke empat user diatas dapat simpulkan dari pengujian beta yang dilakukan terdapat disimpulkan bahwa Sistem Pendukung Pengambilan Keputusan Mutasi Pegawai Dengan Menggunakan Metode *Simple Additive Weighting* Pada Badan Kepegawaian Daerah Kota Bandung ini telah dibuat sesuai dengan harapan *user*, diantaranya sebagai berikut :Kemudahan dalam penggunaan sistem

1. Kemudahan dalam pengolahan data pegawai
2. Kemudahan dalam pembuatan laporan jenis mutasi dan cukup sesuai dengan kebutuhan.