

Index.php

```
<style>
.order-card {
  color: #fff;
}

.bg-c-blue {
  background: linear-
gradient(45deg,#4099ff,#73b4ff);
}

.bg-c-green {
  background: linear-
gradient(45deg,#2ed8b6,#59e0c5);
}

.bg-c-yellow {
  background: linear-
gradient(45deg,#FFB64D,#ffc880);
}

.bg-c-pink {
  background: linear-
gradient(45deg,#FF5370,#ff869a);
}

.bg-c-default {
  /* 525E75 */
  background: linear-
gradient(45deg,#525E75,#76839c);
}

.card {
  border-radius: 5px;
  -webkit-box-shadow: 0 1px 2.94px
0.06px rgba(4,26,55,0.16);
  box-shadow: 0 1px 2.94px 0.06px
rgba(4,26,55,0.16);
  border: none;
  margin-bottom: 30px;
  -webkit-transition: all 0.3s ease-in-
out;
  transition: all 0.3s ease-in-out;
}

.card .card-block {
  padding: 25px;
}

.order-card i {
  font-size: 26px;
}

.f-left {
  float: left;
}

.f-right {
  float: right;
}

<style>
/* Gaya untuk container visi dan
misi */
.visi-misi-container {
  background-color: #f2f2f2;
  padding: 20px;
  border: 1px solid #ccc;
  border-radius: 5px;
}

/* Gaya untuk judul visi dan misi */
.visi-misi-title {
  color: #333;
  font-size: 24px;
  font-weight: bold;
}

/* Gaya untuk daftar misi */
.misi-list {
  list-style-type: decimal;
  padding-left: 20px;
}

/* Gaya untuk setiap item dalam
daftar misi */
.misi-list-item {
  margin-bottom: 10px;
}
```

```

    }
    .visi-container {
      background-color: #f2f2f2;
      padding: 20px;
      border: 1px solid #ccc;
      border-radius: 5px;
    }

    /* Gaya (style) untuk heading visi */
    .visi-container h2 {
      color: #333;
    }

    /* Gaya (style) untuk paragraf visi */
    .visi-container p {
      color: #666;
    }

    .visi-misi-container {
      background-color: #f2f2f2;
      padding: 20px;
      border: 1px solid #ccc;
      border-radius: 5px;
    }
  </style>
  <div class="row data-div filter-
  dashboard page-data" data-
  pagemodule="dashboard" data-
  hakakses="<?=$this->session-
  >hak_akses; ?>" data-url="<?=$
  base_url() ?>" data-
  modul="dashboard">
    <div class="col-sm-12">
      <span class="page-
  title">PEMERINTAHAN DESA
  MARDIHARJO - <?=$this->main-
  >tanggal("D, d M Y"); ?></span>
      <!-- <p class="text-muted page-
  title-alt">Welcome to Ubold admin
  panel !</p> -->
      <div class="portlet-widgets pull-
  right margin-bot-10">

```

```

      <span class="btn-group">
        </span>
      </div>
    </div>
  </div>
  <br>

  <div class="row">
    <div class="col-md-6">
      <div class="chart-container">
        <div class="visi-container">
          <h2>Visi:</h2>
          <p>Desa Mardiharjo memiliki visi
          “Kebersamaan dalam Membangun
          Demi Desa Mardiharjo yang Lebih
          Sejahtera”.</p>
        </div>
      </div>
    </div>
    <div class="col-md-6">
      <div class="chart-container">
        <div class="visi-misi-
  container">
          <h2 class="visi-misi-
  title">Misi:</h2>
          <ul class="misi-list">
            <li class="misi-list-
  item">Meningkatkan pembangunan
  infrastruktur yang mendukung
  perekonomian desa.</li>
            <li class="misi-list-
  item">Meningkatkan pembangunan
  dibidang kesehatan untuk mendorong
  derajat kesehatan masyarakat agar
  dapat bekerja lebbih optimal dan
  memiliki harapan hidup yang lebih
  panjang.</li>
            <li class="misi-list-
  item">Meningkatkan pembangunan
  dibidang pendidikan untuk mendukung
  peningkatan kualitas sumber daya

```

manusia agar memiliki kecerdasan dan daya saing yang lebih baik.

<li class="misi-list-item">Meningkatkan pembangunan ekonomi dengan mendorong semakin tumbuh dan berkembangnya pembangunan dibidang pertanian dalam arti luas, industri, perdagangan, dan pariwisata.

<li class="misi-list-item">Menciptakan tata kelola pemerintahan yang baik (Good Governace) berdasarkan demokratis, transparasi, penegakan hukum, kesetaraan gender dan mengutamakan pelayanan kepada masyarakat.

```
</ul>
</div>
</div>
</div>
</div>
</div>
<?php $this->load-
>view("backend/modal/modal_import"
); ?>
<?php $this->load-
>view("backend/dashboard/modal");
?>
<?php $this->load-
>view("backend/modal/modal_vehicle
_sell"); ?>
<!-- Plugins For This Page -->
<script src="<?= base_url();
?>assets/plugins/chart.js/chart.min.js">
</script>
<script src="<?=
base_url('aset/js/backend/dashboard.js'
); ?>"></script>
```

Bobot.php

```
<?php
defined('BASEPATH') OR exit('No
direct script access allowed');
```

```
class Bobot extends CI_Controller {
    public function __construct()
    {
        parent::__construct();
        $this->load-
>helper('url');
        $this->load-
>model("M_bobot",'bobot');
        $this->main-
>cek_session();
        $this->CompanyID =
$this->session->CompanyID;
    }
    public function index()
    {
        $MenuID
            = $this->main-
>GetMenuID();
        $read
            = $this->main-
>read($MenuID);
        if($read == 0){
            redirect(); }
        $vendor_tambah
            = $this->main-
>menu_tambah($MenuID);
        if($vendor_tambah >
0):
            $tambah = '<button
type="button" class="btn btn-default
waves-effect waves-light"
onclick="tambah()" ><span
class="btn-label"><i class="fa fa-
plus"></i></span> Tambah
Data</button>';
        else:
            $tambah = "";
        endif;
        $data['title'] =
'Bobot';
        $data['tambah'] =
$tambah;
```

```

                $data['modal']      =      Data" onclick="edit(this)" data-
'backend/bobot/modal';              id="".$a->BobotID."" data-
                $data['content']    =      method="view">'. $btn_view.'</a>';
'backend/bobot/list';                if($view_ubah >
                $data['modul']      =      0):
'bobot';                              $edit = '<a
                $data['MenuID']     =      href="javascript:void(0)" title="Edit"
$MenuID;                              onclick="edit(this)" data-id="".$a-
                $this->load-        >BobotID."" data-
>view('backend/index',$data);        method="edit">'. $btn_edit.'</a>';
                }                  endif;
                public function list_data($page
= "")                                  if($view_hapus
                {                  > 0):
                    $MenuID = $this-
                    >input->post("MenuID");
                    $list = $this->bobot-
                    >get_datatables($page);
                    $data = array();
                    $no =
                    $_POST['start'];
                    $i = 1;
                    $btn_delete = $this-
                    >lang->line('btn_delete');
                    $btn_undelete = $this-
                    >lang->line('btn_undelete');
                    $btn_edit =
                    $this->lang->line('btn_edit');
                    $btn_view =
                    $this->lang->line('btn_view');
                    $view_ubah =
                    $this->main->menu_ubah($MenuID);
                    $view_hapus = $this-
                    >main->menu_hapus($MenuID);
                    foreach ($list as $a) {
                        $ID = $a-
                        >BobotID;
                        $view = "";
                        $edit = "";
                        $delete = "";
                        $active = "";
                        $view = '<a
                        href="javascript:void(0)" title="View

```

```

        $action .=                                     $Status = true;
        "<li>$edit</li>";                               $Data = array(
        $action .=                                     "BobotID"    => $a-
        "<li>$delete</li>";                             >BobotID,
        $action .=                                     "Name"
        '</ul>';                                       => $a->Name,
        '</div>';                                       "Value"
        $th_no = "<span
class='th-no'>".$i++."</span>";                       => $this->main-
        $row =                                         >formatComa($a->Value),
        array();                                       "NextID"    => $this-
        $row[] = $no;                                  >bobot->get_id("next",$a->BobotID),
        $row[] =                                       "PrevID"
        $action;                                       => $this->bobot-
        // $row[] =                                     >get_id("prev",$a->BobotID),
        $a->Value;                                       );
        $data[] = $row;                                endif;
        }                                               $output = array(
        $output = array(                                "HakAksesID"=> $this-
            "draw"                                       >session->HakAksesID,
            => $_POST['draw'],                             "Data"
            "recordsTotal"                               => $Data,
            => $this->bobot-                               "Status"
            >count_all($page),                             => $Status,
            "recordsFiltered" => $this-                   "Message"
            >bobot->count_filtered($page),                 => $Message,
            "data"                                       );
            => $data,                                       $this->main-
        );                                               >echoJson($output);
        echo                                           }
        json_encode($output);                           public function
    }                                                 save($save_method = "")
    public function edit($id)                           {
    {
        $Data = array();
        $Status = false;
        $Message = "";
        $a = $this-
        >bobot->get_by_id($id);
        if($a):

```

```

        // $Value      = $this-
>input->post("Value");

        $Data = array(
            "Name"
        => $Name,
            // "Value"
        => $this->main-
>formatDecimal($Value),
        );
        if($save_method ==
"add"):
            $Data["Active"]
        = 1;
            $BobotID
        = $this->bobot-
>save($Data);
            $Status
        = true;
            $Message
        = "Data berhasil
disimpan";
            elseif($save_method ==
"update"):
                $where
            =
array("BobotID"=>$BobotID);
                $Update
            = $this->bobot-
>update($where,$Data);
                $Status
            = true;
                $Message
            = "Data berhasil disimpan";
                endif;
                $a      = $this-
>bobot->get_by_id($BobotID);
                $ID    = $a->BobotID;
                $output = array(
                    "HakAkses" =>
$this->session->HakAkses,
                    "BobotID" =>
$BobotID,
                    "NextID"
                => $this->bobot-
>get_id("next",$BobotID),
                    "PrevID"
                => $this->bobot-
>get_id("prev",$BobotID),
                    "Status"
                => $Status,
                    "Message"
                => $Message
                );
                echo $this->main-
>echoJson($output);
            }
            public function
delete($id="", $status="")
            {
                $Action = $this->input-
>post("Action");
                $Active = 0;
                if($Action):
                    $ID = $this-
>input->post("ID");
                    if($Action ==
"undelete"):
                        $Active =
1;
                    endif;
                    $data = array(
                        "Active"
                    => $Active,
                );
                    foreach($ID as
$a):
                        $this-
>bobot->update(array('BobotID' =>
$a), $data);
                    endforeach;
                else:
                    if($status ==
"active"):
                        $Active =
1;

```

```

        endif;
        $data = array(
            "Active"
=> $active,
        );
        $this->bobot-
>update(array('BobotID' => $id),
        $data);
        endif;
        echo
        json_encode(array("Status" =>
        TRUE,"Message" => $this->lang-
>line("success_transaction"));
    }
}

```

M_bobot.php

```

<?php
defined('BASEPATH') or exit('No
direct script access allowed');

class M_bobot extends CI_Model
{
    var $table      = 'MT_Bobot';
    var $column =
array("A.BobotID","A.Name",
"A.Value");
    var $order      = "A.Active
DESC, A.BobotID ASC";
    public function __construct()
    {
        parent::__construct();
        $this->load-
>database();
        $this->CompanyID =
$this->session->CompanyID;
    }
    private function
_get_datatables_query($page = "")
    {
        $this->db-
>select("A.*");

```

```

        $this->db->from($this-
>table . " as A");
        $this->db-
>where("A.Active", 1);
        $i = 0;
        foreach ($this->column
as $item) :
            if
($ _POST['search']['value']) :
                if ($i
==== 0) :
                    $this->db->group_start();
                    $this->db->like($item,
$_POST['search']['value']);
                else :
                    $this->db->or_like($item,
$_POST['search']['value']);
                endif;
            if
(count($this->column) - 1 == $i) :
                $this->db->group_end();
            endif;
        endif;
        $column[$i] =
$item;
        $i++;
    endforeach;
    if
(isset($_POST['order'])) :
        $this->db-
>order_by($column[$_POST['order']]['
0']['column'],
$_POST['order']['0']['dir']);
    elseif (isset($this-
>order)) :
        $order = $this-
>order;
        $this->db-
>order_by($order);

```

```

        endif;
    }
    public function
get_datatables($page = "")
    {
        $this-
>_get_datatables_query($page);
        if ($_POST['length'] !=
-1)
            $this->db-
>limit($_POST['length'],
$_POST['start']);
        $query = $this->db-
>get();
        return $query->result();
    }
    public function
count_filtered($page = "")
    {
        $this-
>_get_datatables_query($page);
        $query = $this->db-
>get();
        return $query-
>num_rows();
    }
    public function
count_all($page = "")
    {
        $this->db-
>where("A.Active", 1);
        $this->db->from($this-
>table . " as A");
        return $this->db-
>count_all_results();
    }
    public function get_by_id($id,
$method = "")
    {
        if ($method == "code")
:
            $this->db-
>select("BobotID");

```

```

        endif;
        $this->db->from($this-
>table);
        $this->db-
>where('BobotID', $id);
        $query = $this->db-
>get();
        return $query->row();
    }
    public function save($data)
    {
        $this->api-
>Log(array("Type" => "CREATE",
"Name" => "MasterCategory",
"DetailData" => $data));
        $this->db-
>set("UserAdd", $this->session-
>Name);
        $this->db-
>set("DateAdd", date("Y-m-d H:i:s"));
        $this->db->insert($this-
>table, $data);
        return $this->db-
>insert_id();
    }
    public function update($where,
$data)
    {
        $this->api-
>Log(array("Type" => $this->api-
>TypeLog($data), "Name" =>
"MasterCategory", "DetailData" =>
$data));
        $this->db-
>set("UserCh", $this->session-
>Name);
        $this->db-
>set("DateCh", date("Y-m-d H:i:s"));
        $this->db-
>update($this->table, $data, $where);

```

```

        return $this->db-
>affected_rows();
    }
    public function
delete_by_id($id)
    {
        $this->db-
>where('BobotID', $id);
        $this->db-
>delete($this->table);
    }
    public function
get_id($method, $id)
    {
        $new_id = 0;
        $this->db-
>select("BobotID");
        $this->db->from($this-
>table);
        if ($method == "next") :
            $this->db-
>where("BobotID >", $id);
            $this->db-
>order_by("BobotID", "ASC");
        else :
            $this->db-
>where("BobotID <", $id);
            $this->db-
>order_by("BobotID", "DESC");
        endif;
        $this->db->limit(1);
        $query = $this->db-
>get();
        $a = $query->row();
        if ($a) :
            $new_id = $a-
>BobotID;
        endif;
        return $new_id;
    }
}

```

List.php

```

<div class="row page page-data"
data-page_name="<?= $title; ?>" data-
modul="<?= $modul; ?>" data-
menuid="<?= $MenuID; ?>" data-
aplikasi="<?= $this->session->App
?>" data-url="<?= base_url(); ?>">
    <div class="col-sm-12">
        <?php $this->load->view($modul);
        ?>
    </div>
    <div class="col-sm-12">
        <div class="div-table">
            <div class="panel panel-color
panel-custom">
                <div class="panel-heading">
                    <span class="panel-
title">Master <?= $title; ?></span>
                </div>
                <div class="panel-body table-
responsive">
                    <div class="div-btn btn-
group">
                        <?= $tambah; ?>
                    </div>
                    <div class="div-btn btn-
group pull-right">
                        <button class="btn btn-
outline btn-white"
onclick="reload_table()"
type="button">Reload</button>
                    </div>
                    <table id="table" class="table
table-hovered full-width table-pointer"
style="width: 100%">
                        <thead>
                            <tr>
                                <th>No.</th>
                                <th>Nama Bobot</th>
                                <!-- <th>Nilai
Bobot</th> -->
                            </tr>
                        </thead>
                    </table>
                </div>
            </div>
        </div>
    </div>

```

```

        </tbody>
    </table>
</div>
</div>
</div>
</div>
</div>
<!-- Plugins For This Page -->
<link href=""?="
base_url('aset/plugin/datepicker/boots
rap-datepicker3.css'); ?>"
rel="stylesheet"/>
<script src=""?="
base_url('aset/js/backend/bobot.js');
?>"></script>
<script src=""?="
base_url('aset/plugin/datepicker/boots
rap-datepicker.min.js'); ?>"></script>

```

Modal.php

```

<div class="panel-form sembunyi">
    <h3 class="form-title m-b-
20"></h3>
    <div class="approval-status-msg
sembunyi">Informasi</div>
    <form id="form" method="post"
class="form-design-1"
enctype="multipart/form-data">
        <div class="row">
            <div class="col-sm-6">
                <div class="panel panel-color
panel-custom">
                    <div class="panel-heading
clearfix">
                        <span class="btn-group
div-btn-prev">
                            <a href="javascript:;"
id="BtnPrev"
onclick="div_form('prev_data')"
class="btn btn-white btn-outline btn-
xs"><i class="ti-angle-left"></i></a>
                            </span>

```

```

                        <span class="panel-
title">Data <?=" $title ?"></span>
                            <span class="btn-group
pull-right div-btn-next">
                                <a href="javascript:;"
id="BtnNext"
onclick="div_form('next_data')"
class="btn btn-white btn-outline btn-
xs"><i class="ti-angle-right"></i></a>
                                </span>
                            </div>
                        <div class="panel-body
table-responsive">
                            <input type="hidden"
name="RobotID">
                            <div class="row">
                                <div class="col-sm-12">
                                    <div class="row">
                                        <div class="col-sm-
12">
                                            <div class="form-
group">
                                                <label
class="control-label">Nama
Robot</label>
                                                <input
name="Name" id="Name" type="text"
class="form-control"
maxlength="100">
                                                <span
class="help-block"></span>
                                            </div>
                                        </div>
                                    </div>
                                <div class="col-sm-
12" hidden>
                                    <div class="form-
group">
                                        <label
class="control-label">Nilai
Robot</label>
                                        <input
name="Value" id="Value" type="text"
class="form-control fist-input"

```



```

                $btn_delete = $this->
>lang->line('btn_delete');
                $btn_undelete = $this->
>lang->line('btn_undelete');
                $btn_edit =
$this->lang->line('btn_edit');
                $btn_view =
$this->lang->line('btn_view');
                $view_ubah =
$this->main->menu_ubah($MenuID);
                $view_hapus = $this->
>main->menu_hapus($MenuID);
                foreach ($list as $a) {
>TestingID;
                    $view = "";
                    $edit = "";
                    $delete = "";
                    $active = "";

                    $view = '<a
href="javascript:void(0)" title="View
Data" onclick="edit(this)" data-
id="'. $a->TestingID.'" data-
method="view">'. $btn_view.'</a>';
                    if($view_ubah >
0):
                        $edit = '<a
href="javascript:void(0)" title="Edit"
onclick="edit(this)" data-id="'. $a-
>TestingID.'" data-
method="edit">'. $btn_edit.'</a>';
                        endif;
                        if($view_hapus
> 0):
                            if($a->Active == 1):
                                $delete = '<a
href="javascript:void(0)" title="Delete
Data" onclick="hapus('.$a-
>TestingID.'"')">'. $btn_delete.'</a>';
                                else:
                                    $active = '<i
class="fa fa-trash pull-right"></i>';

```

```

                $delete = '<a
href="javascript:void(0)"
title="Undelete Data"
onclick="active('.$a-
>TestingID.'"')">'. $btn_undelete.'</a
>';
                endif;
                endif;
                $no++;

                $action = '<div
class="dropdown">';
                $action .=
'<span href="javascript:;"
class="dropdown-toggle"
type="button" data-
toggle="dropdown">'. $active.$a-
>Name.'</span>';
                $action .= '<ul
class="dropdown-menu">';
                $action .=
"<li>$view</li>";
                $action .=
"<li>$edit</li>";
                $action .=
"<li>$delete</li>";
                $action .=
'</ul>';
                $action .=
'</div>';
                $th_no = "<span
class='th-no'>". $i++."</span>";

                $status = $a-
>Status == '1' ? 'Tidak Mampu' :
'Mampu';
                $predict = $a-
>Predict == '1' ? 'Tidak Mampu' :
'Mampu';

                $row =
array();
                $row[] = $no;

```

```

        $row[] = $a-
>Name;
        $row[] = $a-
>PekerjaanName;
        $row[] = $a-
>TanggungName;
        $row[] = $a-
>PenghasilanName;
        $row[] =
$status;
        $row[] =
$predict;
        $data[] = $row;
    }
    $output = array(
        "draw"
    => $_POST['draw'],
        "recordsTotal"
    => $this->akurasi-
>count_all($page),

        "recordsFiltered" => $this-
>akurasi->count_filtered($page),
        "data"
    => $data,
    );
    echo
json_encode($output);
}
public function calculate(){
    $actualLabels = array();
    $predictedLabels =
array();

    $data = $this-
>akurasi->getData();
    $training = $this-
>akurasi->getTraining();

    foreach($data as $k =>
$v){
        $actual =
isset($training[$k]) ? $training[$k]-

```

```

>Status == '1' ? 'Tidak Mampu' :
'Mampu' : 'Tidak Mampu';
        //
        $actualLabels[] = $v->Status == '1' ?
'Tidak Mampu' : 'Mampu';
        $actualLabels[]
= $actual;

        $predictedLabels[] = $v-
>Predict == '1' ? 'Tidak Mampu' :
'Mampu';
    }

    $confusionMatrix =
ConfusionMatrix::compute($actualLab
els, $predictedLabels);

    //
    if($confusionMatrix[0][0] < 0 &&
$confusionMatrix[0][1] < 0 &&
$confusionMatrix[1][0] < 0 &&
$confusionMatrix[1][1] < 0)
        // {
            $truePositive =
$confusionMatrix[0][0];
            $falseNegative
= $confusionMatrix[0][1];
            $falsePositive =
$confusionMatrix[1][0];
            $trueNegative =
$confusionMatrix[1][1];

            $sensitivity =
$truePositive / ($truePositive +
$falseNegative);
            $specificity =
$trueNegative / ($trueNegative +
$falsePositive);
            $precision =
$truePositive / ($truePositive +
$falsePositive);
            $accuracy =
($truePositive + $trueNegative) /

```

```

array_sum(array_map('array_sum',
$confusionMatrix));

        $result = array(
            "status"
=> true,

        "sensitivity" => $sensitivity,

        "specificity" => $specificity,

        "precision" => $precision,

        "accuracy" => $accuracy
        );
        // } else {
        //     $result = array(
        //         "status"
=> false
        //     );
        // }
        echo
json_encode($result);
    }
}

```

M_Akurasi.php

```

<?php
defined('BASEPATH') or exit('No
direct script access allowed');

class M_akurasi extends CI_Model
{
    var $table      = 'testing';
    var $column =
array("A.TestingID","A.Name",
"ut_pekerjaan.Name","ut_tanggung.
Name","ut_penghasilan.Name","A.Sta
tus","A.Predict");
    var $order      = "A.Active
DESC, A.TestingID ASC";
    public function __construct()
    {

```

```

        parent::__construct();
        $this->load-
>database();
        $this->CompanyID =
$this->session->CompanyID;
    }
    private function
_get_datatables_query($page = "")
    {
        $this->db->select("A.*,

            ut_penghasilan.name as
PenghasilanName,

            ut_tanggung.Name as
TanggungName,

            ut_pekerjaan.Name as
PekerjaanName
        ");
        $this->db->from($this-
>table . " as A");
        $this->db-
>join("ut_penghasilan","ut_penghasila
n.ID = A.PenghasilanID","LEFT");
        $this->db-
>join("ut_tanggung","ut_tanggung.
n.ID = A.TanggungID","LEFT");
        $this->db-
>join("ut_pekerjaan","ut_pekerjaan.ID
= A.PekerjaanID","LEFT");
        $this->db-
>where("A.Active", 1);
        $i = 0;
        foreach ($this->column
as $item) :
            if
($ _POST['search']['value']) :
                if ($i
=== 0) :
                    $this->db->group_start();

```

```

        $this->db->like($item,
$_POST['search']['value']);
        else :
            $this->db->or_like($item,
$_POST['search']['value']);
            endif;
            if
(count($this->column) - 1 == $i) :

                $this->db->group_end();
                endif;
            endif;
            $column[$i] =
$item;
            $i++;
        endforeach;
        if
(isset($_POST['order'])) :
            $this->db-
>order_by($column[$_POST['order']]['
0']['column'],
$_POST['order']['0']['dir']);
            elseif (isset($this-
>order)) :
                $order = $this-
>order;
                $this->db-
>order_by($order);
            endif;
        }
        public function
get_datatables($page = "")
        {
            $this-
>_get_datatables_query($page);
            if ($_POST['length'] !=
-1)
                $this->db-
>limit($_POST['length'],
$_POST['start']);
                $query = $this->db-
>get();
                return $query->result();
            }
        public function
count_filtered($page = "")
        {
            $this-
>_get_datatables_query($page);
            $query = $this->db-
>get();
            return $query-
>num_rows();
        }
        public function
count_all($page = "")
        {
            $this->db-
>where("A.Active", 1);
            $this->db->from($this-
>table . " as A");
            return $this->db-
>count_all_results();
        }
        public function getData()
        {
            $this->db-
>where("A.Active", 1);
            $this->db->from($this-
>table . " as A");
            $query = $this->db-
>get();
            return $query->result();
        }
        public function getTraining()
        {
            $this->db-
>where("A.Active", 1);
            $this->db-
>from("training as A");
            $query = $this->db-
>get();
            return $query->result();
        }
    }
}

```

```

    }
}

```

List.php

```

<div class="row page page-data" data-
page_name="<?= $title; ?>" data-
modul="<?= $modul; ?>" data-
menuid="<?= $MenuID; ?>" data-
aplikasi="<?= $this->session->App
?>" data-url="<?= base_url(); ?>">
  <div class="col-sm-12">
    <?php $this->load->view($modal);
?>
  </div>
  <div class="col-sm-12">
    <div class="div-table">
      <div class="panel panel-color
panel-custom">
        <div class="panel-heading">
          <span class="panel-
title"><?= $title; ?></span>
        </div>
        <div class="panel-body table-
responsive">
          <div class="div-btn btn-
group">
            <button type="button"
class="btn btn-default waves-effect
waves-light" onclick="hitung()">
Hitung</button>
          </div>
          <div class="div-btn btn-
group">
            <button type="button"
class="btn btn-default waves-effect
waves-light" onclick="reset()">
Reset</button>
          </div>

          <div class="div-btn btn-
group pull-right">

```

```

            <button type="button"
class="btn btn-default waves-effect
waves-light" id="export-pdf"> Export
PDF</button>
          </div>
        </div>
        <div class="col-sm-12"
style="margin-bottom: 20px;margin-
left:-7px !important;">
          <div class="col-sm-6">
            <p>Sensitivity : <span
id="sensitivity">0</span> </p>
          </div>
          <div class="col-sm-6">
            <p>Specificity : <span
id="specificity">0</span> </p>
          </div>
          <div class="col-sm-6">
            <p>Precision : <span
id="precision">0</span> </p>
          </div>
          <div class="col-sm-6">
            <p>Accuracy : <span
id="accuracy">0</span></p>
          </div>
          </div>
          <table id="table" class="table
table-hovered full-width table-pointer"
style="width: 100% ">
            <thead>
              <tr>
                <th>No.</th>
                <th>Nama</th>
                <th>Pekerjaan</th>
                <th>Tanggung</th>
                <th>Penghasilan</th>
                <th>Status</th>
                <th>Predict</th>
              </tr>
            </thead>
            <tbody>
            </tbody>
          </table>
        </div>

```

```

    </div>
  </div>
</div>
<!-- Plugins For This Page -->

<script src="<?=  

base_url('aset/plugin/tambahan/pdfma  

ke.min.js'); ?>"></script>
<script src="<?=  

base_url('aset/plugin/tambahan/vfs_fo  

nts.js'); ?>"></script>

<link href="<?=  

base_url('aset/plugin/datepicker/bootst  

rap-datepicker3.css'); ?>"  

rel="stylesheet" />
<script src="<?=  

base_url('aset/js/backend/akurasi.js');  

?>"></script>
<script src="<?=  

base_url('aset/plugin/datepicker/bootst  

rap-datepicker.min.js'); ?>"></script>

```

Modal.php

```

<div class="panel-form sembunyi">
  <h3 class="form-title m-b-  

20"></h3>
  <div class="approval-status-msg  

sembunyi">Informasi</div>
  <form id="form" method="post"  

class="form-design-1"  

enctype="multipart/form-data">
    <div class="row">
      <div class="col-sm-6">
        <div class="panel panel-color  

panel-custom">
          <div class="panel-heading  

clearfix">
            <span class="btn-group  

div-btn-prev">

```

```

      <a href="javascript;"  

id="BtnPrev"  

onclick="div_form('prev_data')"  

class="btn btn-white btn-outline btn-  

xs"><i class="ti-angle-left"></i></a>
    </span>
    <span class="panel-  

title">Data <?=$title ?></span>
    <span class="btn-group  

pull-right div-btn-next">
      <a href="javascript;"  

id="BtnNext"  

onclick="div_form('next_data')"  

class="btn btn-white btn-outline btn-  

xs"><i class="ti-angle-right"></i></a>
    </span>
  </div>
  <div class="panel-body  

table-responsive">
    <input type="hidden"  

name="TrainingID">
    <div class="row">
      <div class="col-sm-12">
        <div class="row">
          <div class="col-sm-  

12">
            <div class="form-  

group">
              <label  

class="control-label">Nama</label>
              <input  

name="Name" id="Name" type="text"  

class="form-control"  

maxlength="100">
              <span  

class="help-block"></span>
            </div>
          </div>
        </div>
      <div class="col-sm-  

12">
        <div class="form-  

group">

```



```

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

```

Pekerjaan.php

```

<?php
defined('BASEPATH') OR exit('No
direct script access allowed');

class Pekerjaan extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load-
>helper('url');
        $this->load-
>model("M_pekerjaan",'pekerjaan');
        $this->main-
>cek_session();
        $this->CompanyID =
$this->session->CompanyID;
    }
    public function index()
    {
        $MenuID
            = $this->main-
>GetMenuID();
        $read
            = $this->main-
>read($MenuID);
        if($read == 0){
            redirect(); }
        $vendor_tambah
            = $this->main-
>menu_tambah($MenuID);
        if($vendor_tambah >
0):
            $tambah = '<button
type="button" class="btn btn-default
waves-effect waves-light"

```

```

onclick="tambah()" ><span
class="btn-label"><i class="fa fa-
plus"></i></span> Tambah
Data</button>';
        else:
            $tambah = "";
        endif;
        $data['title'] = 'Bobot
Pekerjaan';
        $data['tambah'] =
$tambah;
        $data['modal'] =
'backend/pekerjaan/modal';
        $data['content'] =
'backend/pekerjaan/list';
        $data['modul'] =
'pekerjaan';
        $data['MenuID'] =
$MenuID;
        $this->load-
>view('backend/index',$data);
    }
    public function list_data($page
="")
    {
        $MenuID = $this-
>input->post("MenuID");
        $list = $this-
>pekerjaan->get_datatables($page);
        $data = array();
        $no =
$_POST['start'];
        $i = 1;
        $btn_delete = $this-
>lang->line('btn_delete');
        $btn_undelete = $this-
>lang->line('btn_undelete');
        $btn_edit =
$this->lang->line('btn_edit');
        $btn_view =
$this->lang->line('btn_view');
        $view_ubah =
$this->main->menu_ubah($MenuID);

```

```

                $view_hapus = $this->
main->menu_hapus($MenuID);
                foreach ($list as $a) {
                    $ID = $a->ID;

                    $view = "";
                    $edit = "";
                    $delete = "";
                    $active = "";

                    $view = '<a
href="javascript:void(0)" title="View
Data" onclick="edit(this)" data-
id="'. $a->ID.'" data-
method="view">'. $btn_view.'</a>';
                    if($view_ubah >
0):
                        $edit = '<a
href="javascript:void(0)" title="Edit"
onclick="edit(this)" data-id="'. $a-
>ID.'" data-
method="edit">'. $btn_edit.'</a>';
                        endif;
                        if($view_hapus
> 0):
                            if($a->Active == 1):
                                $delete = '<a
href="javascript:void(0)" title="Delete
Data" onclick="hapus(' . $a->ID.
' . $a->ID. ' . ')>'. $btn_delete.'</a>';
                                else:
                                    $active = '<i
class="fa fa-trash pull-right"></i>';
                                    $delete = '<a
href="javascript:void(0)"
title="Undelete Data"
onclick="active(' . $a->ID.
' . ')>'. $btn_undelete.'</a>';
                                    endif;
                                    endif;
                                    $no++;

                                    $action = '<div
class="dropdown">';

```

```

                $action .=
'<span href="javascript:;"
class="dropdown-toggle"
type="button" data-
toggle="dropdown">'. $active. $a-
>Name.'</span>';
                $action .= '<ul
class="dropdown-menu">';
                $action .=
" <li>$view</li>";
                $action .=
" <li>$edit</li>";
                $action .=
" <li>$delete</li>";
                $action .=
'</ul>';
                $action .=
'</div>';
                $th_no = "<span
class='th-no'>". $i++. "</span>";

                $row =
array();
                $row[] = $no;
                $row[] =
$action;
                $row[] = $a-
>Bobot;
                $data[] = $row;
            }
            $output = array(
                "draw"
=> $_POST['draw'],
                "recordsTotal"
=> $this->pekerjaan-
>count_all($page),
                "recordsFiltered" => $this-
>pekerjaan->count_filtered($page),
                "data"
=> $data,
            );

```

```

        echo
        json_encode($output);
    }
    public function edit($id)
    {
        $Data = array();
        $Status = false;
        $Message = "";
        $a = $this->pekerjaan->get_by_id($id);
        if($a):
            $Status = true;
            $Data = array(
                "ID"
                => $a->ID,
                "Name"
                => $a->Name,
                "Bobot"
                => $this->main->formatComa($a->Bobot),
                "NextID" => $this->pekerjaan->get_id("next",$a->ID),
                "PrevID"
                => $this->pekerjaan->get_id("prev",$a->ID),
            );
            endif;
            $output = array(
                "HakAksesID"=> $this->session->HakAksesID,
                "Data"
                => $Data,
                "Status"
                => $Status,
                "Message"
                => $Message,
            );
            $this->main->echoJson($output);
        }
    }

```

```

    public function
    save($save_method = "")
    {
        $Status = false;
        $Message = "";
        $this->validation->Pekerjaan($save_method);
        $ID = $this->input->post("ID");
        $Name = $this->input->post("Name");
        $Bobot = $this->input->post("Bobot");
        $Data = array(
            "Name"
            => $Name,
            "Bobot"
            => $this->main->formatDecimal($Bobot),
        );
        if($save_method ==
        "add"):
            $Data["Active"]
            = 1;
            $ID
            = $this->pekerjaan->save($Data);
            $Status
            = true;
            $Message
            = "Data berhasil
            disimpan";
        elseif($save_method ==
        "update"):
            $where
            = array("ID"=>$ID);
            $Update
            = $this->pekerjaan->update($where,$Data);
            $Status
            = true;

```

```

                $Message
            = "Data berhasil disimpan";
            endif;
            $a          = $this-
>pekerjaan->get_by_id($ID);
            $ID        = $a->ID;
            $output = array(
                "HakAkses" =>
$this->session->HakAkses,
                "ID"      => $ID,
                "NextID"
            => $this->pekerjaan-
>get_id("next",$ID),
                "PrevID"
            => $this->pekerjaan-
>get_id("prev",$ID),
                "Status"
            => $Status,
                "Message"
            => $Message
            );
            echo $this->main-
>echoJson($output);
        }
        public function
delete($id="", $status="")
        {
            $Action = $this->input-
>post("Action");
            $active = 0;
            if($Action):
                $ID = $this-
>input->post("ID");
                if($Action ==
"undelete"):
                    $active =
1;
                    endif;
                    $data = array(
                        "Active"
                    => $active,
                );

```

```

                foreach($ID as
$a):
                    $this-
>pekerjaan->update(array('ID' => $a),
                    $data);
                endforeach;
            else:
                if($status ==
"active"):
                    $active =
1;
                endif;
                $data = array(
                    "Active"
                => $active,
                );
                $this-
>pekerjaan->update(array('ID' => $id),
                $data);
            endif;
            echo
json_encode(array("Status" =>
TRUE, "Message" => $this->lang-
>line("success_transaction")));
        }
    }
}

```

M_pekerjaan.php

```

<?php
defined('BASEPATH') or exit('No
direct script access allowed');

class M_pekerjaan extends CI_Model
{
    var $table      =
'UT_Pekerjaan';
    var $column =
array("A.ID", "A.Name", "A.Bobot");
    var $order      = "A.Active
DESC, A.ID ASC";
    public function __construct()
    {
        parent::__construct();
    }
}

```

```

        $this->load-
>database());
        $this->CompanyID =
$this->session->CompanyID;
    }
    private function
_get_datatables_query($page = "")
    {
        $this->db-
>select("A.*");
        $this->db->from($this-
>table . " as A");
        $this->db-
>where("A.Active", 1);
        $i = 0;
        foreach ($this->column
as $item) :
            if
($ _POST['search']['value']) :
                if ($i
=== 0) :
                    $this->db->group_start();

                    $this->db->like($item,
$_POST['search']['value']);
                else :
                    $this->db->or_like($item,
$_POST['search']['value']);
                endif;
                if
(count($this->column) - 1 == $i) :
                    $this->db->group_end();
                endif;
            endif;
            $column[$i] =
$item;
            $i++;
        endforeach;
        if
(isset($_POST['order'])) :

```

```

        $this->db-
>order_by($column[$_POST['order']]['
0']['column']],
        $_POST['order']['0']['dir']);
        elseif (isset($this-
>order)) :
            $order = $this-
>order;
            $this->db-
>order_by($order);
        endif;
    }
    public function
get_datatables($page = "")
    {
        $this-
>_get_datatables_query($page);
        if ($_POST['length'] !=
-1)
            $this->db-
>limit($_POST['length'],
$_POST['start']);
            $query = $this->db-
>get();
            return $query->result();
        }
        public function
count_filtered($page = "")
        {
            $this-
>_get_datatables_query($page);
            $query = $this->db-
>get();
            return $query-
>num_rows();
        }
        public function
count_all($page = "")
        {
            $this->db-
>where("A.Active", 1);
            $this->db->from($this-
>table . " as A");

```

```

        return $this->db-
>count_all_results();
    }
    public function get_by_id($id,
$method = "")
    {
        if ($method == "code")
:
            $this->db-
>select("ID");
        endif;
        $this->db->from($this-
>table);
        $this->db->where('ID',
$id);
        $query = $this->db-
>get();
        return $query->row();
    }
    public function save($data)
    {
        $this->api-
>Log(array("Type" => "CREATE",
"Name" => "MasterBobotPekerjaan",
"DetailData" => $data));
        $this->db-
>set("UserAdd", $this->session-
>Name);
        $this->db-
>set("DateAdd", date("Y-m-d H:i:s"));
        $this->db->insert($this-
>table, $data);
        return $this->db-
>insert_id();
    }
    public function update($where,
$data)
    {
        $this->api-
>Log(array("Type" => $this->api-
>TypeLog($data), "Name" =>

```

```

"MasterBobotPekerjaan", "DetailData"
=> $data));
        $this->db-
>set("UserCh", $this->session-
>Name);
        $this->db-
>set("DateCh", date("Y-m-d H:i:s"));
        $this->db-
>update($this->table, $data, $where);
        return $this->db-
>affected_rows();
    }
    public function
delete_by_id($id)
    {
        $this->db->where('ID',
$id);
        $this->db-
>delete($this->table);
    }
    public function
get_id($method, $id)
    {
        $new_id = 0;
        $this->db-
>select("ID");
        $this->db->from($this-
>table);
        if ($method == "next") :
            $this->db-
>where("ID >", $id);
            $this->db-
>order_by("ID", "ASC");
        else :
            $this->db-
>where("ID <", $id);
            $this->db-
>order_by("ID", "DESC");
        endif;
        $this->db->limit(1);
        $query = $this->db-
>get();

```



```

        public function __construct()
        {
            parent::__construct();
            $this->load-
>helper('url');
            $this->load-
>model("M_testing",'testing');
            $this->main-
>cek_session();
            $this->CompanyID =
$this->session->CompanyID;
        }
        public function index()
        {
            $MenuID
                = $this->main-
>GetMenuID();
            $read
                = $this->main-
>read($MenuID);
            if($read == 0){
                redirect(); }
            $vendor_tambah
                = $this->main-
>menu_tambah($MenuID);
            if($vendor_tambah >
0):
                $tambah = '<button
type="button" class="btn btn-default
waves-effect waves-light"
onclick="tambah()" ><span
class="btn-label"><i class="fa fa-
plus"></i></span> Tambah
Data</button>';
            else:
                $tambah = "";
            endif;
            $data['title'] =
'Testing';
            $data['tambah'] =
$tambah;
            $data['modal'] =
'backend/testing/modal';

            $data['content'] =
'backend/testing/list';
            $data['modul'] =
'testing';
            $data['MenuID'] =
$MenuID;
            $this->load-
>view('backend/index',$data);
        }
        public function list_data($page
= "")
        {
            $MenuID = $this-
>input->post("MenuID");
            $list = $this->testing-
>get_datatables($page);
            $data = array();
            $no =
$_POST['start'];
            $i = 1;
            $btn_delete = $this-
>lang->line('btn_delete');
            $btn_undelete = $this-
>lang->line('btn_undelete');
            $btn_edit =
$this->lang->line('btn_edit');
            $btn_view =
$this->lang->line('btn_view');
            $view_ubah =
$this->main->menu_ubah($MenuID);
            $view_hapus = $this-
>main->menu_hapus($MenuID);
            foreach ($list as $a) {
                $ID = $a-
>TestingID;
                $view = "";
                $edit = "";
                $delete = "";
                $active = "";
                $view = '<a
href="javascript:void(0)" title="View
Data" onclick="edit(this)" data-

```



```

        $output = array(
            "draw"
            => $_POST['draw'],
            "recordsTotal"
            => $this->testing-
>count_all($page),

            "recordsFiltered" => $this-
>testing->count_filtered($page),
            "data"
            => $data,
        );
        echo
        json_encode($output);
    }
    public function edit($id)
    {
        $Data = array();
        $Status = false;
        $Message = "";
        $a = $this-
>testing->get_by_id($id);
        if($a):
            $Status = true;
            $Data = array(

                "TestingID" => $a-
>TestingID,

                "Name"
                => $a->Name,

                "PekerjaanID"
                => $a->PekerjaanID,

                "TanggunganID"
                => $a->TanggunganID,

                "PenghasilanID" => $a-
>PenghasilanID,

                "Status"
                => $a->Status,

                "NextID" => $this-

```

```

>testing->get_id("next",$a-
>TestingID),

                "PrevID"
                => $this->testing-
>get_id("prev",$a->TestingID),
            );
        endif;
        $output = array(

            "HakAksesID"=> $this-
>session->HakAksesID,
            "Data"
            => $Data,
            "Status"
            => $Status,
            "Message"
            => $Message,
        );
        $this->main-
>echoJson($output);
    }
    public function
    save($save_method = "")
    {
        $Status = false;
        $Message = "";
        $this->validation-
>Testing($save_method);
        $TestingID = $this-
>input->post("TestingID");
        $Name
        = $this->input->post("Name");
        $PekerjaanID = $this-
>input->post("PekerjaanID");
        $TanggunganID =
        $this->input->post("TanggunganID");
        $PenghasilanID =
        $this->input->post("PenghasilanID");
        $StatusInp =
        $this->input->post("Status");

        $param = array(

```

```

        $this->testing-
>getBobot("pekerjaan",
$PekerjaanID),
        $this->testing-
>getBobot("tanggung",
$TanggungID),
        $this->testing-
>getBobot("penghasilan",
$PenghasilanID)
    );
    $predict = $this-
>predict($param);
    // print_r($predict);exit;
    $Data = array(
        "Name"
=> $Name,
        "PekerjaanID"
=> $PekerjaanID,
        "TanggungID" =>
$TanggungID,
        "PenghasilanID"
=> $PenghasilanID,
        // "Status"
=> $StatusInp,
        "Predict" =>
$predict == 'Tidak Mampu' ? 1 : 2,
    );
    if($save_method ==
"add"):
        $Data["Active"]
= 1;
        $TestingID
= $this->testing->save($Data);
        $Status
= true;
        $Message
= "Data berhasil
disimpan";
    elseif($save_method ==
"update"):

```

```

        $where
=
array("TestingID"=>$TestingID);
        $Update
= $this->testing-
>update($where,$Data);
        $Status
= true;
        $Message
= "Data berhasil disimpan";
    endif;
    $a = $this-
>testing->get_by_id($TestingID);
    $ID = $a-
>TestingID;
    $output = array(
        "HakAkses" =>
$this->session->HakAkses,
        "TestingID"
=> $TestingID,
        "NextID"
=> $this->testing-
>get_id("next",$TestingID),
        "PrevID"
=> $this->testing-
>get_id("prev",$TestingID),
        "Status"
=> $Status,
        "Message"
=> $Message
    );
    echo $this->main-
>echoJson($output);
}
public function
delete($id="", $status="")
{
    $Action = $this->input-
>post("Action");
    $active = 0;
    if($Action):
        $ID = $this-
>input->post("ID");

```

```

        if($Action ==
"undelete"):
            $active =
1;
            endif;
            $data = array(
                "Active"
=> $active,
            );
            foreach($ID as
$a):
                $this-
>testing->update(array('TestingID' =>
$a), $data);
            endforeach;
        else:
            if($status ==
"active"):
                $active =
1;
                endif;
                $data = array(
                    "Active"
=> $active,
                );
                $this->testing-
>update(array('TestingID' => $id),
$data);
            endif;
            echo
json_encode(array("Status" =>
TRUE,"Message" => $this->lang-
>line("success_transaction")));
        }
        private function
predict($param){
            $samples = array();
            $labels = array();

            $dataTraining = $this-
>testing->getTraining();

            foreach($dataTraining
as $k => $v){
                $samples[] =
[$v->BobotPekerjaan, $v-
>BobotTanggungan, $v-
>BobotPenghasilan];
                $labels[] = $v-
>Status == '1' ? 'Tidak Mampu' :
'Mampu';
            }
            $classifier = new
KNearestNeighbors(5);
            $classifier-
>train($samples, $labels);

            return $classifier-
>predict($param);
        }
        public function modal_result(){
            $page = "modal_result";
            $testing_id = $this-
>input->post("testing_id");
            $list = $this->testing-
>get_datatables_perhitungan($page,
$testing_id);
            $data = array();
            $no = $this->input-
>post("start");
            $i = 1;
            foreach ($list as $a) {
                $status = $a-
>Status == '1' ? 'Tidak Mampu' :
'Mampu';
                $row = array();
                $row[] = $i++;
                $row[] = $a->Name;
                $row[] = $a-
>PekerjaanName;
                $row[] = $a-
>TanggunganName;
                $row[] = $a-
>PenghasilanName;
            }
        }
    }
}

```

```

        $row[] =
$status;
        $row[] = $a-
>Result;
        $data[] = $row;
    }
    $output = array(
        "page"      => $page,
        "draw"      => $this->input-
>post("draw"),
        "recordsTotal" => $this-
>testing-
>count_all_perhitungan($page),
        "recordsFiltered" => $this-
>testing-
>count_filtered_perhitungan($page,$te
string_id),
        "data"      => $data,
    );
    echo json_encode($output);
}
}

```

M_testing.php

```

<?php
defined('BASEPATH') or exit('No
direct script access allowed');

class M_testing extends CI_Model
{
    var $table      = 'testing';
    // var $column =
array("A.TestingID","A.Name",
"ut_pekerjaan.Name","ut_tanggung.
Name","ut_penghasilan.Name","A.Sta
tus","A.Predict");
    var $column =
array("A.TestingID","A.Name",
"ut_pekerjaan.Name","ut_tanggung.
Name","ut_penghasilan.Name","A.Pre
dict");
    var $order      = "A.Active
DESC, A.TestingID ASC";

```

```

public function __construct()
{
    parent::__construct();
    $this->load-
>database();
    $this->CompanyID =
$this->session->CompanyID;
}
private function
_get_datatables_query($page = "")
{
    $this->db->select("A.*,
        ut_penghasilan.name as
PenghasilanName,
        ut_tanggung.Name as
TanggungName,
        ut_pekerjaan.Name as
PekerjaanName
");
    $this->db->from($this-
>table . " as A");
    $this->db-
>join("ut_penghasilan","ut_penghasila
n.ID = A.PenghasilanID","LEFT");
    $this->db-
>join("ut_tanggung","ut_tanggung.
n.ID = A.TanggungID","LEFT");
    $this->db-
>join("ut_pekerjaan","ut_pekerjaan.ID
= A.PekerjaanID","LEFT");
    $this->db-
>where("A.Active", 1);
    $i = 0;
    foreach ($this->column
as $item) :
        if
($ _POST['search']['value']) :
            if ($i
=== 0) :

```

```

        $this->db->group_start();

        $this->db->like($item,
$_POST['search']['value']);
        else :

        $this->db->or_like($item,
$_POST['search']['value']);
        endif;
        if
(count($this->column) - 1 == $i) :

        $this->db->group_end();
        endif;
        endif;
        $column[$i] =
$item;
        $i++;
    endforeach;
    if
(isset($_POST['order'])) :
        $this->db-
>order_by($column[$_POST['order']]['
0']['column'],
$_POST['order']['0']['dir']);
        elseif (isset($this-
>order)) :
            $order = $this-
>order;
            $this->db-
>order_by($order);
        endif;
    }
    public function
get_datatables($page = "")
    {
        $this-
>_get_datatables_query($page);
        if ($_POST['length'] !=
-1)
            $this->db-
>limit($_POST['length'],
$_POST['start']);
        $query = $this->db-
>get();
        return $query->result();
    }
    public function
count_filtered($page = "")
    {
        $this-
>_get_datatables_query($page);
        $query = $this->db-
>get();
        return $query-
>num_rows();
    }
    public function
count_all($page = "")
    {
        $this->db-
>where("A.Active", 1);
        $this->db->from($this-
>table . " as A");
        return $this->db-
>count_all_results();
    }
    public function get_by_id($id,
$method = "")
    {
        if ($method == "code")
:
            $this->db-
>select("TestingID");
        endif;
        $this->db->from($this-
>table);
        $this->db-
>where('TestingID', $id);
        $query = $this->db-
>get();
        return $query->row();
    }
}

```

```

        public function save($data)
        {
            $this->api-
>Log(array("Type" => "CREATE",
>Name" => "Testing", "DetailData" =>
$data));

            $this->db-
>set("UserAdd", $this->session-
>Name);

            $this->db-
>set("DateAdd", date("Y-m-d H:i:s"));
            $this->db->insert($this-
>table, $data);
            return $this->db-
>insert_id();
        }
        public function update($where,
$data)
        {
            $this->api-
>Log(array("Type" => $this->api-
>TypeLog($data), "Name" =>
"Testing", "DetailData" => $data));

            $this->db-
>set("UserCh", $this->session-
>Name);

            $this->db-
>set("DateCh", date("Y-m-d H:i:s"));
            $this->db-
>update($this->table, $data, $where);
            return $this->db-
>affected_rows();
        }
        public function
delete_by_id($id)
        {
            $this->db-
>where("TestingID", $id);
            $this->db-
>delete($this->table);
        }

```

```

        public function
get_id($method, $id)
        {
            $new_id = 0;
            $this->db-
>select("TestingID");
            $this->db->from($this-
>table);

            if ($method == "next") :
                $this->db-
>where("TestingID >", $id);
                $this->db-
>order_by("TestingID", "ASC");
            else :
                $this->db-
>where("TestingID <", $id);
                $this->db-
>order_by("TestingID", "DESC");
            endif;
            $this->db->limit(1);
            $query = $this->db-
>get();

            $a = $query->row();
            if ($a) :
                $new_id = $a-
>TestingID;
            endif;
            return $new_id;
        }
        public function
getBobot($stable, $id){
            $this->db-
>select("A.Bobot");
            $this->db-
>from("ut_{$stable} as A");
            $this->db->where("ID",
$id);

            $query = $this->db-
>get();

            $a = $query->row();
            if ($a) :
                $bobot = $a-
>Bobot;

```

```

        endif;
        return $bobot ?? 0;
    }
    public function getTraining(){
        $this->db->select("

        ut_penghasilan.Bobot as
        BobotPenghasilan,

        ut_tanggungan.Bobot as
        BobotTanggungan,

        ut_pekerjaan.Bobot as
        BobotPekerjaan,

        A.Status

        ");
        $this->db-
>from("training as A");
        $this->db-
>join("ut_penghasilan","ut_penghasila
n.ID = A.PenghasilanID","LEFT");
        $this->db-
>join("ut_tanggungan","ut_tanggunga
n.ID = A.TanggunganID","LEFT");
        $this->db-
>join("ut_pekerjaan","ut_pekerjaan.ID
= A.PekerjaanID","LEFT");
        $query = $this->db-
>get();
        return $query->result();
    }

    // PERHITUNGAN
    private function
_get_datatables_query_perhitungan($p
age = "", $testing_id)
    {
        $result = "(
                SELECT
SQRT(

```

```

                POW((d.Bobot -
ut_pekerjaan.Bobot), 2)
                +
                POW((c.Bobot -
ut_tanggungan.Bobot), 2)
                +
                POW((b.Bobot -
ut_penghasilan.Bobot), 2)
                ) as result
                FROM testing
as X
                LEFT JOIN
ut_penghasilan b ON b.ID =
x.PenghasilanID
                LEFT JOIN
ut_tanggungan c ON c.ID =
x.TanggunganID
                LEFT JOIN
ut_pekerjaan d ON d.ID =
x.PekerjaanID
                WHERE
x.Active = 1
                AND
x.TestingID = $testing_id
                )";
        $column =
array("A.TrainingID","A.Name",
"ut_pekerjaan.Name","ut_tanggungan.
Name","ut_penghasilan.Name","A.Sta
tus","Result");
        $order = "A.Active
DESC, Result ASC";
        $this->db->select("A.*,

        ut_penghasilan.name as
        PenghasilanName,

        ut_tanggungan.Name as
        TanggunganName,

```

```

        ut_pekerjaan.Name as
PekerjaanName,
        $result as Result
    );
    $this->db-
>from("training as A");
    $this->db-
>join("ut_penghasilan","ut_penghasila
n.ID = A.PenghasilanID","LEFT");
    $this->db-
>join("ut_tanggungans","ut_tanggungana
n.ID = A.TanggungansID","LEFT");
    $this->db-
>join("ut_pekerjaan","ut_pekerjaan.ID
= A.PekerjaanID","LEFT");
    $this->db-
>where("A.Active", 1);
    $i = 0;
    foreach ($column as
$item) :
        if
($ _POST['search']['value']) :
            if($item
== "Result") $item = $result;
            if ($i
=== 0) :
                $this->db->group_start();

                $this->db->like($item,
$_POST['search']['value']);
            else :
                $this->db->or_like($item,
$_POST['search']['value']);
            endif;
            if
(count($column) - 1 == $i) :
                $this->db->group_end();
            endif;
        endif;
    endif;

```

```

        $column[$i] =
$item;
        $i++;
    endforeach;
    if
(isset($_POST['order'])) {
        $this->db-
>order_by($column[$_POST['order']]['
0']['column'],
$_POST['order']['0']['dir']);
    } elseif (isset($order)) {
        $this->db-
>order_by($order);
    }
    }
    public function
get_datatables_perhitungan($page =
"", $testing_id)
    {
        $this-
>_get_datatables_query_perhitungan($
page, $testing_id);
        if ($_POST['length'] !=
-1)
            $this->db-
>limit($_POST['length'],
$_POST['start']);
        $query = $this->db-
>get();
        return $query->result();
    }
    public function
count_filtered_perhitungan($page =
"", $testing_id)
    {
        $this-
>_get_datatables_query_perhitungan($
page, $testing_id);
        $query = $this->db-
>get();
        return $query-
>num_rows();
    }

```



```

<h3 class="form-title m-b-20"></h3>
<div class="approval-status-msg sembunyi">Informasi</div>
<form id="form" method="post" class="form-design-1" enctype="multipart/form-data">
  <div class="row">
    <div class="col-sm-6">
      <div class="panel panel-color panel-custom">
        <div class="panel-heading clearfix">
          <span class="btn-group div-btn-prev">
            <a href="javascript:;" id="BtnPrev" onclick="div_form('prev_data')" class="btn btn-white btn-outline btn-xs"><i class="ti-angle-left"></i></a>
          </span>
          <span class="panel-title">Data <?= $title ?></span>
          <span class="btn-group pull-right div-btn-next">
            <a href="javascript:;" id="BtnNext" onclick="div_form('next_data')" class="btn btn-white btn-outline btn-xs"><i class="ti-angle-right"></i></a>
          </span>
        </div>
        <div class="panel-body table-responsive">
          <input type="hidden" name="TestingID">
          <div class="row">
            <div class="col-sm-12">
              <div class="row">
                <div class="col-sm-12">
                  <div class="form-group">

```

```

          <label class="control-label">Nama</label>
          <input name="Name" id="Name" type="text" class="form-control" maxlength="100">
          <span class="help-block"></span>
        </div>
      </div>
    </div>
  </div>
  <div class="col-sm-12">
    <div class="form-group">
      <label class="control-label">Pekerjaan</label>
      <select name="PekerjaanID" class="form-control pekerjaan_select select2 fist-input" data-active="1" tabindex="-1">
        <option value="none">Pilih Pekerjaan</option>
      </select>
      <span class="help-block PekerjaanIDAlert"></span>
    </div>
  </div>
  <div class="col-sm-12">
    <div class="form-group">
      <label class="control-label">Tanggungans</label>
      <select name="TanggungansID" class="form-control tanggungan_select select2 fist-input" data-active="1" tabindex="-1">

```

```

        <option
value="none">Pilih
Tanggung</option>
    </select>
    <span
class="help-block
TanggungIDAlert"></span>
    </div>
</div>
<div class="col-sm-
12">
    <div class="form-
group">
        <label
class="control-
label">Penghasilan</label>
        <select
name="PenghasilanID" class="form-
control penghasilan_select select2 fist-
input" data-active="1" tabindex="-1">
            <option
value="none">Pilih
Penghasilan</option>
        </select>
        <span
class="help-block
PenghasilanIDAlert"></span>
        </div>
    </div>
<div class="col-sm-
12 hidden">
    <div class="form-
group">
        <label
class="control-label">Status</label>
        <select
name="Status" class="form-control
select2 fist-input" data-active="1"
tabindex="-1">
            <option
value="none">Pilih Status</option>
            <option
value="1">Tidak Mampu</option>

```

```

        <option
value="2">Mampu</option>
    </select>
    <span
class="help-block
StatusAlert"></span>
    </div>
</div>
</div>
</div>
</div>
</div>
<div class="panel-footer">
    <?= $this->main-
>form_btn(['position' => 'left']) ?>
</div>
</div>
</div>
</div>
</form>
</div>

```

Training.php

```

<?php
defined('BASEPATH') OR exit('No
direct script access allowed');

class Training extends CI_Controller {
    public function __construct()
    {
        parent::__construct();
        $this->load-
>helper('url');
        $this->load-
>model("M_training",'training');
        $this->main-
>cek_session();
        $this->CompanyID =
$this->session->CompanyID;
    }
    public function index()
    {

```

```

        $MenuID
        = $this->main-
>GetMenuID();
        $read
        = $this->main-
>read($MenuID);
        if($read == 0){
redirect(); }
        $vendor_tambah
        = $this->main-
>menu_tambah($MenuID);
        if($vendor_tambah >
0):
            $tambah = '<button
type="button" class="btn btn-default
waves-effect waves-light"
onclick="tambah()" ><span
class="btn-label"><i class="fa fa-
plus"></i></span> Tambah
Data</button>';
            else:
                $tambah = "";
            endif;
            $data['title'] =
'Training';
            $data['tambah'] =
$tambah;
            $data['modal'] =
'backend/training/modal';
            $data['content'] =
'backend/training/list';
            $data['modul'] =
'training';
            $data['MenuID'] =
$MenuID;
            $this->load-
>view('backend/index',$data);
        }
        public function list_data($page
="")
        {
            $MenuID = $this-
>input->post("MenuID");
            $list = $this-
>training->get_datatables($page);
            $data = array();
            $no =
$_POST['start'];
            $i = 1;
            $btn_delete = $this-
>lang->line('btn_delete');
            $btn_undelete = $this-
>lang->line('btn_undelete');
            $btn_edit =
$this->lang->line('btn_edit');
            $btn_view =
$this->lang->line('btn_view');
            $view_ubah =
$this->main->menu_ubah($MenuID);
            $view_hapus = $this-
>main->menu_hapus($MenuID);
            foreach ($list as $a) {
                $ID = $a-
>TrainingID;
                $view = "";
                $edit = "";
                $delete = "";
                $active = "";
                $view = '<a
href="javascript:void(0)" title="View
Data" onclick="edit(this)" data-
id="'.$a->TrainingID.'" data-
method="view">'. $btn_view. '</a>';
                if($view_ubah >
0):
                    $edit = '<a
href="javascript:void(0)" title="Edit"
onclick="edit(this)" data-id="'.$a-
>TrainingID.'" data-
method="edit">'. $btn_edit. '</a>';
                    endif;
                    if($view_hapus
> 0):
                        if($a->Active == 1):
                            $delete = '<a
href="javascript:void(0)" title="Delete

```

```

Data" onclick="hapus('.'.$a-
>TrainingID.'.'.$a-')>'. $btn_delete.'</a>'
;
    else:
        $active = ' <i
class="fa fa-trash pull-right"></i>';
        $delete = ' <a
href="javascript:void(0)"
title="Undelete Data"
onclick="active('.'.$a-
>TrainingID.'.'.$a-')>'. $btn_undelete.'</
a>';
    endif;
    endif;
    $no++;

    $action = ' <div
class="dropdown">';
    $action .=
' <span href="javascript:;"
class="dropdown-toggle"
type="button" data-
toggle="dropdown">'. $active.$a-
>Name.'</span>';
    $action .= ' <ul
class="dropdown-menu">';
    $action .=
" <li>$view</li>";
    $action .=
" <li>$edit</li>";
    $action .=
" <li>$delete</li>";
' </ul>';
    $action .=
' </div>';

    $th_no = " <span
class='th-no'>". $i++. " </span>";

    $status = $a-
>Status == '1' ? 'Tidak Mampu' :
'Mampu';

```

```

$row =
array();
$row[] = $no;
$row[] =
$row[] = $a-
>PekerjaanName;
$row[] = $a-
>TanggungName;
$row[] = $a-
>PenghasilanName;
$row[] =
$status;
    $data[] = $row;
}
$output = array(
    "draw"
=> $_POST['draw'],
    "recordsTotal"
=> $this->training-
>count_all($page),
    "recordsFiltered" => $this-
>training->count_filtered($page),
    "data"
=> $data,
);
echo
json_encode($output);
}
public function edit($id)
{
    $Data = array();
    $Status = false;
    $Message = "";
    $a = $this-
>training->get_by_id($id);
    if($a):
        $Status = true;
        $Data = array(
            "TrainingID" => $a-
>TrainingID,

```

```

        "Name"
        => $a->Name,

        "PekerjaanID"
        => $a->PekerjaanID,

        "TanggungansID"
        => $a->TanggungansID,

        "PenghasilanID" => $a-
>PenghasilanID,

        "Status"
        => $a->Status,

        "NextID" => $this-
>training->get_id("next",$a-
>TrainingID),

        "PrevID"
        => $this->training-
>get_id("prev",$a->TrainingID),
    );
    endif;
    $output = array(

        "HakAksesID"=> $this-
>session->HakAksesID,
        "Data"
        => $Data,
        "Status"
        => $$status,
        "Message"
        => $Message,
    );
    $this->main-
>echoJson($output);
}
public function
save($save_method = "")
{
    $Status = false;
    $Message = "";
    $this->validation-
>Training($save_method);
        $TrainingID = $this-
>input->post("TrainingID");
        $Name
        = $this->input->post("Name");
        $PekerjaanID = $this-
>input->post("PekerjaanID");
        $TanggungansID =
$this->input->post("TanggungansID");
        $PenghasilanID =
$this->input->post("PenghasilanID");
        $StatusInp =
$this->input->post("Status");

        $Data = array(
            "Name"
            => $Name,
            "PekerjaanID"
            => $PekerjaanID,

            "TanggungansID" =>
$TanggungansID,
            "PenghasilanID"
            => $PenghasilanID,
            "Status"
            => $StatusInp,
        );
        if($save_method ==
"add"):
            $Data["Active"]
            = 1;
            $TrainingID
            = $this->training-
>save($Data);
            $Status
            = true;
            $Message
            = "Data berhasil
disimpan";
        elseif($save_method ==
"update"):
            $where
            =
array("TrainingID"=>$TrainingID);

```

```

        $Update
    = $this->training-
>update($where,$Data);
        $Status
    = true;
        $Message
    = "Data berhasil disimpan";
    endif;
    $a = $this-
>training->get_by_id($TrainingID);
    $ID = $a-
>TrainingID;
    $output = array(
        "HakAkses" =>
$this->session->HakAkses,
        "TrainingID"
=> $TrainingID,
        "NextID"
=> $this->training-
>get_id("next",$TrainingID),
        "PrevID"
=> $this->training-
>get_id("prev",$TrainingID),
        "Status"
=> $Status,
        "Message"
=> $Message
    );
    echo $this->main-
>echoJson($output);
}
public function
delete($id="", $status="")
{
    $Action = $this->input-
>post("Action");
    $active = 0;
    if($Action):
        $ID = $this-
>input->post("ID");
        if($Action ==
"undelete"):

```

```

        $active =
1;
    endif;
    $data = array(
        "Active"
=> $active,
    );
    foreach($ID as
$a):
        $this-
>training->update(array('TrainingID'
=> $a), $data);
    endforeach;
    else:
    if($status ==
"active"):
        $active =
1;
    endif;
    $data = array(
        "Active"
=> $active,
    );
    $this->training-
>update(array('TrainingID' => $id),
$data);
    endif;
    echo
json_encode(array("Status" =>
TRUE,"Message" => $this->lang-
>line("success_transaction")));
}
}

```

M_training.php

```

<?php
defined('BASEPATH') or exit('No
direct script access allowed');

class M_training extends CI_Model
{
    var $table = 'training';

```

```

        var $column =
array("A.TrainingID","A.Name",
"ut_pekerjaan.Name","ut_tanggung.
Name","ut_penghasilan.Name","A.Sta
tus");
        var $order    = "A.Active
DESC, A.TrainingID ASC";
        public function __construct()
        {
                parent::__construct();
                $this->load-
>database();
                $this->CompanyID =
$this->session->CompanyID;
        }
        private function
_get_datatables_query($page = "")
        {
                $this->db->select("A.*,

                ut_penghasilan.name as
PenghasilanName,

                ut_tanggung.Name as
TanggungName,

                ut_pekerjaan.Name as
PekerjaanName
                ");
                $this->db->from($this-
>table . " as A");
                $this->db-
>join("ut_penghasilan","ut_penghasila
n.ID = A.PenghasilanID","LEFT");
                $this->db-
>join("ut_tanggung","ut_tanggung
n.ID = A.TanggungID","LEFT");
                $this->db-
>join("ut_pekerjaan","ut_pekerjaan.ID
= A.PekerjaanID","LEFT");
                $this->db-
>where("A.Active", 1);
                $i = 0;

```

```

        foreach ($this->column
as $item) :
                if
($ _POST['search']['value']) :
                        if ($i
=== 0) :
                                $this->db->group_start();

                                $this->db->like($item,
$_POST['search']['value']);
                        else :
                                $this->db->or_like($item,
$_POST['search']['value']);
                                endif;
                                if
(count($this->column) - 1 == $i) :
                                        $this->db->group_end();
                                endif;
                                endif;
                                $column[$i] =
$item;
                                $i++;
                        endforeach;
                if
(isset($_POST['order'])) :
                        $this->db-
>order_by($column[$_POST['order']]['column'],
$_POST['order']['0']['dir']);
                        elseif (isset($this-
>order)) :
                                $order = $this-
>order;
                                $this->db-
>order_by($order);
                                endif;
                }
        public function
_get_datatables($page = "")
        {

```

```

        $this->
>_get_datatables_query($page);
        if ($_POST['length'] !=
-1)
            $this->db-
>limit($_POST['length'],
$_POST['start']);
        $query = $this->db-
>get();
        return $query->result();
    }
    public function
count_filtered($page = "")
    {
        $this->
>_get_datatables_query($page);
        $query = $this->db-
>get();
        return $query->
>num_rows();
    }
    public function
count_all($page = "")
    {
        $this->db-
>where("A.Active", 1);
        $this->db->from($this->
>table . " as A");
        return $this->db-
>count_all_results();
    }
    public function get_by_id($id,
$method = "")
    {
        if ($method == "code")
:
            $this->db-
>select("TrainingID");
            endif;
            $this->db->from($this->
>table);
            $this->db-
>where("TrainingID", $id);
        $query = $this->db-
>get();
        return $query->row();
    }
    public function save($data)
    {
        $this->api-
>Log(array("Type" => "CREATE",
"Name" => "Training", "DetailData"
=> $data));
        $this->db-
>set("UserAdd", $this->session-
>Name);
        $this->db-
>set("DateAdd", date("Y-m-d H:i:s"));
        $this->db->insert($this->
>table, $data);
        return $this->db-
>insert_id();
    }
    public function update($where,
$data)
    {
        $this->api-
>Log(array("Type" => $this->api-
>TypeLog($data), "Name" =>
"Training", "DetailData" => $data));
        $this->db-
>set("UserCh", $this->session-
>Name);
        $this->db-
>set("DateCh", date("Y-m-d H:i:s"));
        $this->db-
>update($this->table, $data, $where);
        return $this->db-
>affected_rows();
    }
    public function
delete_by_id($id)
    {

```

```

        $this->db-
>where("TrainingID", $id);
        $this->db-
>delete($this->table);
    }
    public function
get_id($method, $id)
    {
        $new_id = 0;
        $this->db-
>select("TrainingID");
        $this->db->from($this-
>table);
        if ($method == "next") :
            $this->db-
>where("TrainingID >", $id);
            $this->db-
>order_by("TrainingID", "ASC");
        else :
            $this->db-
>where("TrainingID <", $id);
            $this->db-
>order_by("TrainingID", "DESC");
        endif;
        $this->db->limit(1);
        $query = $this->db-
>get();
        $a = $query->row();
        if ($a) :
            $new_id = $a-
>TrainingID;
        endif;
        return $new_id;
    }
}

```