



- Your Technology Solutions Partner -

Questions related to the manual, Call eNova at 9345636682 or 9500777781

Ref URL: <http://nova-portal.com/Prototype>

Create Button should be: New

Sample Code:

```
<div class="table-toolbar" style="float:right">
  <div class="btn-group">
    <a class="btn btn-info" id="createNewVersion" href="javascript:;"> <i class="fa fa-
plus"></i>New</a>
  </div>
</div>
```

Data Table:

1. STATIC TABLE CODE:

```
<form method="post" id="form" >
  <button type="button" id="pdf" class="btn_empty" style="margin-left: 3px;border: 0;opacity:
0;"></button>
  <input type="hidden" name="sear_val" id="sear_val" value="">
  <table cellpadding="0" cellspacing="0" border="0" id="dTable1" class="display table table-
bordered" width="100%">
    <thead>
      <tr>
        <th>Id</th>
        <th width="30%">Version</th>
        <th width="30%">Regulation</th>
        <th width="20%">Batch</th>
        <th width="20%">Action</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td colspan="4" class="dataTables_empty">No Data available</td>
      </tr>
    </tbody>
  </table>
</form>
```

2. SCRIPT:

```
$('#dTable1').dataTable({
    "sPaginationType": "full_numbers",
    "bJQueryUI": false,
    "bAutoWidth": false,
    "bLengthChange": true,
    'bProcessing' : false,
    'bServerSide' : true,
    'iDisplayLength' : 10,
    'iDisplayStart' : 0,
    'sAjaxSource' : '<?php echo site_url("Prototype/listing"); ?>',
    'aoColumns' : [
        {sName: 'V.version_id',"bVisible":false,
        "render": function ( data, type, row ) {
            return '<input type="checkbox" class="check-row" name="user[]" value='+data+'
    />';
        }
    },
    {sName: 'V.version'},
    {sName: 'regulation'},
    {sName: 'batch'},
    { 'sName': 'action',"bSortable":false},
    ],
    "aaSorting": [[ 1, "DESC" ]],
    'fnServerData': function(sSource, aoData, fnCallback) {
    $.ajax ({
        'dataType': 'json',
        'type' : 'POST',
        'url' : sSource,
        'data' : aoData,
        'success': fnCallback
    });
    }
});
```



```
        required:'This field is mandatory',
    },
    'batch_id':{
        required:'This field is mandatory'
    },
    'regulation_id':{
        required:'This field is mandatory'
    }
    },
    submitHandler: function(form) {
        var regulation_id = $("#regulation_id").val();
        var batch_id = $("#batch_id").val();
        var version = $("#version").val();
        var version_id = $("#version_id").val();

        $.ajax({
            type: "POST",
            url : '<?php echo site_url("Prototype/checkVersion"); ?>',
            data:
            {'version':version,"regulation_id":regulation_id,"batch_id":batch_id,"version_id":version_id},
            success: function(data) {
                if(data == 0){
                    $.ajax({
                        type: "POST",
                        url : '<?php echo site_url("Prototype/createVersion"); ?>',
                        data:
                        {'version':version,"regulation_id":regulation_id,"batch_id":batch_id,"version_id":version_id},
                        success: function(data) {
                            if(data != 0){

                                var dt = new Date();
                                var time = dt.getHours() + ":" + dt.getMinutes() + ":" + dt.getSeconds();

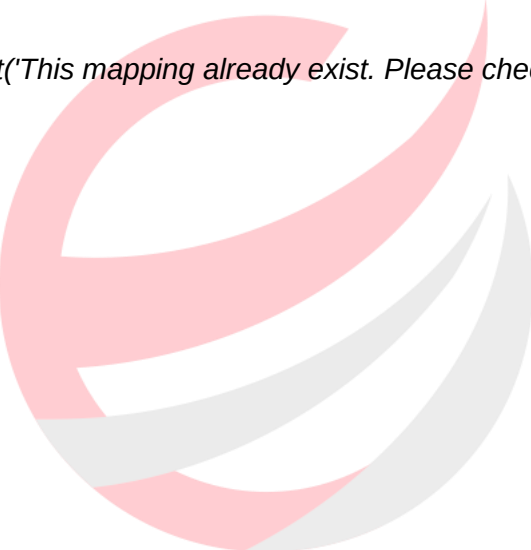
                                $("#suc_alert_message").text('Successfully Saved! - '+time);
                                $("#suc_alert_message").removeClass('hide');

                                setTimeout(function() {
                                    $("#dTable1").dataTable().fnDraw();
                                    $("#suc_alert_message").addClass('hide');
                                    $("#close_icon").trigger('click');
                                },3000);
                                return false;
                            }else{
```

```
$("#suc_alert_message").text('Successfully Saved! - '+time);
$("#suc_alert_message").removeClass('hide');

setTimeout(function() {
  $("#dTable1").dataTable().fnDraw();
  $("#suc_alert_message").addClass('hide');
  $("#close_icon").trigger('click');
},3000);

bootbox.alert('Error occured while crate version details please try again.');
```



```
return false;
}
}
});
return false;
},
});
```

5. CHECK DUPLICATE ENTRY:

Controller Function:

```
public function checkVersion(){
    $details = $_POST;
    echo $this->Prototype_model->checkVersionexistornot($details);
}
```

Model Function:

```
public function checkVersionexistornot($details){
    $version_id = $details['version_id'];
    unset($details['version_id']);
    $this->db->where($details);
    $this->db->where('version_id !=', $version_id);
    return $this->db->get('copo_versions')->num_rows();
}
```

Insert Process:

Controller Function:

```
public function createVersion(){
    $details = $_POST;
    echo $this->Prototype_model->createVersion($details);
}
```

Model Function:

```
public function createVersion($data){
    $version_id = $data['version_id'];
    unset($data['version_id']);

    if($version_id){
        $data['created_by'] = $_SESSION['uid'];
        $data['date_created'] = date('Y-m-d H:i:s');
        $this->db->where('version_id', $version_id);
        $this->db->update('copo_versions', $data);
        return $this->db->affected_rows();
    }else{
        $data['created_by'] = $_SESSION['uid'];
        $data['date_created'] = date('Y-m-d H:i:s');
        $this->db->insert('copo_versions', $data);
        return $this->db->insert_id();
    }
}
```

```
}  
}
```

EDIT:

Retrieve data by ID edit purpose

POP UP Process:

Controller Function:

```
public function getDetails(){  
    $details = $_POST;  
    $result = $this->Prototype_model->getDetails($details['id']);  
    echo json_encode($result);  
}
```

Model Function:

```
public function getDetails($id){  
    $this->db->where('version_id',$id);  
    return $this->db->get('copo_versions')->row_array();  
}
```

New page Process:

Controller Function:

```
public function edit_regulation($id){  
  
    $this->form_validation->set_rules('regulation[name]', 'name', 'required');  
    $this->form_validation->set_rules('regulation[code]', 'code', 'required');  
    //breadcrumb process  
    $breadcrumb="";  
    $breadcrumb.='<li                                     class="breadcrumb-item"><a  
href="'.site_url('Dashboard').'">Home</a></li>';  
    $breadcrumb.='<li class="breadcrumb-item"><span>Academics</span></li>';  
    $breadcrumb.='<li class="breadcrumb-item"><span>Settings</span></li>';  
    $breadcrumb.='<li                                     class="breadcrumb-item"><a  
href="'.site_url('Regulations/index').'">Regulations</a></li>';  
    $breadcrumb.='<li class="breadcrumb-item"><span>Edit Regulation</span></li>';  
    $this->data['breadcrumb']= $breadcrumb;  
    //menu active process  
    $this->data['main_menu']='Academics';  
    $this->data['submenu'] = 'Academic settings';  
  
    if($this->form_validation->run()==FALSE){
```



```

        $this->data['title']='Edit Regulation';

        $this->data['regulation']=$this->Regulations_model->getRegulation($id);
        $breadcrumb_text      = '<li> <a href="'. site_url('Regulations')." " class="tip"
title="Back to Regulations">Regulations</a></li>';
        $breadcrumb_text     .= '<span class="divider"><span class="icon16 icomoon-
icon-arrow-right-2"></span></span>';
        $breadcrumb_text     .= '<li class="active">Edit</li>';
        $this->data['breadcrumb_text']=$breadcrumb_text;
        $this->data['action']=site_url('Regulations/edit_regulation/'.$id);
        $this->data['save_button']='Update';

        $this->data['view_content']='Regulations/regulation_form';
        $this->load->view('template',$this->data);

    }
    else{
        $data=$this->input->post('regulation');
        $data['updated_date']=date('Y-m-d H:i:s');
        $data['updated_by']=$this->session->userdata('user_id');
        $this->Regulations_model->editRegulation($data,$id);
        $this->session->set_flashdata('message_success', 'Successfully Saved! -'
.date('d-m-Y h:i a'));
        redirect('Regulations');
    }
}

```

DELETE:Script:

```

$(document).on('click','delete',function(event){
    var id = $(this).attr('data-attr');
    bootbox.confirm({
        message: "Are you sure want to delete?",
        buttons: {
            cancel: {
                label: 'No',
                className: 'btn btn-danger'
            },
            confirm: {
                label: 'Yes',
                className: 'btn btn-info'
            }
        }
    });

```

```

    },
  },
  callback: function (result) {

    if(result){
      $.ajax({
        type: "POST",
        url : '<?php echo site_url("Prototype/deleteRecord"); ?>',
        data: {'id':id},
        success: function(data) {
          window.location.reload();
        }
      });
    }
  }
});
});

```

Controller Function:

```

public function deleteRecord(){
  $id = $this->input->post('id');
  $data = array(
    'status'      => 1,
    'updated_by'  => $_SESSION['uid'],
    'date_updated' => date("Y-m-d H:i:s"),
  );

  $result = $this->Prototype_model->deleteRecord($id,$data);

  if($result){
    $this->session->set_flashdata('message_type', 'alert-success');
    $this->session->set_flashdata('message', 'Successfully Deleted - '.date('d-m-Y
h:i a'));
  }else{
    $this->session->set_flashdata('message_type', 'alert-danger');
    $this->session->set_flashdata('message', 'Error Deleting - '.date('d-m-Y h:i a'));
  }
}
}

```

Model Function:

```
$this->db->where('version_id',$id);  
    $this->db->update('prototype',$data);  
    return $this->db->affected_rows();  
}
```

Alert Process:

Controller:

```
if($result){  
    $this->session->set_flashdata('message_type', 'alert-success');  
    $this->session->set_flashdata('message', 'Successfully Deleted - '.date('d-m-Y  
h:i a'));  
}else{  
    $this->session->set_flashdata('message_type', 'alert-danger');  
    $this->session->set_flashdata('message', 'Error Deleting - '.date('d-m-Y h:i a'));  
}
```

POPUP:

```
<button type="button" class="btn btn-info btn-lg hide" id="create" data-toggle="modal" data-  
target="#myModal"></button>  
<!-- Modal -->  
<div id="myModal" class="modal fade" role="dialog">  
    <div class="modal-dialog">  
        <!-- Modal content-->  
        <div id="popup_tab" class="modal-content">  
            <div class="popup_head">  
                <span id="close_icon" class="close" data-dismiss="modal" aria-  
hidden="true">&times;</span>  
                <h4 id="heading"> Create New Version </h4>  
            </div>  
            <div class="act_sec1">  
                <form class="form-horizontal" method="post" action="javascript:;" id="form1"  
enctype="multipart/form-data">  
                    <div class="col-md-12 entexam_blk">  
                        <input type="hidden" name="department_Id" id="depld" value="<?php echo  
$DepId; ?>">
```

```

        <input type="hidden" name="map_Id" id="mapId" value="<?php echo $mapId;
?>">

        <div class="form-group" >
            <label                class="form-label                col-md-5">Regulation<span
style="color:red">*</span></label>
            <div class="col-sm-6">
                <select id="regulation_id" class="form-control" name="regulation_id">
                    <option value="">Please select Regulation</option>
                    <?php foreach ($regulations as $reg) { ?>
                        <option value="<?php echo $reg['regulation_id'];?>"><?php echo $reg['name'];
?></option>
                    <?php } ?>
                </select>
            </div>
        </div>

        <div class="form-group" >
            <label                class="form-label                col-md-5">Batch<span
style="color:red">*</span></label>
            <div class="col-sm-6">
                <select id="batch_id" class="form-control" name="batch_id">
                    <option value="">Please select Batch</option>
                    <?php foreach ($batches as $b) { ?>
                        <option value="<?php echo $b['batch_id'];?>"><?php echo $b['year'];
?></option>
                    <?php } ?>
                </select>
            </div>
        </div>

        <input type="hidden" name="version_id" id="version_id"></ins>
        <div class="form-group" >
            <label                class="form-label                col-md-5">Version<span
style="color:red">*</span></label>
            <div class="col-sm-6">
                <input type="text" id="version" autocomplete="off" placeholder="Ex: Version 1"
class="form-control" name="version">
            </div>
        </div>

        <div class="popup_footer form-group">
            <div class="alret_message_group">
                <span id="suc_alert_message" class=""></span>
                <span id="failure_alert_message" class=""></span>
            </div>

```

```
<div class="ctr_group">
  <button type="submit" id="save" class="btn btn-info save">Save</button>
  <a href="javascript:;" class="btn btn-danger" data-dismiss="modal" aria-
hidden="true"> Cancel</a>
</div>
</div>
</form>
</div>
</div>
</div>
</div>
```



Page Link: 192.168.130.204/demo.novaportal.com/Profile/edit_profile/1074

File Path: D:\xampp\htdocs\novaportal\application\modules\Profile\views

Profile Image and Signature Image Upload Section _ Coding:

```
<div class="profile_imgblk">
    <?php //if($_SESSION['user_role_id']!=15){?>
    <div class="form-group">
        <label class="col-md-3 col-sm-2 control-label" for="Student Image">Profile
Image <span class="star" style="color: red;">*</span>
        <p class="green_alert">Image (.jpg) file only. Aspect Ratio= 1:1. File Size: 10
Kb to 20 Kb </p>
        </label>
        <div class="col-md-3 col-sm-4 mar-top-10 text_me">
            <input type="file" name="userfile" id="userfile" accept="image/jpeg,image/png"
class="form-control pad_none1" />
            <input type="hidden" id="img" name="image" value="<?php echo
isset($db_data['image'])? $db_data['image'].": "; ?>" />
            <span class="upload_status1"></span>
        </div>
        <div class="bg_layer ">
            <div id="uploadPreview" class="profile_logosetionblk">
                <!--  -->
                
            </div>
        </div>
    </div> <?php //}?>

    <div class="form-group userLogo_blk ">
        <label class="col-md-3 col-sm-2 control-label" >Signature
Image<span class="star" style="color: red;">*</span>
        <p class="green_alert">Image (.jpg) file only. Aspect Ratio= 4:1.
File Size: 30 Kb to 50 Kb </p>
        </label>
        <div class=" col-md-5 user_logo_block">
```

```

<input type="file" class="form-control signature"
name="signature" id="signature" />
<span class="upload_status1"></span>
</div>
<!-- <p style="color:#060;font-size: 90%">Image (.jpg) file
only.Aspect Ratio= 4:1. File Size: 30 Kb to 50 Kb </p> -->
<div class="bg_layer ">
<div id="uploadPreview" class="sign_logosetionblk ">
<!--  -->

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

```

Screenshot:

NOVAPORTAL
VERSION 2.0

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS

MASTER SETTINGS ADMINISTRATION ACADEMICS REPORTS

Behaviors | Medium

4 | Classroom Size | High

3 | COMPUTER | Medium

2 | FEES | Critical

1 | test enova | High

View All

PUBLISHER

1. WILLIAM BLAKES

2. EVELYN WAUGH

3. HARPER COLLINS

4. NIALL FERGUSON

5. SAMANTHA POWER

They are Recent Ebook Publishers.

View All

UserName

enova

New Password

Confirm Password

Mobile *

9966115902

Email *

mano@gmail.com

Profile Image *

Image (.jpg) file only, Aspect Ratio= 1:1. File Size: 10 Kb to 20 Kb

Browse... No file selected.

Signature Image *

Image (.jpg) file only, Aspect Ratio= 4:1. File Size: 30 Kb to 50 Kb

Browse... No file selected.

Update Cancel

2021

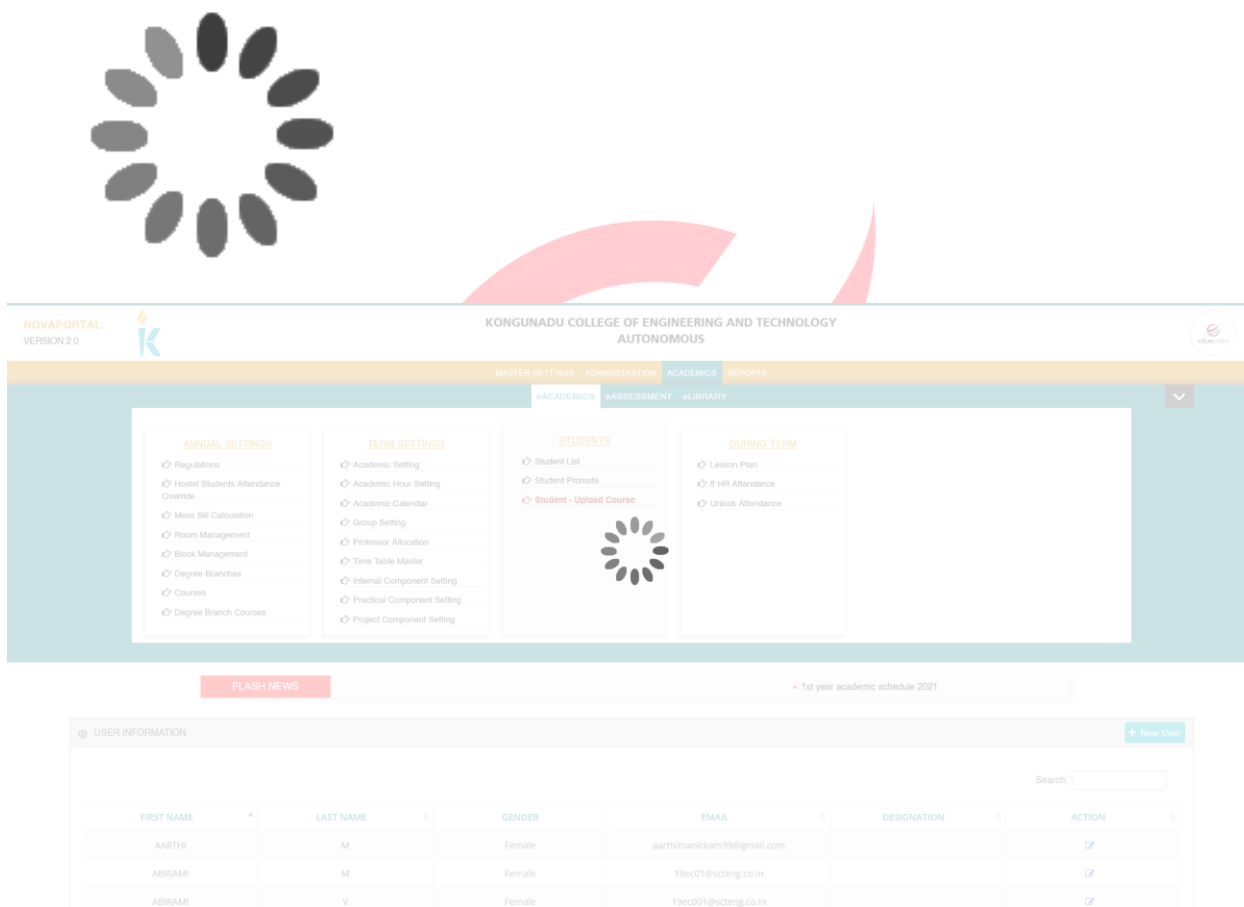
View All

ACADEMIC SETTING

View All

Common Loader

This is the common loader for all nova portals. The color should be grey.



Application Testing stages to be followed at eNova are as follows:

1. **UNIT TEST** - Testing of the module by the individual who programs the module is termed as unit testing. It is typically done by the programmer and all checks related to how it works to specifications, matching coding standards and so on.
2. **SYSTEM TEST** - The system test requires the programmer to make sure the impacted modules and reports related to the module he/she worked on must be checked for appropriate output. It must be done by the programmer. Any standards used related to the inter module communication must be checked also. Once the UNIT and SYSTEM test are done and verified, the process testing then goes to QUALITY testers who are our client advocates.
3. **QUALITY TEST** – The client advocate is the final gate keeper to make sure there is 0 rework and bugs. They must cross check for standards and also check for system level flow and functionality.
4. **PRE PRODUCTION TEST** – After completing 3 stages of testing, the code must be per expectation and error free. It's the programmers responsibility first followed by the client advocate

