

OSTicket -> Admin Panel

Okay, so first we need to create a custom list with the list of statuses that you want.

For this list, I set the extra value to be the color you want the background to be. You don't have to do it this way but it seemed the easiest way to accomplish it to me. Choose how you want to have the color in there (if you want to put the # for colors in the extra field or build into the php code). Looking back, I would put the # sign in the extra field, which opens up the option to use words for colors (black, white, red, etc)

Along with the custom list, be sure to add a field to your ticket detail for the ticket status.

Custom List

Update custom list

Custom lists are used to provide drop-down lists for custom forms

Name:

Plural Name:

Sort Order:

Showing 1 - 17 of 17 list items

Value	Extra — abbreviations and such
Open	FFF
Assigned	EE82EE
Awaiting Approval	FF0
Approved	0C3
Denied	F00
New Ticket	AAA
Awaiting Response	FCC
Follow Up	F60
Image	6AA
Out for Repair	CF3
Parts Waiting	DCDCDC
Ready to Deploy	CFF
Reimage	606
Scheduled	9CF
Project	C60
Testing	FFD700
Training	0CC

Sort	Label	Type	Internal	Required	Variable	Delete
↕	<input type="text" value="Ticket Status"/>	<input type="text" value="Ticket Statuses"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="text" value="stat"/>	<input type="checkbox"/>

For the *Type*, be sure to choose your custom status list so that your statuses will show up there.

Database Setup/Prep

Once the status list is set up, we will begin by finding some information in the database that we will need later on. It is important to do this step after setting up the custom list and adding the field to the form so that you can get the values you need.

The first value we need is your Field ID, which will be used in the *set_ticket_status()* function, in class.ticket.php.

Includes -> Staff -> tickets.inc.php

There are a couple of changes that need to happen in here. First, you want to find where your select statement is in here. It looks like this:

```
$qselect='SELECT DISTINCT ticket.ticket_id,lock_id,ticketID,ticket.dept_id,ticket.staff_id,ticket.team_id '
.',MAX(IF(field.name = \'subject\', ans.value, NULL)) as `subject`'
.',MAX(IF(field.name = \'priority\', ans.value, NULL)) as `priority_desc`'
.',MAX(IF(field.name = \'priority\', ans.value_id, NULL)) as `priority_id`'
```

Add these lines into the statement (make sure to mind where the ; is at the end of \$qselect):

```
.,MAX(IF(field.name = \'stat\', itm.value, NULL)) as `stat_name`  
.,MAX(IF(field.name = \'stat\', itm.extra, NULL)) as `stat_color`
```

In the \$qfrom variable, we need to add the new ost_list_items to the INNER JOIN list, so that those items can be retrieved. You need to find:

```
$qfrom=' FROM '.TICKET_TABLE.' ticket '  
' LEFT JOIN '.USER_TABLE.' user ON user.id = ticket.user_id'.  
' LEFT JOIN '.USER_EMAIL_TABLE.' email ON user.id = email.user_id'.  
' LEFT JOIN '.DEPT_TABLE.' dept ON ticket.dept_id=dept.dept_id'.  
' LEFT JOIN '.FORM_ENTRY_TABLE.' entry ON entry.object_id=ticket.ticket_id'.  
' LEFT JOIN '.FORM_ANSWER_TABLE.' ans ON ans.entry_id = entry.id'.  
' LEFT JOIN '.FORM_FIELD_TABLE.' field ON field.id=ans.field_id';
```

Insert the following line beneath the LEFT JOIN for the FORM_ANSWER_TABLE.

```
' LEFT JOIN ost_list_items itm ON itm.id=ans.value_id'.
```

I have not looked at the table name variables which is why it is listed as ost_list_items.

[OPTIONAL] If you want to have your tickets sorted on the ticket queue by the status name, you will want to modify \$order_by to include the status name.

```
if(!$order_by) {  
  if($showanswered)  
    $order_by='ticket.lastresponse, ticket.created'; //No priority sorting for answered tickets.  
  elseif(!strcasecmp($status,'closed'))  
    $order_by='ticket.closed, ticket.created'; //No priority sorting for closed tickets.  
  elseif($showoverdue) //priority> duedate > age in ASC order.  
    $order_by='priority_id, ISNULL(duedate) ASC, duedate ASC, effective_date ASC, ticket.created';  
  else //XXX: Add due date here?? No -  
    $order_by='building ASC, stat_name ASC, priority_id, effective_date DESC, ticket.created';
```

Here, you are going to want to change the \$order_by under the else statement to include your stat_name, which is the name we gave our variable above. **Remember**, the order goes from left to right, so if you want the status to be the first thing it sorts by, put it at the beginning of the statement.

Next, we have to make the queue show your status. Part one of this process is to add the header to the table. Because I did not use the ticket priority, I simply replaced it with the status. If you do not want to do that, you can copy one of the other <th> rows and paste it. The code you are looking for is:

```

<thead>
  <tr>
    .....
  </tr>
</thead>

```

I have not added in the ability to sort by status, so my column simply looks like this:

```
<th width="120">Status</th>
```

Make sure that you pay attention which columns you put the status field between. This way, when you insert the <td>, your columns still line up. Next, we need to add the actual row of data for each ticket. You want to find the following:

```

<tbody>
  <?php
    .....
  endif; ?>
</tbody>

```

In there, you will add this line:

```

<td style='text-align:center;'><div id='stat' style='background-color:#<?php echo
$row['stat_color'];?>';><?php echo $row['stat_name'];?></div></td>

```

Note that the # is in the style for the div. If you put the # in the extra field of the custom list, make sure you remove the # here.

Includes -> class.ticket.php

You need to add three new functions in this class. You can paste them anywhere in class.ticket.php for the most part.

```

function setTicketStatus($statusID, $created){
    $status_val = (int) $statusID; // only allowed to use int

    $sql = "SELECT e.id, v.value_id FROM ost_form_entry e INNER JOIN ost_form_entry_values v ON
v.entry_id=e.id WHERE e.object_id=" .
        db_input($this->getId()) . " AND e.object_type='T' AND v.field_id='24'";
    if(($res=db_query($sql)) && db_num_rows($res)) {
        list($entry_id, $item_value)=db_fetch_row($res);
    }

    if(!$created || is_null($item_value)){
        $item_value = $this->getStatusName($statusID);
        $sql = "UPDATE ost_form_entry_values SET value='\" . $item_value . "\", value_id='\" . $statusID . \"'
WHERE field_id='24' AND entry_id='\" . $entry_id . \"'";
    }
}

```

```

    $res = db_query($sql);
  }
}

function getStatusName($status){
    $sql = "SELECT value FROM ost_list_items WHERE id=" . $status;
    if(($res=db_query($sql)) && db_num_rows($res)) {
        list($item_value)=db_fetch_row($res);
    }

    return $item_value;
}

function getTicketStatuses(){
    $count = 0;
    $sql = "SELECT id, value FROM ost_list_items WHERE list_id=4 ORDER BY value ASC";
    if(($res=db_query($sql)) && db_num_rows($res)) {
        echo "[" . db_num_rows($res) . "]";
        while(list($id, $value)=db_fetch_row($res)) {
            echo "<option value=" . $id . ">" . $value . "</option>";
        }
    }
}
}

```

Make sure that you change the field_id from 24 to whatever your field id for the status is. This was found in the database/prep section of these instructions.

In the ticket create() function, we need to add a line of code at the end to set the status on the ticket creation. Right above `return $ticket;`, add the following line:

```

//Add status to the ticket
$ticket->setTicketStatus(26, 1);

```

Make sure to change 26 to the value of the status you want to be the default one, as found in the database/prep section of these instructions.

Include -> Staff -> ticket-view.inc.php

The only thing that we need to add here is a way for you and your staff to change the status of the ticket. To do that, you are going to want to find where you want the <select> and <submit> located on your ticket view. I put mine between the Internal and External notes and the posting options.

