

Web Advanced

Verslag

Daniela Lupo

Nida Ilyas

Marnix Meuwissen

Tim Van Den Bosh

Britt Verschakelen

# Handleiding website

Je surft naar: <http://localhost/project/web/www/>

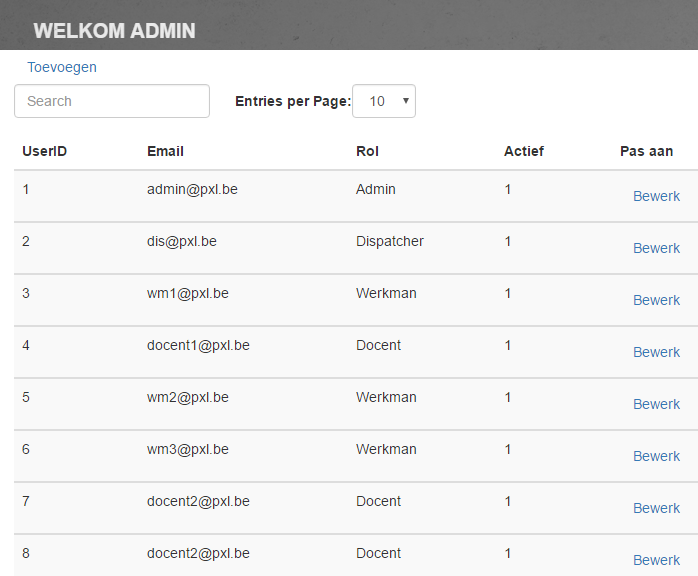
Je komt op de login pagina uit van het ticketingsysteem.

Inloggegevens:

|  |  |
| --- | --- |
| Inlog | Wachtwoord |
| admin@pxl.be | admin |
| docent1@pxl.be | pxl |
| docent2@pxl.be | pxl |
| wm1@pxl.be | pxl |
| wm2@pxl.be | pxl |
| wm3@pxl.be | pxl |
| dis@pxl.be | pxl |

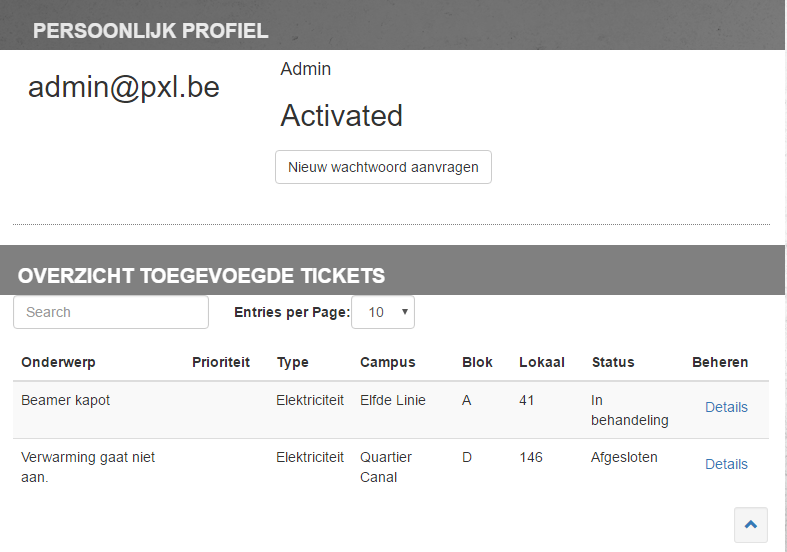
## Admin

Overzicht profielen (welkom pagina admin):

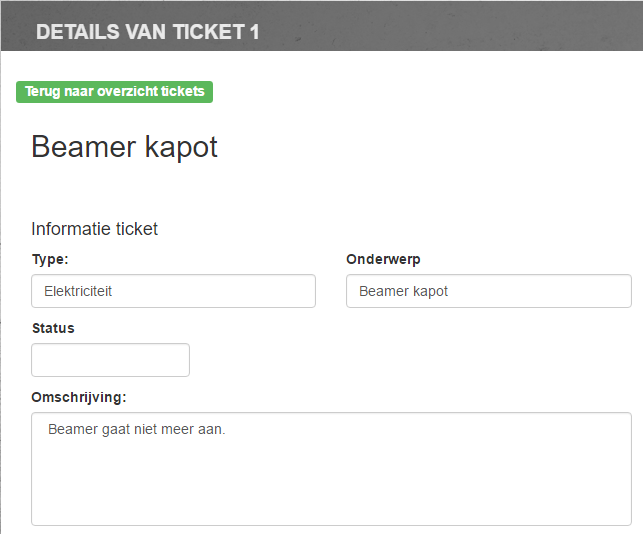


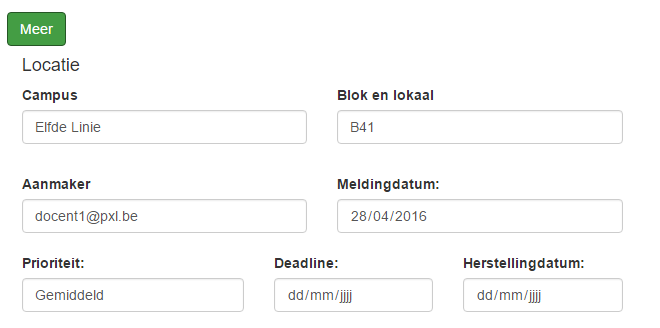
Hier zie je een overzicht van alle profielen. 1 is actief en 0 is non-actief. De admin kan ook profielen bewerken:

Navigatie profiel:



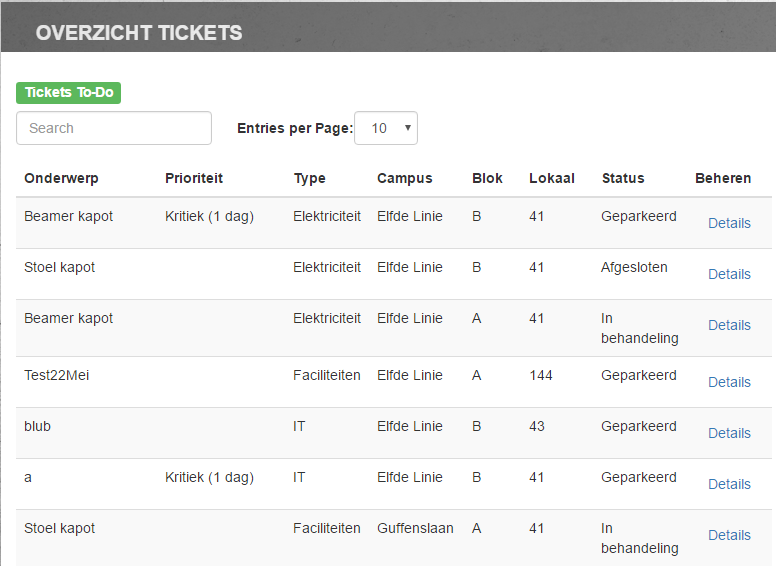
Het eigen profiel overzicht is identiek voor alle users (dispatcher, werkman en docent). Hier kan de user een nieuw wachtwoord aanvragen. Maar ook de details van hun eigen tickets.



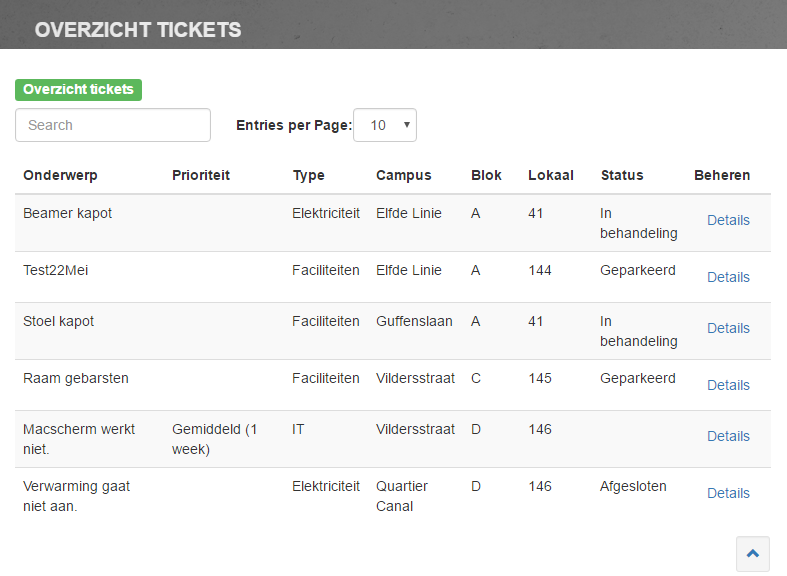


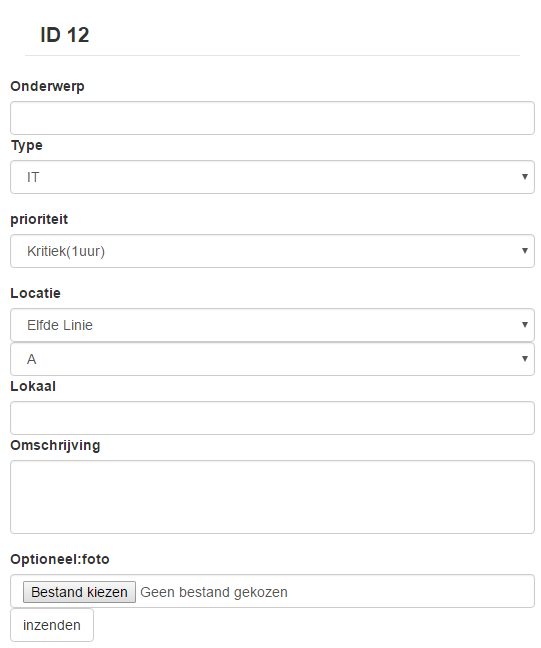
## Werkman

Overzicht tickets (welkom pagina werkman):



Hier ziet de werkman alle tickets die er zijn wanneer de werkman klikt op Tickets To-Do

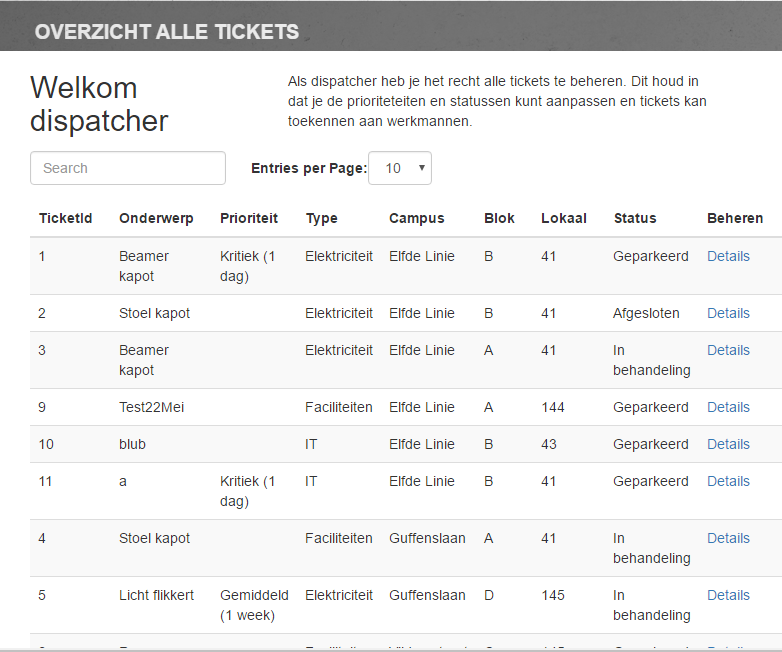
Navigatie nieuw ticket:



Is bij alle gebruikers hetzelfde (Dispatcher, werkman en docent). Hier kan de gebruiker een nieuw ticket aanmaken.

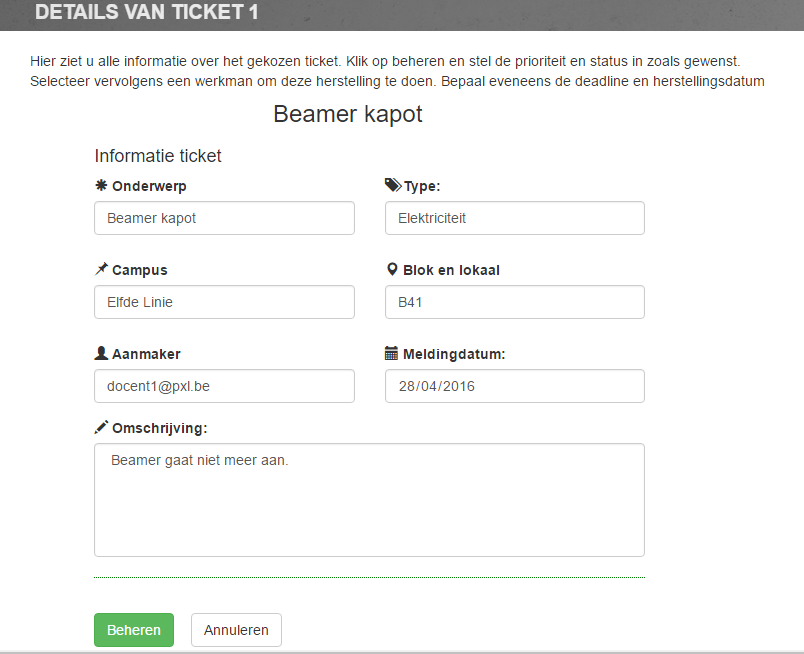
## Dispatcher

Overzicht alle tickets (home pagina dispatcher):

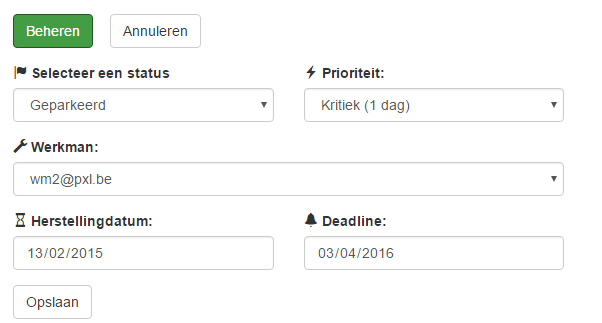


De dispatcher kan alle tickets zien en bewerken bij details.

Details van tickets (laatste kolom van de tickets):

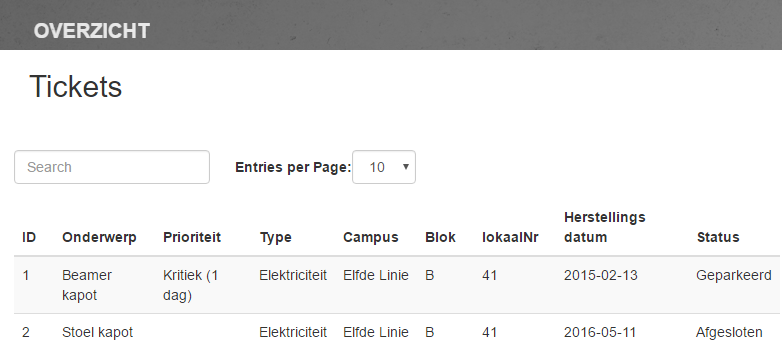


Na het klikken van “Beheren”:



De dispatcher kan ook tickets aanmaken en zijn profiel bekijken.

## Docent (user)

Overzicht eigen tickets (home pagina docent): 

De docent kan ook tickets aanmaken en zijn profiel bekijken.

# Lijst van geleende code-fragmenten

Controllers: verifylogin.php, home.php, login.php

**class** Login **extends** CI\_Controller  
{  
 **function** \_\_construct()  
 {  
 **parent**::*\_\_construct*();  
 }  
  
 **function** index()  
 {  
 $this -> **load** -> view(**'login\_view'**);  
 }  
}

$this -> **session** -> unset\_userdata(**'logged\_in'**);  
session\_destroy();  
redirect(**'login'**, **'refresh'**);

**function** logout()  
{  
 $session\_data = $this -> **session** -> userdata(**'logged\_in'**);  
 $userID = $session\_data[**'userID'**];  
 $rol = $this -> **user\_model** -> neem\_rol($userID);  
 $enquete = $this -> **user\_model** -> check\_enquete($userID);

Models: user\_model

function login :

$this -> **db** -> limit(1);  
  
$query = $this -> **db** -> get();  
  
*// Als de gebruiker gevonden wordt in de db***if**($query -> num\_rows() == 1)  
{  
 **return** $query -> result();  
}  
**else**{  
 **return false**;  
}

is gebaseerd op: <http://www.iluv2code.com/login-with-codeigniter-php.html>

# Codeconventions

## Bekendmaking in commentaar

/\* @author = Naam

\* Date = dd/mm/yyyy

\* Bron = link fso

\* Uitleg functie

\*/

## Edit van code onder aanmaker zetten

/\* @edit = Naam

\* Date = dd/mm/yyyy

\* Bron = bron

\* Uitleg functie

\*/

## Algemene regels

* Inspringen met tab
* Keywords grote letters (FALSE, TRUE, ...)
* Variabele declaratie: camelcasing, \_, begint met kleine letter ditIsEenVoorbeeld1
* Klasses beginnen met een hoofdletter
* Methodes beginnen met een kleine letter

## Spaties

* No: $a=$b+$c; $this->load->view('');
* Yes: $a = $b + $c; $this -> load -> view('');
* geen spatie bij parameters: $a = getFoo($b);
* geen spatie bij geen paramater: $c = getBar();
* spaties bij commentaar: // dit: is een goed voorbeeld //dit is geen goed voorbeeld (geen spatie bij begin)
* functies en lussen: function test() { hier komt code; }

## Controllers

* Geen echo/print
* Geen SQL

## Models

* geen html

## vieuws

* geen SQL