**Documentation technique**

**Groupe SIO1**

**http://www.si6andrewiseppi.esy.es**

**Iseppi Andrew**

Table des matières

[Présentation du travail 2](#_Toc417838083)

[Objectifs 2](#_Toc417838084)

[Méthodes 3](#_Toc417838085)

[Accès 5](#_Toc417838086)

[Avancement du projet 7](#_Toc417838087)

[Contenu : 7](#_Toc417838088)

[Tableau de bord / Accueil du site 7](#_Toc417838089)

[News 7](#_Toc417838090)

[Presse 8](#_Toc417838091)

[Compte utilisateurs 8](#_Toc417838092)

[Création et ajout 8](#_Toc417838093)

[News 9](#_Toc417838094)

[Presse 9](#_Toc417838095)

[Compte utilisateurs 10](#_Toc417838096)

[MySql 12](#_Toc417838097)

[Base de données 12](#_Toc417838098)

[Structures des tables 13](#_Toc417838099)

[News 13](#_Toc417838100)

[Presse 13](#_Toc417838101)

[User 13](#_Toc417838102)

# Présentation du travail

## Objectifs

1. Intégrer un template (template signifiant modèle, le modèle du site, sa mise en forme etc): ceci consistait à donner une apparence à notre site en récupérant un template ainsi que son code dans l’index afin de la reproduire et de la remanier pour nôtre site.
2. Créer les fonctionnalités demandées dans un brouillon sans qu’elles n’enregistrent d’informations dans la base de données.
3. Mise en place la base de données sous MySql (ici à l’aide de PHPMyAdmin via wampserver) afin de pouvoir enregistrer et stocker ces informations.
4. L’objectif final était d’avoir un affichage « public » pour les visiteurs, qu’ils puissent voir le contenu sans tous ses détails et de pouvoir nous permettre avec un compte ayant les droits de modifier ou supprimer le contenu (cela devait former la sécurité du site).

## Méthodes

1. Pour le premier objectif il nous a fallu faire une recherche afin de trouver un template sur le web (de préférence simple et épuré afin de pouvoir l’adapter facilement à nôtre utilisation).

Nous avons utilisé nos compétences en PHP afin de pouvoir recopier le code du fichier index.php pour l’appliquer à notre index.php, nous avons ensuite remis le template dans le dossier du site (ici le template a été trié dans des dossiers et donc des modifications dans les chemins d’accès indiqués dans l’index ont été nécessaires afin de pouvoir continuer d’utiliser le CSS)

1. Nous avons créé les méthodes et les pdo, nous avons également ajouter des pages et des liens pour renvoyer vers elles, une fois sur ces pages nous avons ajouté les liens pour utiliser les fonctions (ici le site fonctionne à partir de Controllers).

Controller de news:

<?php

require\_once "pdo/MyPdo.php";

require\_once "pdo/NewsPdo.php";

require\_once "class/News.php";

//fait appel à ces fichiers nécessaires

$connection = new NewsPdo();

//Si la variable $\_GET["action"] existe

if(isset($\_GET["action"]))

{

//Récupération de l'action passée dans l'url

$action=$\_GET["action"];

switch ($action) {

case 'create':

include("page/news/create.php");

break;

case 'store'://récup des infos du formulaire

$uneNews=new News();

$uneNews->titre=$\_POST["titre"];

$uneNews->description=$\_POST["description"];

$connection->create($uneNews);

break;

case 'delete':

$lewNews=$connection->delete($\_GET["id"]);

break;

case 'modify':

$lewNews= new NewsPdo();

$lewnews=$lewNews->edit($\_GET["id"]);

include("page/news/modify.php");

break;

case 'login':

$unUser=new User();

$unUser->login="";

$unUser->password="";

$\_SESSION["user"]=$unUser;

header('Location: index.php');

break;

default:

include("page/acceuil.php");

break;

}

}

//si pas d'action dans l'url

else

{ $lesNews=$connection->getAll();

include("page/news/index.php");

}

?>

Par exemple le bouton news sur le site donne ce lien :

<http://si6andrewiseppi.esy.es/index.php?controller=news>

Ceci correspond au Controller des news, il gère les actions faire sur les news (donc par défaut si il n’y a pas d’action il envoie sur l’accueil des news)

En second : <http://si6andrewiseppi.esy.es/index.php?controller=news&action=create>

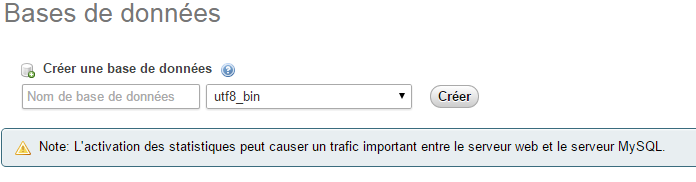
Ce lien renvoie au Controller l’action « create » et l’exécute, ce qui nous renvoie au formulaire :

case 'create':

include("page/news/create.php");

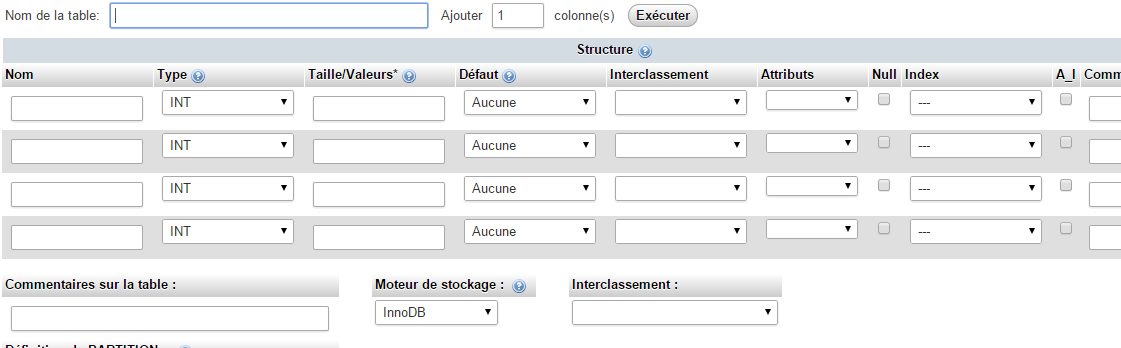
break;

1. Pour ce troisième objectif j’ai utilisé PHPMyAdmin fourni avec WampServer



Cette base est au format utf8\_bin.

Il a fallu ensuite ajouter les tables et leurs structures :



Les structures des tables actuelles sont disponibles dans la partie MySql.

1. Affichage public :



Cependant les fonctions liées aux utilisateurs ne fonctionnent pas encore.

## Accès

**Compte utilisateurlambda du site :**

Nom : userlambda

Mot de passe : 770hfEI

**Compte Admin404 du site :**

Nom : admin404

Mot de passe : Ga72ng2Q

**Compte FTP Admin :**

Nom : u814013179.admin404

Mot de passe : T9SIBqUZ9vqk

**Compte Base de données MySql :**

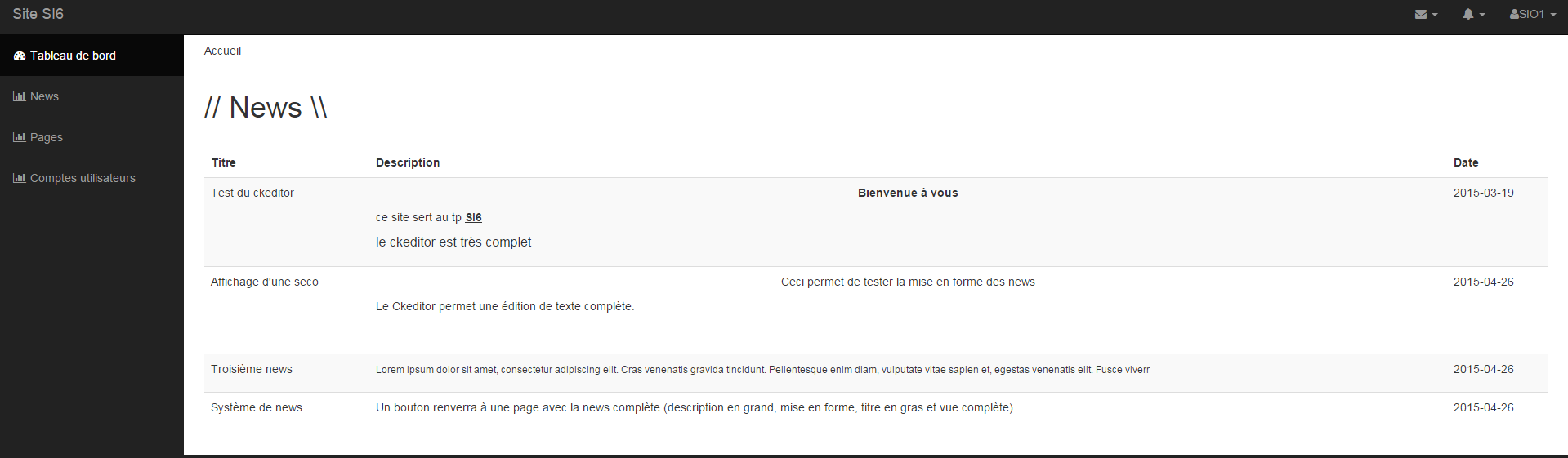
Nom : u814013179\_404

Mot de passe : dw7CfDvYfsPwu

# Avancement du projet

## Contenu :

### Tableau de bord / Accueil du site



Le tableau possède uniquement ces 4 colonnes dans l’accueil, c’est dans la section News qu’il sera plus détaillé et plus complet.

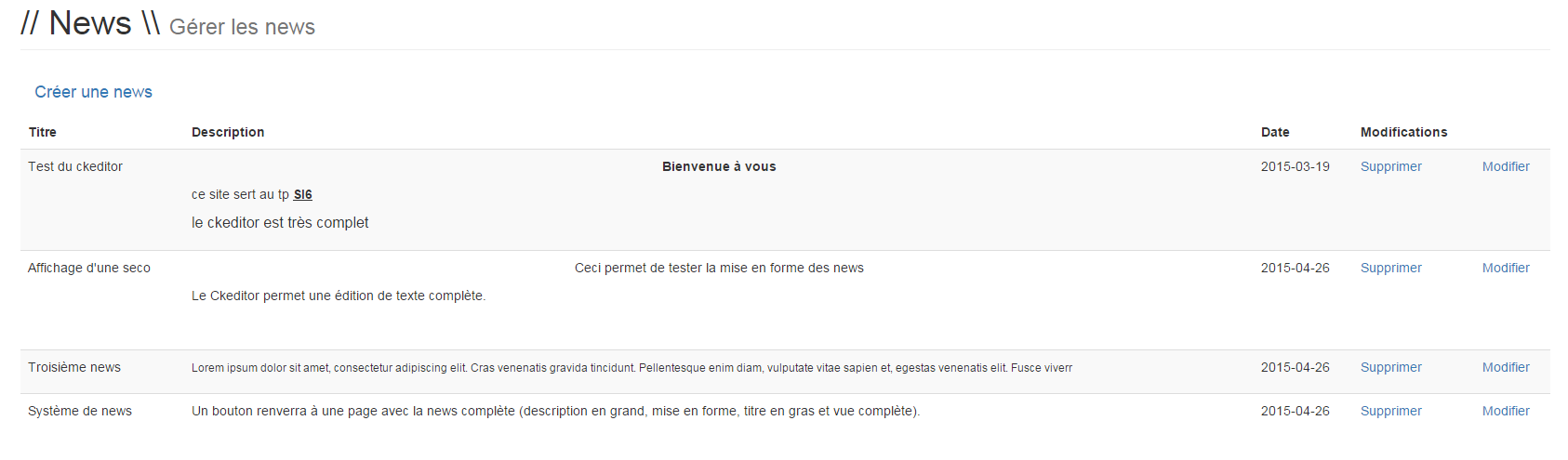
Le site est trié et les fonctionnalités sont regroupées dans des dossiers, ce tri permet l’organisation, le travail en équipe, le détail et la reprise du travail.

1. News (affiche et gère les news)
2. Presse (ou Pages en fonction de sa futur évolution)
3. Comptes utilisateurs (affiche et permet de supprimer les utilisateurs)

### 

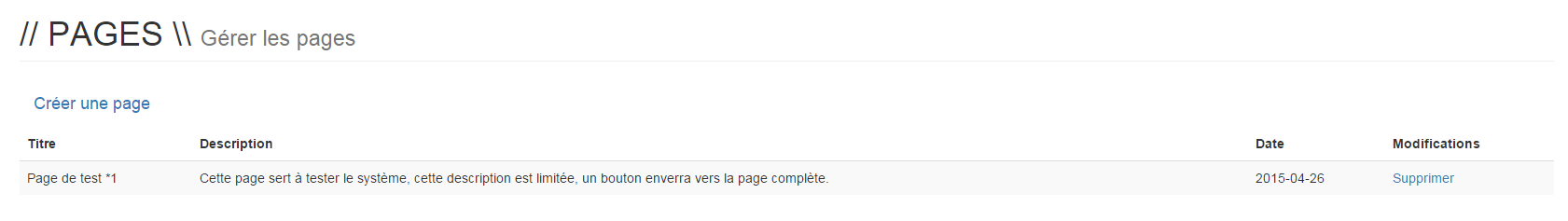
### News

Les news sont affichées dans cette page sous forme de tableau :



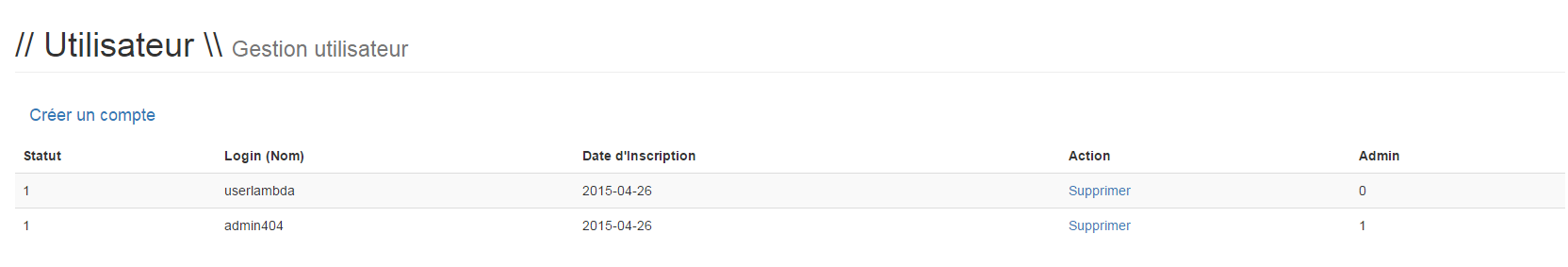
Sur cette page apparaît le bouton pour créer une news. Le bouton « Modifier » n’est pas encore prêt pour être utilisé mais le bouton pour supprimer fonctionne, il récupère l’id de la news et la met dans l’url avec l’action delete pour faire appel à la fonction delete.

### Presse



Sur cette page on retrouve la gestion de la presse (renommée en « Pages » afin de répondre à d’autres critères d’utilisation.

### Compte utilisateurs

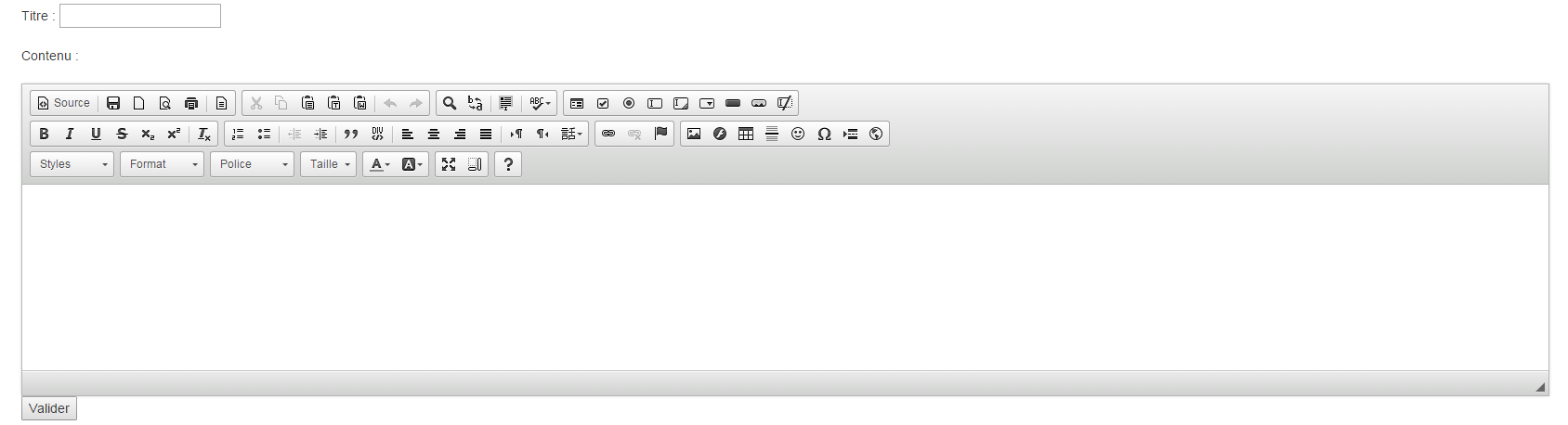


Voici la page de gestion des utilisateurs : la colonne « statut » correspond à la vérification du compte (1 pour activé, 0 pour non activé), la colonne « Admin » donne le niveau de l’utilisateur (1 = admin, 0 = utilisateur lambda).

## Création et ajout

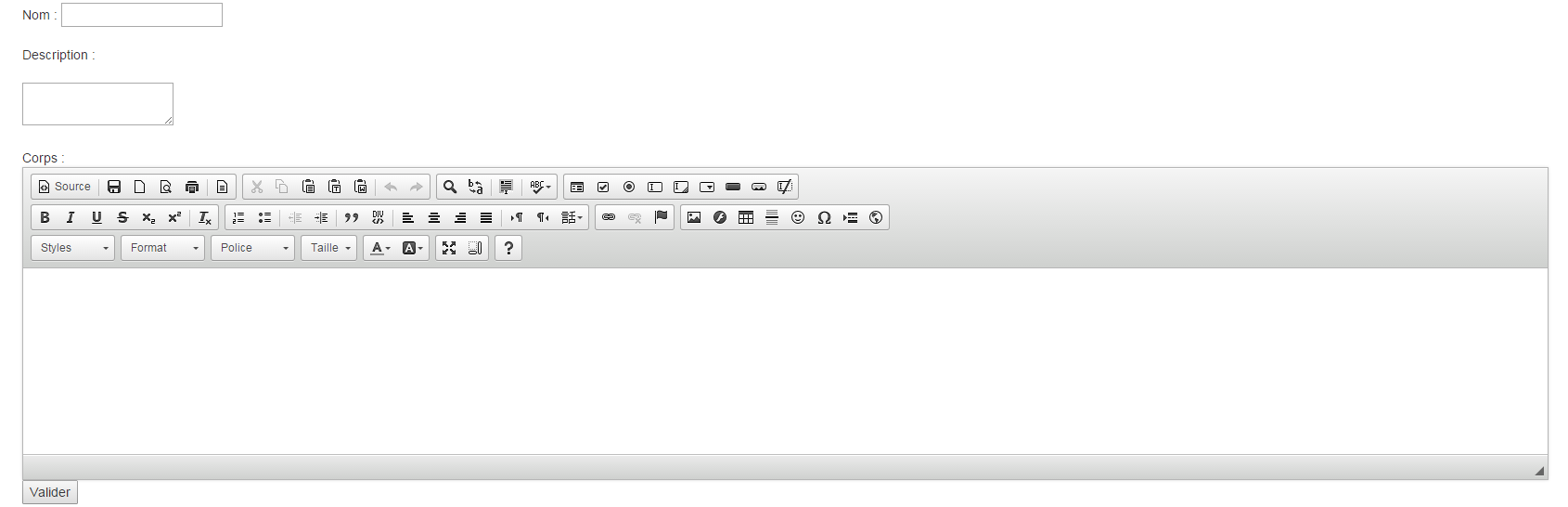
Chaque page renvoi à un formulaire de création pour ajouter une news, un article / une page, un utilisateur en fonction de sa catégorie.

### News



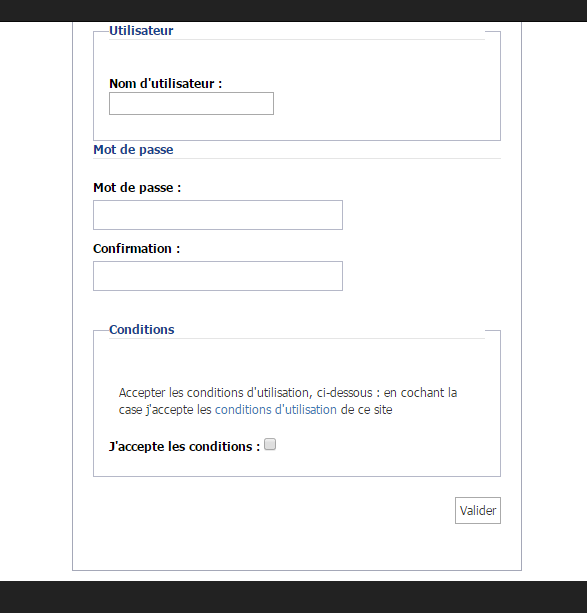
Ce formulaire est simple et inclut le Ckeditor afin de permettre une édition complète de la description.

### Presse

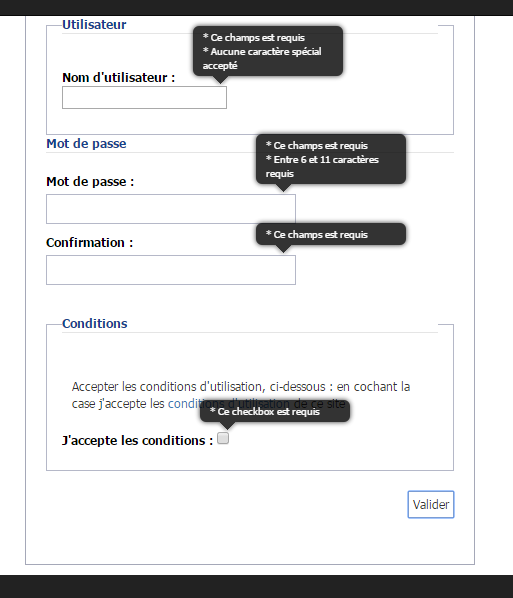


Ce formulaire est simple mais un peu différent, la description n’utilise pas le Ckeditor (car la description restera très courte par rapport à l’article), et le camp « Corps » contient le contenu de l’article, il prend donc le Ckeditor afin de permettre une édition complète et une mise en forme avec ajout de contenu (toujours en maintenance jusque là).

### Compte utilisateurs

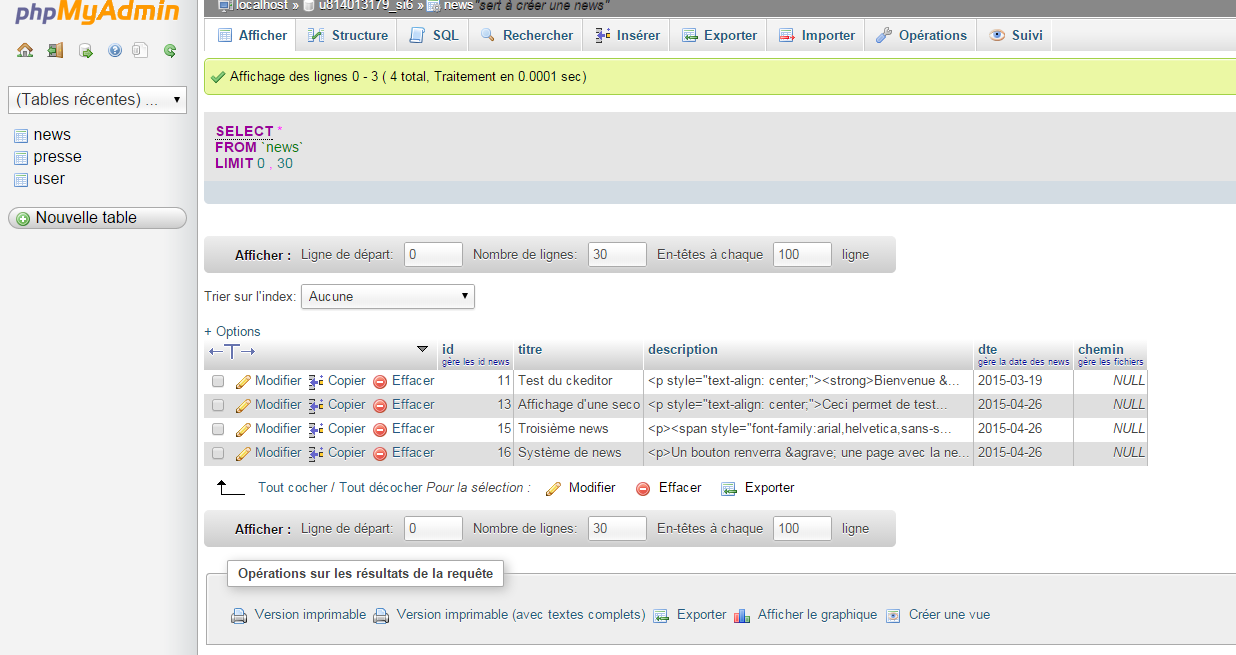


Ce formulaire est une inclusion d’un formulaire en Jquery, il est très intéressant car très complet et permet de renvoyer à des liens (comme les conditions d’utilisation par exemple) et de récupérer les erreurs et de bloquer la validation si celles-ci sont toujours présentes :



# MySql

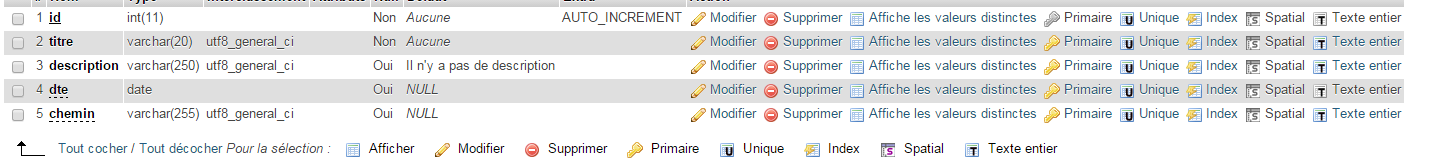
## Base de données



L’arrivée sur la base de données affiche le contenu de la première table (ici news).

## Structures des tables

### News

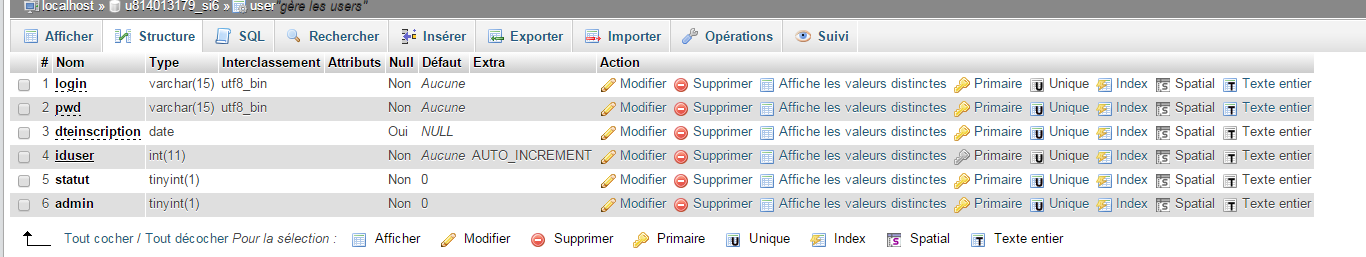
\*

### Presse



\*

### User



\*