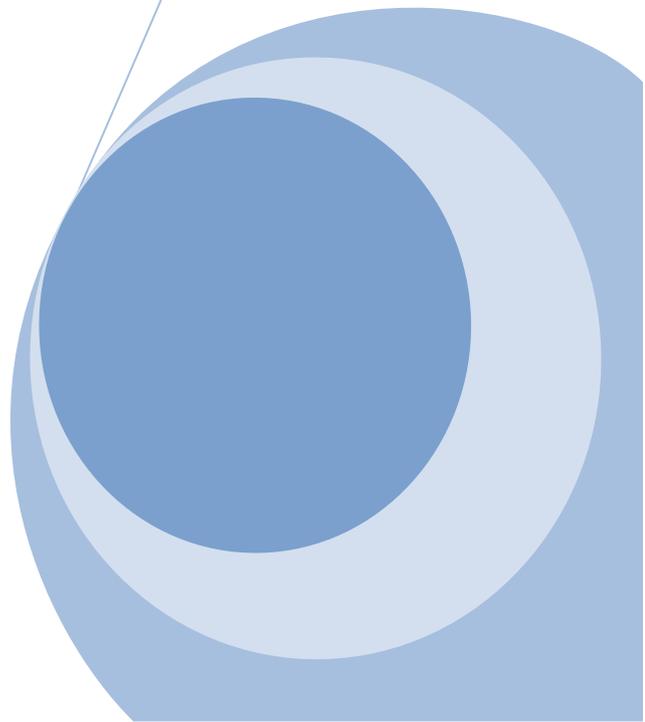


Projet Web

Tim Burton

Refonte complète du site de Tim Burton
en utilisant les dernières technologies du web :
HTML 5 / CSS 3 / JavaScript ...

Par Omar EDDASSER – L3 ISC parcours MIAGE
Sous l'enseignement de : M. BELAÏD
2011 / 2012



Sommaire

I.	Procédure d'installation	- 2 -
II.	Structure du projet : Aspect Multicouches	- 2 -
III.	La base de données	- 3 -
IV.	Langage PHP	- 3 -
V.	Le Framework.....	- 4 -
	1) Choix du Framework.....	- 4 -
	2) Implémentation du Framework	- 4 -
	a) Arborescence générale de l'application.....	- 4 -
	b) Les modèles	- 5 -
	c) Les contrôleurs	- 5 -
	d) Les vues	- 6 -
VI.	HTML 5 – CSS 3	- 6 -
	1) Structure HTML 5	- 6 -
	2) Animation CSS 3	- 9 -
	3) Audio	- 9 -
	4) Vidéo.....	- 9 -
VII.	Librairies compatible HTML5 / CSS 3.....	- 9 -
	1) JQuery.....	- 9 -
	2) Yahoo User Interface	- 11 -
VIII.	Sécurité.....	- 11 -
IX.	Tablette tactile, Smartphone.....	- 12 -
X.	Cas de problème.....	- 12 -

I. Procédure d'installation

Les étapes :

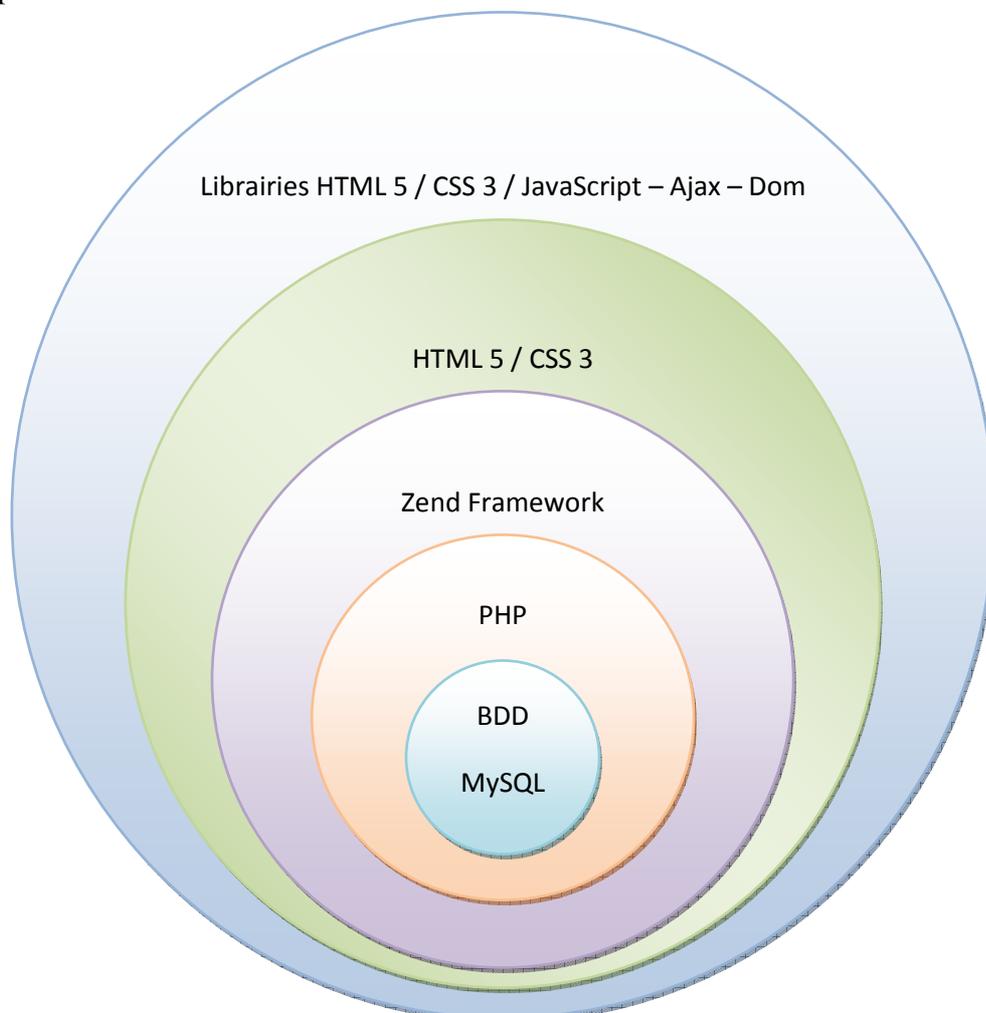
- Importer sous MySQL la base de données fournit dans le fichier « **export_base_sql.sql** »
- Copier le répertoire « **Projet-Web-Tim-Burton** » dans le répertoire « **www** » de votre serveur
- Eventuellement modifier le fichier de configuration « **config.ini** » si vous avez par exemple un mot de passe ou un nom d'utilisateur pour accéder à la base de données.

C'est prêt

II. Structure du projet : Aspect Multicouches

Mon projet a été bâti sur un aspect multicouches utilisant un maximum de technologie :

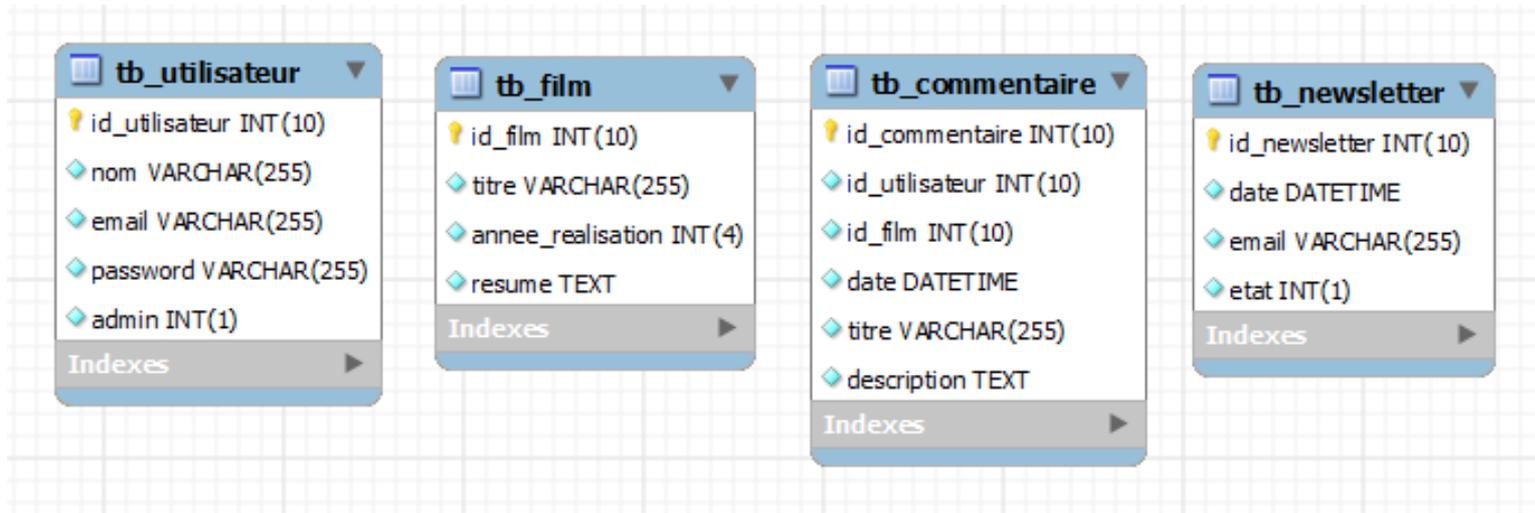
- 1) La Base De Données (BDD): pour le stockage des informations (sous MySQL) : Il s'agit du noyau de l'application
- 2) PHP : Langage qui interagit et fait le lien avec la base de données
- 3) Zend Framework : Couche supplémentaire à PHP : génération du HTML/CSS/JS (couche 4), XML et AJAX
- 4) HTML5 / CSS3 : Couche interaction utilisateur
- 5) Bibliothèques compatibles HTML5 / CSS 3 : JQUERY, YUI, Scriptaculous : couche supplémentaire au HTML5/CSS3/JS



Dans ce rapport, je traiterais de ces couches dans l'ordre.

III. La base de données

Voici le modèle de la base :



Mon site utilise 4 tables :

- **Utilisateur** : recense tous les utilisateurs de l'application, un booléen permet d'identifier les administrateurs
- **Film** : recense tous les films
- **Commentaire** : recense tous les commentaires de chaque film (avec l'utilisateur)
- **Newsletter** : liste toutes les adresses mails des internautes s'étant inscrit à la newsletter du site

IV. Langage PHP

Au niveau du langage, c'est PHP qui a été utilisé, car d'une part imposé par le sujet et d'autre part, car la plupart des hébergeurs ne dispose pas de serveur avec Glassfish ou encore Tomcat, donc pas de possibilité d'utiliser du JEE (Java Entreprise Edition).

De plus, PHP est parfaitement adapté aux besoins de l'application et je dispose d'une maîtrise du langage. Le site repose donc entièrement sur le langage interprété PHP.

V. Le Framework

1) Choix du Framework

Dans le but de réduire le temps de développement, de conserver une homogénéité du code et une pérennité dans ce projet, mais aussi de sécuriser plus facilement l'application, le site utilise Zend Framework.

Le choix a été porté sur ce Framework, car :

- Il est très documenté en anglais et en français (<http://framework.zend.com/manual/fr/>),
- Il dispose d'une très grande communauté notamment francophone (<http://www.z-f.fr/>),
- Il est distribué sous la « New BSD license » (licence libre),
- Il est construit en utilisant 100 % de code Orienté-objet,
- Il permet d'utiliser nativement le principe de MVC (Modèle-Vue-Contrôleur) mais ne l'oblige pas,
- La structure de ses composants est quelque peu unique, c'est-à-dire que chaque composant est conçu avec de faibles dépendances envers les autres composants. Cette architecture faiblement couplée permet aux développeurs d'utiliser les composants individuellement (développement dit « use-at-will »),
- Il est facile à prendre en main,
- ...

Zend Framework a été créé par Zend Technologies, la société fondatrice du PHP tel qu'il est connu de nos jours, ce qui assure une régularité dans l'évolution du Framework (actuellement en version 1.11). De plus, ce Framework est reconnu et très utilisé par les professionnels, **Google** le présente d'ailleurs parmi ses outils pour développeurs.

Pour finir, Zend Framework est très souple au niveau de la base de données, ainsi on peut utiliser dans un premier temps une base de donnée MySQL puis par la suite Oracle, pour cela, il suffira simplement de spécifier au Framework l'adaptateur à utiliser (Zend_Db_Adapter, pdo_mysql pour MySQL, et pdo_oci pour Oracle,...).

2) Implémentation du Framework

a) Arborescence générale de l'application

Le site utilise le principe du **MVC (Modèle-Vue-Contrôleur)** pour l'architecture générale de l'application afin de simplifier d'éventuelles modifications. En effet, ce « **design pattern** » vise à séparer la logique d'application de la logique d'affichage.

En voici l'arborescence :

- /application** // Comporte les fichiers relatifs à l'application
- /utils** // Les classes générales utilisés par l'application
- /models** // Les modèles
- /modules**
 - /controllers** // Les contrôleurs
 - /forms** // Les formulaires
 - /views** // Les vues
 - /scripts** // Toutes les autres vues
 - /layouts** // Les vues générales
- /resources**
 - /images** // Les icones et images
 - /js** // Les éventuels scripts JavaScript
 - /css** // Design – charte graphique
 - /galerie** // Les fichiers audio / vidéo / images propres à chaque galerie
- /library**
 - /Zend** // Librairie du Framework
 - /yui** // Librairie Yahoo User Interface
 - / (Autres) // Les éventuelles autres librairies
- index.php** // Point d'entrée du portail
- config.ini** // Fichier de configuration de l'application (base de données...)
- .htaccess** // Fichier contenant les directives Apache

b) Les modèles

L'ensemble des tables présentent dans le modèle de données (cf. partie III) sont des classes entendant `Zend_Db_Table_Abstract`. Chaque classe redéfinit les propriétés `$_name` (avec le nom de la table à laquelle la classe est associée) et `$_primary` (pour ce qui est de la clé primaire) en plus de redéfinir les différentes méthodes `find()` (ex : `findById`, `findAll`...).

c) Les contrôleurs

L'ensemble des contrôleurs de l'application listés ci-dessous héritent de `Zend_Controller_Action` :

- `AdministrationController` : contient tous les traitements liés à l'administration (n'est donc accessible que pour un administrateur connecté)
- `BiographieController` : gère la page sur la biographie
- `ContactController` : gère la page de contact
- `ErrorController` : gère les cas d'erreurs
- `FilmographieController` : gère les pages liés à la filmographie (liste tous les films, affiche le détail d'un film en particulier, les commentaires du film...)
- `IndexController` : gère la page d'accueil, c'est le contrôleur par défaut
- `InformationController` : gère tous les affichages informatifs
- `MultimediaController` : gère toutes les galeries (audio, vidéo, images...)
- `NewsletterController` : gère la partie newsletter (inscription / désinscription)

- RobotsController : gère au niveau du référencement le fichier robots.txt
- SitemapController : gère au niveau du référencement le fichier sitemap.xml
- UtilisateurController : gère tous les traitements liés aux utilisateurs (inscription, connexion, déconnexion, modification de profil,...)

Pour des raisons de convention de nommage établies par Zend Framework, tous les noms de contrôleurs doivent se terminer par « Controller ».

d) Les vues

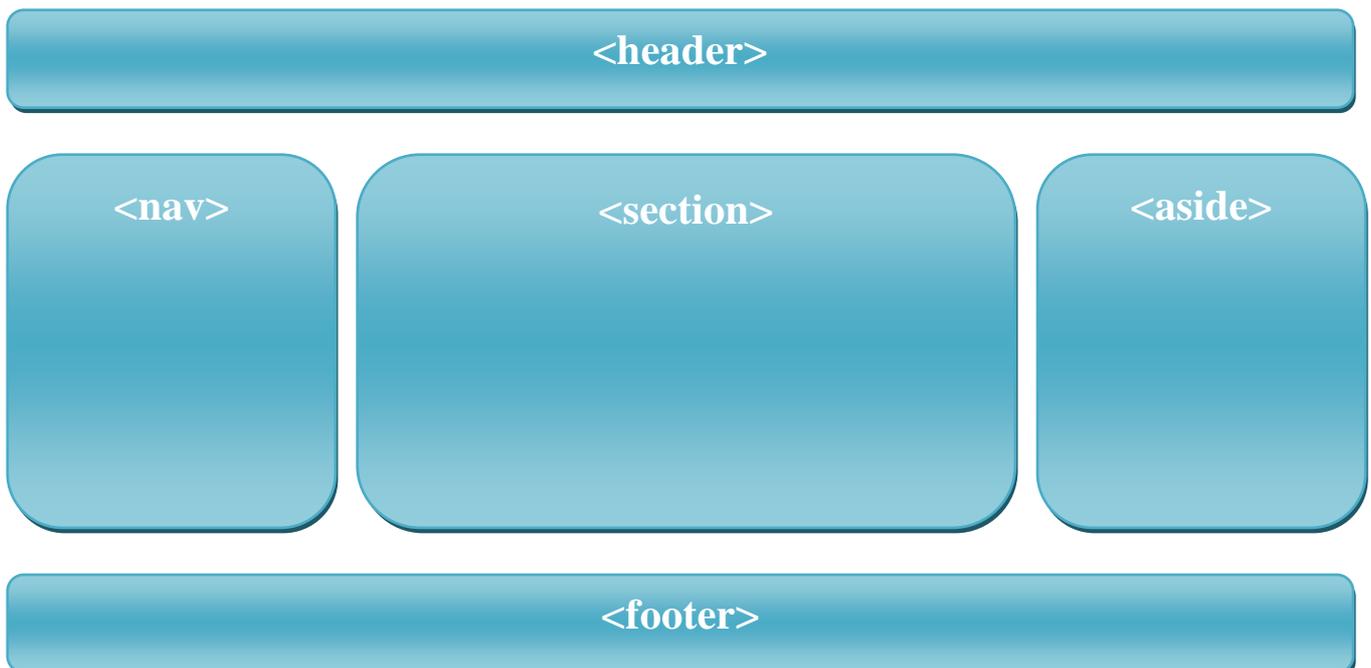
Le Framework veut que les vues générales soient dans le répertoire /application/modules/views/layout ; par vues générales, on entend des vues qui peuvent englober d'autres vues (c'est le cas du layout main) ou des vues qui peuvent être appelée par plusieurs contrôleurs. Toutes les autres vues sont dans le répertoire /application/modules/views/scripts, il existe dans ce dossier, un répertoire par contrôleur et dans chacun d'eux existe une vue par méthode du contrôleur (Il est toutefois possible d'utiliser une autre vue).

Les vues sont des fichiers avec l'extension .phtml, elles sont composées de HTML5 avec des fragments de PHP.

VI. HTML 5 - CSS 3

1) Structure HTML 5

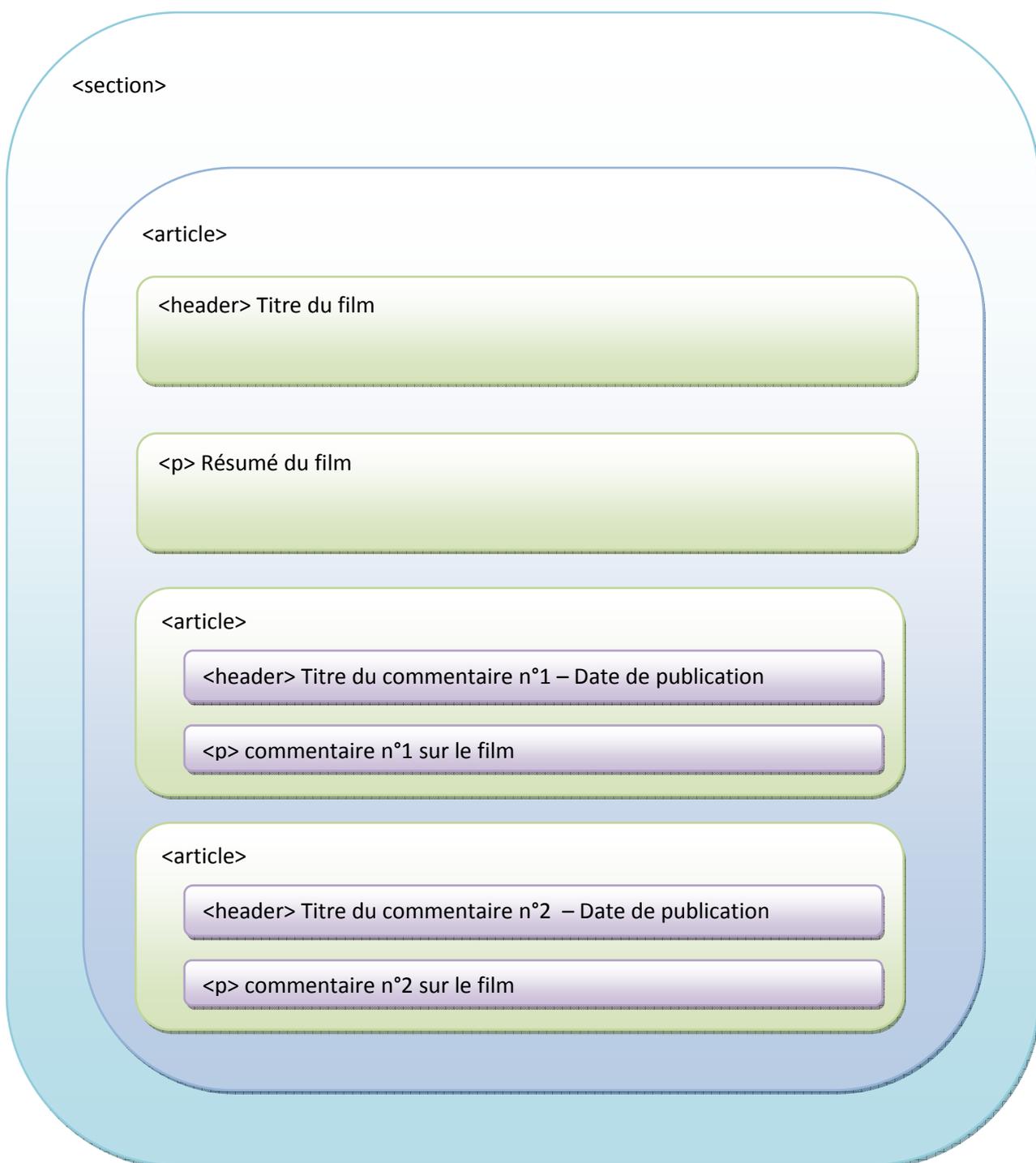
Voici ci-dessous un schéma de la structure adopté pour le site :



Il s'agit de la même structure que celle de « **Smashing Magazine** » vu en cours. Ci-dessous une capture d'écran du site web :



Structure au niveau de la filmographie (détail d'un film) :



Les films sont traités sous la forme **d'article**, et les commentaires des films sous la forme de « **sous-article** »

2) Animation CSS 3

De nombreuses animations CSS 3 sont présentes sur le site web. En voici une liste non exhaustive :

- Effet d'ombrage, changement de couleur et déplacement du titre avec le survol de bannière (le tout avec une durée d'animation +/- rapide)
- Arrondi des conteneurs
- Drag & Drop de tous les conteneurs sur la page Biographie
- Changement de taille, de couleur, de background au survol d'un élément du menu
- Rotation d'images dans la biographie
- Agrandissement des icônes du footer au survol
- Galeries d'images avec différents effets
- ...

3) Audio

En plus de la galerie audio, une balise audio cachée est créée dynamiquement avec du JavaScript permet d'animer encore plus le site. En effet, au survol des différents éléments du menu, une musique différente démarre (et s'arrête avec la fin du survol).

4) Vidéo

La galerie vidéo reprend un lecteur vidéo que j'ai adapté, elle permet notamment d'appliquer différents filtres et transformations sur la vidéo en cours de lecture.

VII. Librairies compatible HTML5 / CSS 3

De nombreuses librairies compatibles avec les normes HTML 5 / CSS 3 ont été intégrées au projet pour le rendre plus animé, ci-après sont décrites les principales.

1) JQuery

JQuery est une bibliothèque JavaScript libre qui porte sur l'interaction entre JavaScript (comprenant AJAX) et HTML, et a pour but de simplifier des commandes communes de JavaScript.

Le Framework contient notamment les fonctionnalités suivantes :

- Parcours et modification du DOM (y compris le support des sélecteurs CSS 1 à 3 et un support basique de XPath) ;
- Événements ;
- Effets et animations ;
- Manipulations des feuilles de style en cascade (ajout/suppression des classes, d'attributs...)

AJAX ;

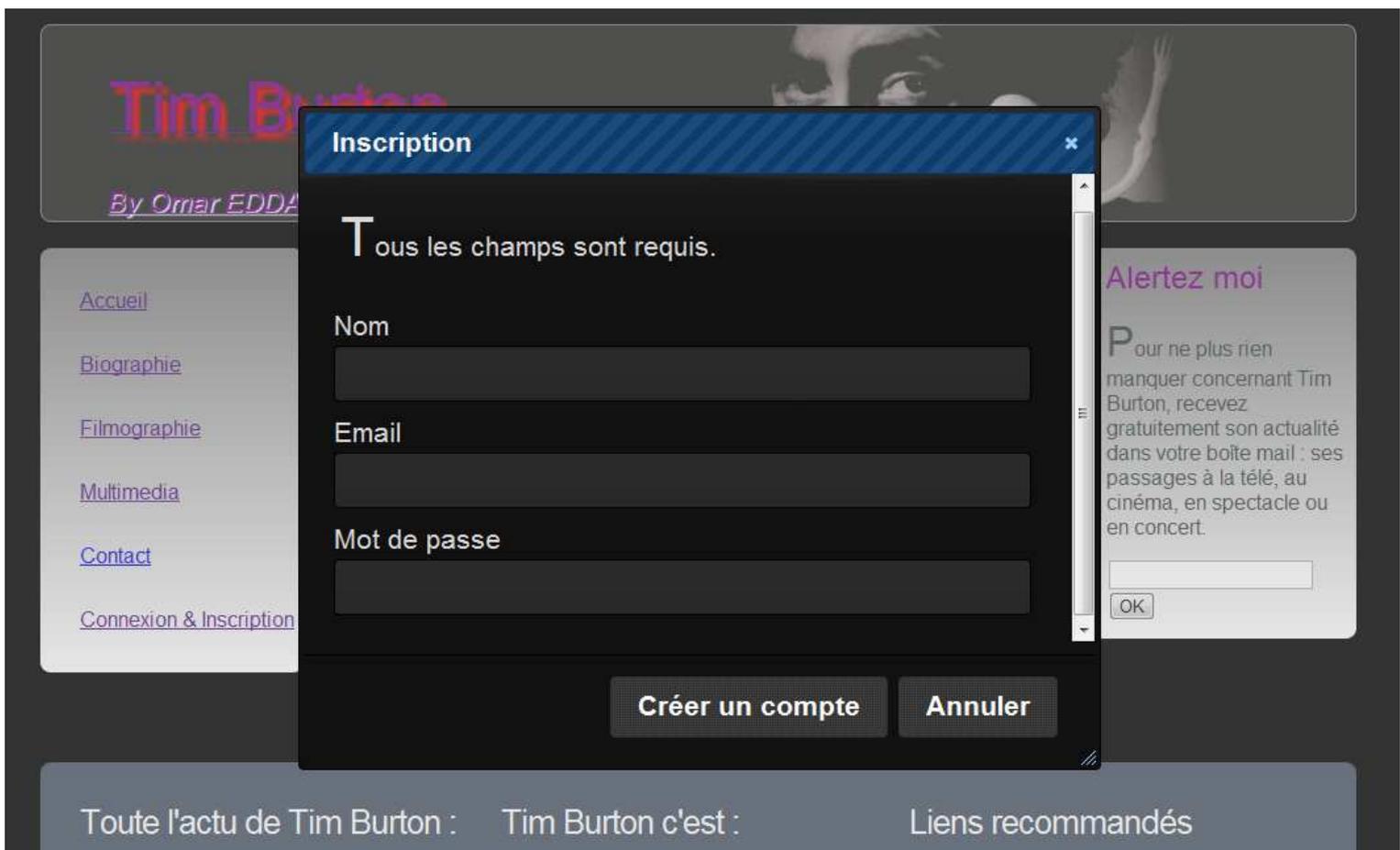
- Plugins ;
- Utilitaires (version du navigateur...).

Dans le cadre de mon projet, elle est utilisée notamment pour avoir des fenêtres de dialogue :

- Pour la connexion
- Pour l'inscription et la modification de profil
- Ou encore, le « A propos » du lecteur de vidéo (dans la galerie vidéo)

En plus d'être utilisée pour simplifier l'usage du JavaScript.

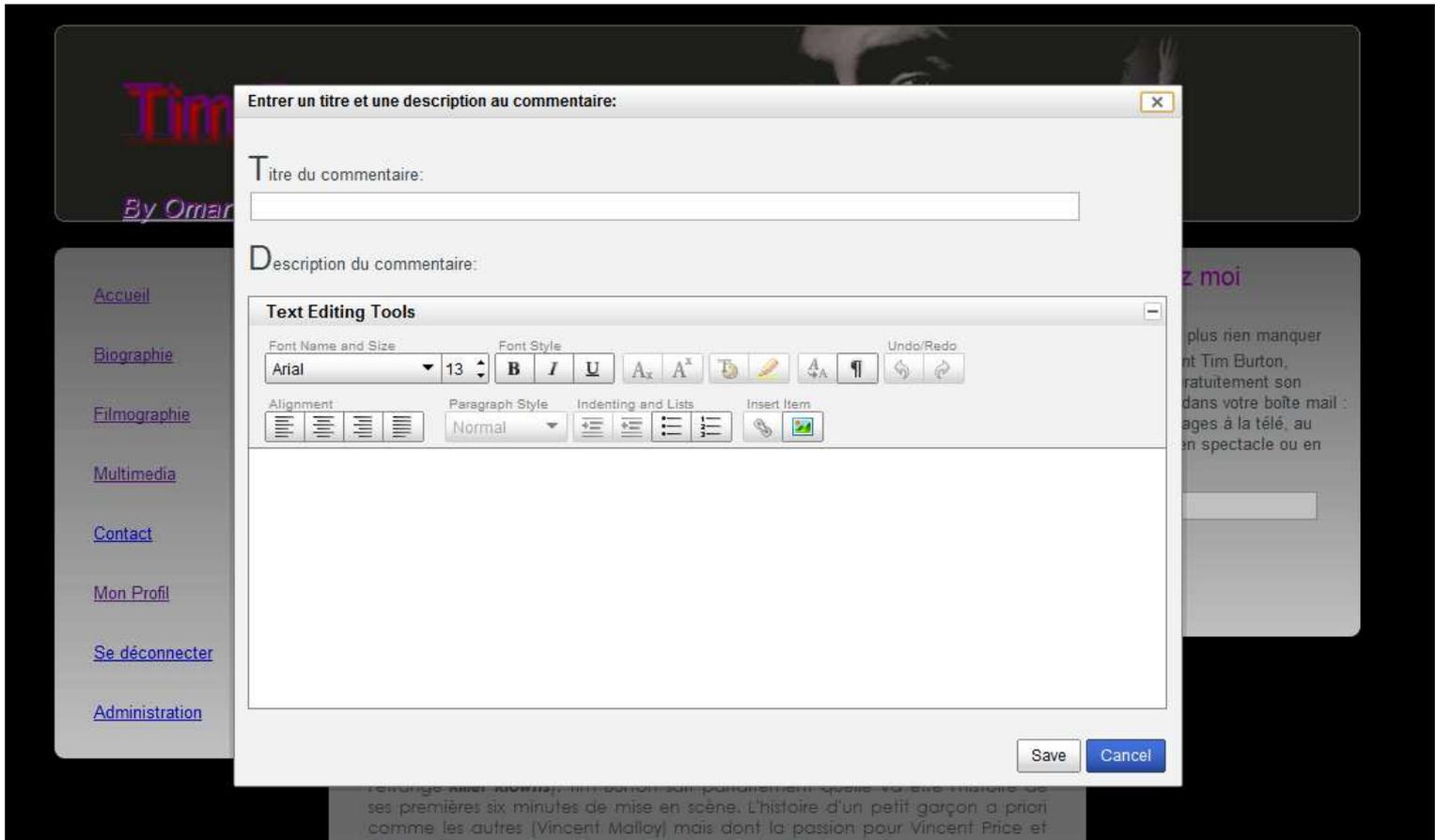
Ci-dessous une capture d'écran d'un exemple d'usage de la librairie :



2) Yahoo User Interface

Yahoo! User Interface Library (YUI) est une bibliothèque logicielle écrite en JavaScript et publiée sous licence BSD par Yahoo! Elle recourt à AJAX, DHTML et DOM, ce qui facilite le développement d'applications web. Elle inclut également plusieurs fonctions permettant d'interagir à la volée avec les ressources CSS.

J'ai utilisé cette librairie par exemple pour l'ajout de commentaire, ci-dessous une capture d'écran :



VIII. Sécurité

Un aspect important du site a été la sécurité. J'ai choisi d'avoir une double-sécurité :

- La première au niveau client avec du JavaScript,
- La seconde au niveau serveur avec du PHP (et Zend Framework)

C'est par exemple le cas au niveau de l'inscription, avant d'envoyer le formulaire au serveur, je vérifie d'abord avec du JavaScript que le formulaire a bien été rempli, que les champs ont le bon format... Puis au niveau PHP, je revérifie qu'il n'y a pas d'erreur.

IX. Tablette tactile, Smartphone

Le site à été conçu et tester pour fonctionner également sous différent supports récent.

J'ai notamment procéder à des tests avec une tablette **Asus Eee Pad** sous **Android**.

X. Cas de problème

En cas de problème lors de l'installation (ou autre), merci de me contacter par email :

omar.eddasser@etudiant.univ-nancy2.fr