

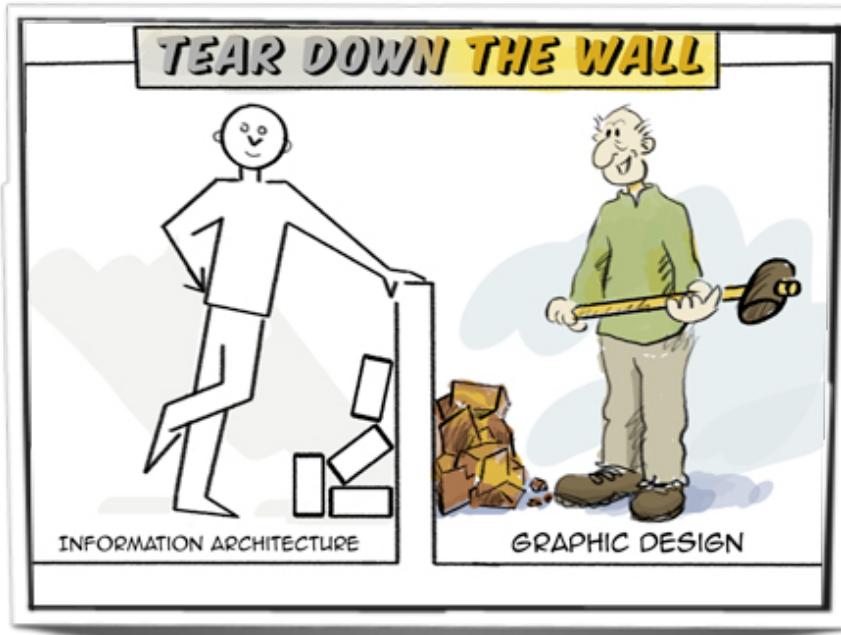
TP mise en place d'une application WebTV

Objectif

L'objectif de ce TP est la création d'une application WebTV avec les fonctions de base.

1. Soumission des vidéos
2. Encodage et traitement
3. Diffusion

Méthodologie



Au sein de groupe CampusPlex on a adopté une méthodologie de développement basé sur l'utilisateur (User-Centric).

La conception centré utilisateur est une démarche ISO qui vise à intégrer l'utilisateur final dans toutes les phases de la mise en place d'une interface, ou autre. l'User-Centric est une démarche qui est également utilisé dans d'autres contextes telles que l'optimisation, la mise en place d'une organisation du travail.

Cette méthode permet de comprendre :

- Le contexte d'activité et organisationnel dans lequel les utilisateurs se trouvent.
- Leurs besoins réels
- Leurs contraintes
- Leurs attentes

Ce TP est un exemple simple de la conception d'une application basé l'User-Centric

User interface de l'application - Homepage

The wireframe illustrates the layout of the WEB TV homepage. At the top, a large header area contains the text "WEB TV". Below this, a search bar is on the left and a "Ajouter une vidéo" (Add a video) button is on the right. A "Titre de la vidéo" (Video title) input field is positioned above a video player. The video player includes a play button, a progress bar, and volume controls. To the right of the video player is a grid of four video thumbnails, each with a yellow sticky note indicating "Player 16:9 640x360". Below this grid is another larger grid of ten video thumbnails, with a central one labeled "Affichage des vidéos les plus récentes" (Display of the most recent videos). At the bottom of the page is a "Nuage de tags" (Tag cloud) containing various tags like "software", "statistics", "teaching", "tool", "tools", "web", "web2.0", "webdesign", "webdev", "tutorial", "tutorials", "tv", "twitter", "typography", "ubuntu", "usability", "video", "videos", "visualization", "work", "writing", and "youtube".

created with Balsamiq Mockups - www.balsamiq.com

User interface de l'application - Page de soumission vidéo

WEB TV

Titre de la vidéo

Tags

Upload video 

created with Balsamiq Mockups - www.balsamiq.com

- Exemple Tags : (ajaccio, campusPlex, Wmaker) [Séparé par des , ou ;]
- Upload :
 - Limité à 25Mo
 - Formats autorisés : (3gp, avi, asf, asx, avi, mov, mpg, mpeg, wma, divx, wmv, flv, m4v, mp4)

Design et Conception de l'interface (1/3)

A partir des interfaces, proposez un design simple en utilisant n'importe quel outil (Photoshop...) ou même en dessinant avec votre crayon.

A savoir avant d'attaquer le design d'une application Web :

- Sujet de l'application web (Video, Map...)
- Le public visé (tranche d'âge, niveau de connaissances, fréquentation prévue...)
- Contraintes :
 - Couleurs
 - Logo
 - Résolution de l'écran :
 - Statistiques :
 - 640 x 480 (0%)
 - 800 x 600 (1%)
 - 1024 x 768 (20%)
 - > 1024 (76%)
- Il existe trois techniques pour s'adapter à toutes les résolutions :
 - Icy : Le site à une largeur fixe et se colle à droite ou à gauche
 - Liquide : Le site à une largeur fixe mais se colle toujours au milieu du navigateur
 - Fluide : Le site à une largeur proportionnelle à celle de navigateur
 - Ce choix il le plus intelligent vue qu'on peut toucher le plus grand nombre d'utilisateurs.
- Exemple d'intégration en CSS :
 - ```
div {
 width: 80%;
 max-width: 1000px;
 min-width: 700px;
}
```
- Réflexion :
  - Rechercher et visiter les sites similaires (hulu.com, youtube, dailymotion, blip.tv....)
  - Simuler son inspiration ([www.misspato.com/](http://www.misspato.com/))

## Design et Conception de l'interface (2/3)

Concevoir le squelette xHTML/CSS de votre application, pensez à séparer les interfaces en plusieurs zones :

Exemple (zone\_header, zone\_footer, zone\_content ....)

Le langage utilisé sera le PHP dans une plateforme LAMP (Linux Apache Mysql PHP)

– Pensez à installé un environnement Web PHP sur votre machine (Wamp, EasyPHP)

Pensez à utiliser le langage PHP comme moteur de template, pour des raisons d'utilisabilité et de simplicité.

Si vous êtes familier avec un autre moteur de template (Zend, Smarty...) utilisez le.

Un moteur de template sert à séparer le HTML d'une application web de son code métier. l'idée c'est de créer des fichiers contenant exclusivement du l'HTML et des balises compréhensibles par le moteur de template. Au moment de l'exécution, ces fichiers de templates seront parsés pour générer le rendu final d'un site.

Exemple pour zone\_header.php

```
<?php
$titre_application = 'toto';
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title> <?=$titre_application?> </title>
</head>
<body>
 src='<?=$logo?>' alt=' '
....
```

Vous pouvez aussi adopter cette démarche :

```
$header = '<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml1-strict.dtd"\n';
$header .= '<html ..\n';
.....
echo $header;
```

## Design et Conception de l'interface (3/3)

Pensez à faire une architecture pour votre application, et séparez vos fichiers de présentation de vos fichiers de codes métier.

Exemple d'architecture MVC (modèle Vue contrôleur):

- index.php
- application
  - ctrl => Les fichiers des contrôleurs (lecture de modèle et renvoie des données à la vue)
  - models => C'est la logique de l'application [interaction avec votre base de donnée]
  - views => HTML...
- library => Ici on met les librairies externes (Zend framework, classes utiles...)
- public
  - css
  - images
  - js

### Comment fonctionne l'architecture MVC :

MVC est un modèle de conception qui impose la séparation entre les données, les traitements et la présentation. C'est pour cette raison que l'application est divisée en trois composants fondamentaux: le modèle, la vue et le contrôleur. Chacun de ces composants tient un rôle bien défini.

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Dans le cas des applications web, il s'agissait historiquement d'une interface HTML. Si le HTML reste l'interface dominante pour les applications web, de nouveaux formats sont rapidement apparus. Ce sont notamment Macromedia Flash et d'autres langages de balises tels que XHTML, XML/XSL, WML et les services web.

La gestion de toutes ces interfaces dans votre application devient de plus en plus difficile. L'un des grands avantages du MVC tient au fait qu'il gère l'utilisation de différentes vues pour votre application. Aucun traitement n'est effectué dans la vue; qu'il s'agisse d'une boutique en ligne ou d'une liste de salariés, elle sert uniquement à afficher les données et permettre à l'utilisateur d'agir sur ces données. Le deuxième composant du MVC, le modèle, représente les données et les règles métier. C'est là que s'effectuent les traitements. Les bases de données en font partie, de même que des objets tels que les EJB et composants ColdFusion. Les données renvoyées par le modèle sont indépendantes de la présentation, c'est-à-dire que le modèle ne réalise aucune mise en forme. Les données d'un seul modèle peuvent ainsi être affichées dans plusieurs vue. Cette capacité permet de factoriser le code, car le code du modèle n'est écrit qu'une seule fois puis réutilisé par toutes les vues.

Pour finir, le contrôleur interprète les requêtes de l'utilisateur et appelle du modèle et de la vue nécessaires pour répondre à la requête. Ainsi, lorsque l'utilisateur clique sur un lien ou soumet un formulaire HTML, le contrôleur ne produit rien et n'effectue aucun traitement. Il intercepte la requête et détermine quels modèles et quelles vues doivent être associés.

## CoreSystem (1/2)

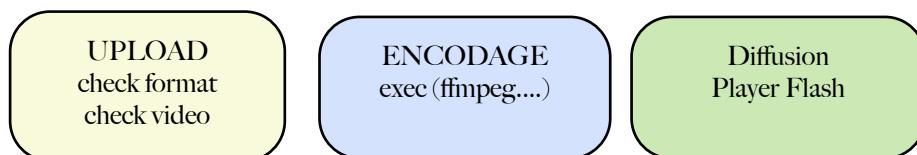
1. Créez la base de donnée Mysql à partir des interfaces fournis.
2. Créez le modèle métier de l'application (Upload, Gestion des vidéos /tags)
3. Documentez vous sur l'encodage vidéo en utilisant l'outil FFMPEG, comment convaincre votre future employeur que l'utilisation de FFmpeg et mieux qu'acheter un outil propriétaire ?
4. Dans ce TP on vous demande pas d'implémenter la solution d'encodage, on suppose que les vidéos que l'utilisateur upload sont déjà adapté pour la diffusion Web. Qu'elle est la solution que vous proposez pour contrôler les vidéos adapté pour le Web.

Pour notre webTV on aimerais automatiser la génération les vignettes (JPEG) à partir des vidéos uploadés

5. Donnez la syntaxe de la commande en ligne qui permet de générer les vignettes JPEG avec une meilleur qualité.

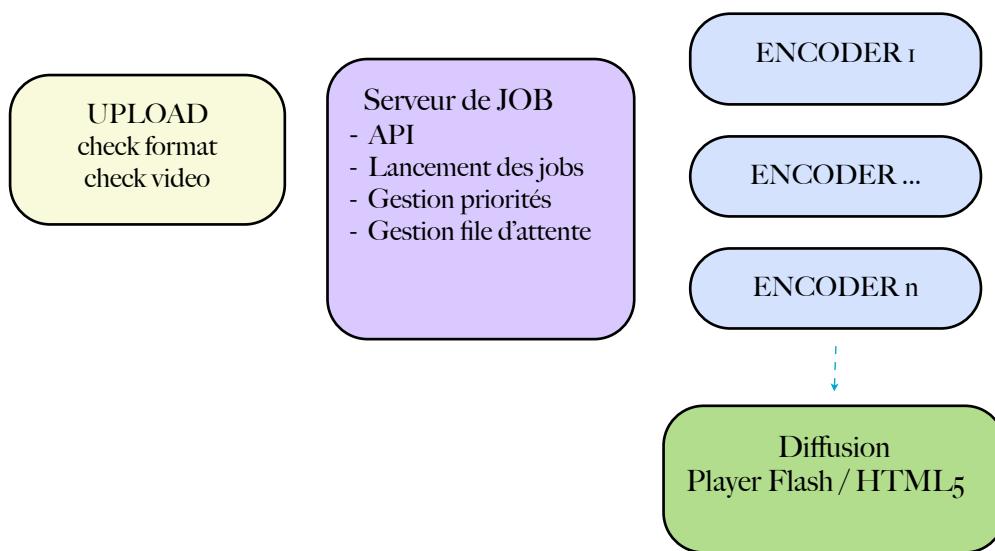
### Modèles d'architectures

Pour notre application de TP le processus d'encodage est optionnel:



Le schéma précédent est simpliste mais ne convient pas pour la monté en charge.

**A titre d'information** voici le schéma qui conviendrait pour une solution de webTV à forte charge



## CoreSystem (1/2)

- 6 . Choisissez votre codec vidéo et justifier votre choix.
7. Choisissez le conteneur adapté pour le player flash.
8. Si on vous demande de faire une version web iphone pour votre webTV, quel format/ conteneur vous allez choisir, et avec quels spécifications pour le codec choisi.
9. Donner la liste des commandes FFMPEG à utiliser pour notre webTV pour générer la vidéo en ce basant sur votre choix technique de codec/conteneur
10. Donner la liste des commandes FFMPEG à utiliser pour générer les images qui correspondent au frame ( $t/2$  de la video) :  $t$  étant la durée totale de la vidéo.

### Exécuter des commandes en PHP

- **string exec ( string \$command [, array &\$output [, int &\$return\_var ]] )**
- **string system ( string \$command [, int &\$return\_var ] )**
- **void passthru ( string \$command [, int &\$return\_var ] )**

10. On aimerais contrôler les vidéos uploadés et renvoyer une erreur à l'utilisateur s'il s'agit d'un fichier avec une séquence vidéo non valide. (Pour avoir une vidéo qui n'est pas valide vous pouvez prendre une image par exemple et changer son extension en .avi)
11. C'est quoi l'avantage des tags par rapport à une recherche classique ?

### Structures des répertoires vidéos

videos/flv => répertoire contenant les fichiers vidéos générés

videos/images => répertoire contenant les vignettes des vidéos

## Diffusion des vidéos

1. C'est quoi l'avantage de Adobe flash pour la diffusion des vidéos ?
2. Trouvez un player vidéo flash open source qui est capable de lire vos fichiers vidéos uploadés;
3. Intégrer ce player flash dans votre application
4. C'est quoi la différence entre le streaming, le pseudo-streaming et le progressive download ?
5. Qu'utilise la plateforme Youtube/Dailymotion par exemple comme méthode de streaming.
6. En terme d'économie de bande passante comparez le protocole RTMP avec le progressive download
7. Pourquoi Youtube n'utilise t'il pas le protocole RTMP ?

Je vous conseille d'utiliser le player flash JW Player : [www.longtailvideo.com/players/jw-flv-player/](http://www.longtailvideo.com/players/jw-flv-player/)

Après l'intégration de player :

8. Que se passe t-il lorsqu'on désire lire une partie de la vidéo qui n'est pas encore chargé dans le buffer de player vidéo?
9. Comment expliquez vous ce comportement ?
10. Suggérez une solution technique pour mettre en place le pseudo-streaming

On aimerais proposer nos vidéos aussi sur l'Ipad de Apple

9. Trouvez les spécification techniques pour encoder les vidéos pour l'Ipad (codec, conteneur, paramètres d'encodage)
10. Est-ce que notre player flash est compatible avec la visualisation sur l'Ipad ? pourquoi ?

11. Visitez cette page : [jilion.com/sublime/video](http://jilion.com/sublime/video), le player proposé par ce site utilise t'il la technologie Flash ?
12. A votre avis avec l'arrivée de l'HTML5 est ce que c'est la fin de flash vidéo.
13. Intégrez une version HTML5 de votre player :

Pour aller plus loin (Optionnel),

- Développez un backoffice pour votre webTV en suivant la même méthode
- Développez la fonctionnalité qui permet d'importer et exporter les vidéos vers Youtube
- Développez un système de partage vers les réseaux sociaux (Facebook/Twitter...)