|  |
| --- |
| INF4173 – Projet Synthèse |
| Rapport final |
| Développement d’un site interactif « French as Second Language » |



|  |
| --- |
| **François Charette Nguyen** 30/04/2010 |

Table des matières

[1. Introduction 1](#_Toc260234742)

[2. Description du projet 1](#_Toc260234743)

[Objectifs 1](#_Toc260234744)

[Méthode de travail 1](#_Toc260234745)

[Outils utilisés 2](#_Toc260234746)

[3. Description de l'implémentation 6](#_Toc260234747)

[Travail d'intégration 6](#_Toc260234748)

[Format des questionnaires en XML 6](#_Toc260234749)

[Interface étudiant 7](#_Toc260234750)

[Interface professeur 8](#_Toc260234751)

[Hiérarchie du répertoire web 9](#_Toc260234752)

[4. Travail futur 10](#_Toc260234753)

[5. Conclusion 10](#_Toc260234754)

[6. Remerciements 10](#_Toc260234755)

# Introduction

Tout d’abord, voici un rappel de la description du projet :

Le projet consiste à élaborer un prototype d’un module de questionnaires pour un site d’apprentissage du français en tant que langue seconde. Le site sera sous la forme d’une plate-forme sociale et interactive sous le nom de « French as Second Language » (FSL). Le but du site est de permettre à des membres anglophones d’apprendre le français en les mettant en contact avec du contenu authentique provenant de la culture francophone. Le projet sera coordonné et supervisé par M. Michal Iglewski avec la collaboration de M. Alain Charbonneau.

Le document suivant servira donc de documenter le travail qui a été fait pour le projet. Il sera premièrement question de décrire le module de façon générale. Ensuite, il sera question de présenter les outils et technologies utilisées pour réaliser le projet. Finalement, une présentation du produit obtenu sera donnée.

# Description du projet

## Objectifs

La première étape a été de déterminer des objectifs plus spécifiques au niveau informatique. Voici la liste qui a été utilisée pour ensuite explorer les possibilités d'implémentation.

* Deux fonctionnalités principales: édition pour les professeurs et consultation pour les étudiants
* Questionnaires centrés sur le contenu vidéo (YouTube)
* Choix de réponse de type texte, audio ou image
* Rapidité et simplicité d'utilisation
* Possibilité de formater l'énoncé (couleurs, tableaux, etc.)
* Possibilité d'envoyer des fichiers images et audio pour enrichir les questionnaires

## Méthode de travail

Au début du projet, il a fallu réunir de l’information sur diverses technologies et évaluer les avantages de chacune. Pour ce faire, j’ai puisé dans mes connaissances acquises lors des cours de technologies internet et programmation web avancée ainsi que dans plusieurs sites de programmation et blogues de programmeurs.

J’ai dès le début choisi d’utiliser les technologies XHTML, CSS, PHP, Javascript et Flash simplement car je le connaissais déjà à divers degré et qu’elles sont des standards répandus dans le développement web pour du contenu dynamique (XHTML/PHP/Javascript) et vidéo (Flash). Pour faciliter ma tâche, j'ai ensuite testé plusieurs outils facilitant la création de contenu web concernant l'intégration de fonctionnalités multimédia, l'utilisation de fonctionnalité pour l'interface graphique et pour créer un éditeur de texte dans le navigateur. Ceux-ci seront présenté dans la section suivante.

## Outils utilisés

Dans cette section, vous retrouverez de l'information sur les outils utilisés pour créer le module de questionnaires. Ces outils sont en quelques sorte les blocs de construction à intégrer ensemble.

### YouTube

**Description:**

YouTube est un site permettant l'hébergement de contenu vidéo gratuitement.

**Raison de l'utilisation:**

* Banque de vidéo déjà énorme
* Gratuit (pas besoin de se soucier de la bande passante)
* L'utilisateur n'a pas a se soucier de l'encodage de ses vidéos
* Fiabilité et rapidité du service
* API complète permettant d'interagir avec le lecteur vidéo sur un site externe

**Aperçu de l'outil:**

****

Voir: <http://code.google.com/apis/youtube/getting_started.html>

### CKEditor

**Description:**

CKEditor est un module en javascript permettant d'incorporer un éditeur WYSIWYG (What You See Is What You Get) HTML pour faciliter le formatage de texte à travers un navigateur web.

**Raison de l'utilisation:**

* Faciliter la rédaction et le formatage des énoncés pour les questions
* Aucune connaissance HTML est nécessaire pour l'utilisateur
* Similaire à un éditeur de texte conventionnel tel que Microsoft Word
* Gratuit et open source

**Aperçu de l'outil:**

****

Voir: <http://ckeditor.com/>

### Swfobject:

**Description:**

Swfobject est une librairie javascript permettant de contrôler les objets SWF (Shockwave Flash)

**Raison de l'utilisation:**

* Librairie requise pour exploiter l'API du lecteur YouTube

Voir: <http://code.google.com/p/swfobject/>

### jQuery

**Description:**

Voici la description officielle de jQuery telle que sur la page principale de leur site (<http://jquery.com/>):

*jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development. jQuery is designed to change the way that you write JavaScript.*

**Raison de l'utilisation:**

* Condenser le contenu des questionnaires sous forme d'accordéon afin d'éviter les longues pages.
* Permettre d'afficher des fenêtres d'aide sans avoir à appeler les fonctionnalité par défaut des navigateurs web (plusieurs utilisateurs ont les "pop-ups" bloqués)
* Esthétisme: permettre d'avoir des animations reflétant les actions de l'utilisateur ("fade-in" lors d'ajout de choix de réponse, "fade-out" lors de l'effacement).
* Librairie requise pour Uploadify

Voir: <http://jquery.com/>

### Niftyplayer

**Description:**

NiftyPlayer est un lecteur mp3 compact en flash et entièrement accessible via le langage javascript.

**Raison de l'utilisation:**

* Pouvoir lire le contenu mp3 envoyé par les professeurs
* Fonctionnalités de base seulement, mais suffisantes (pas besoin de "playlists", de filtre audio, etc.)

**Aperçu de l'outil:**



Voir: <http://www.varal.org/media/niftyplayer/>

### Uploadify

**Description:**

Uploadify est un plugin de jQuery facilitant le téléversement de fichiers vers un site web.

**Raison de l'utilisation:**

* Permettre aux professeurs d'envoyer des fichiers images ou mp3

**Aperçu de l'outil:**



Voir: <http://www.uploadify.com/>

# Description de l'implémentation

## Travail d'intégration

La tâche principale a été de comprendre le fonctionnement de chacun des outils précédemment présentés. La difficulté était dans le fait que chaque outil avait une documentation de qualité variable et des philosophies ou façon de fonctionner différentes. Par exemple, pour certains outils il fallait spécifier l'encodage à plusieurs endroits alors que pour d'autres une seule place suffisait. Certains outils étaient accompagnés de documentation par tutoriels de démonstration alors que d'autres n'avait qu'une documentation technique. De plus, le passage de l'information entre les divers outils et langages représentait un défi.

## Format des questionnaires en XML

Il a fallu créer une structure de données pouvant contenir toutes les informations nécessaires pour les questionnaires pour guider le reste du développement.

Voici un fichier type d’un questionnaire à une seule question :

<?xml version="1.0" encoding="UTF-8"?>

<Questionnaire>

 <type\_questionnaire>formatif</type\_questionnaire>

 <video\_url>http://www.youtube.com/watch?v=c-BYrXY9bfQ</video\_url>

 <titre>Un questionnaire</titre>

 <Question>

 <TypeQuestion>checkbox</TypeQuestion>

 <Enonce>Une question</Enonce>

 <StartTime>120</StartTime>

 <EndTime>150</EndTime>

 <Ponderation>5</Ponderation>

 <Reponses>

 <Reponse reponse\_type="ReponseImage" pts="1">uploadify/uploads/sound.png</Reponse>

 <Reponse reponse\_type="ReponseImage" pts="0">uploadify/uploads/piment.jpg</Reponse>

 </Reponses>

 </Question>

</Questionnaire>

On remarque donc qu’un questionnaire a un type, un lien vers la vidéo, un titre et ensuite une série de questions. Ces questions ont un type, un énoncé, un temps de début et de fin (dans la vidéo), une pondération et des choix de réponses ayant un type et une valeur pondérée.

## Interface étudiant

Figure 1 - Interface de l'étudiant



* L’interface de l’étudiant permet à l’utilisateur de choisir un questionnaire et d’y répondre
* L’interface a un visionneur vidéo à gauche et le questionnaire à droite. Cela permet de répondre aux questions tout en visionnant l'extrait vidéo
* Les questions sont séparées par des onglets se déroulant. Cela permet d’éviter à avoir à utiliser la barre de défilement pour alterner entre la vidéo et les questions
* Chaque question a un lien qui permet de positionner la vidéo au temps spécifié par l’auteur du questionnaire. Cela permet d’éviter à chercher la réponse à une question dans l'extrait vidéo en entier
* La pondération de chaque question est donnée entre parenthèses

## Interface professeur

Figure 2 - Interface du professeur



* L’interface du professeur permet à l’utilisateur de créer des questionnaires en spécifiant une adresse d’un vidéo YouTube, un titre et une description du questionnaire
* L’interface permet d’ajouter des questions parmi trois types:
	+ Cases à cocher (une ou plusieurs bonnes réponses)
	+ Radio (une seule bonne réponse)
	+ Réponse courte
* L’énoncé peut être modifié facilement grâce à l’interface graphique conviviale de CKEditor
* Le questionnaire est sauvegardé en fichier XML de façon transparente à l’utilisateur
* Le professeur peut téléverser des fichiers annexes (images et mp3).

## Hiérarchie du répertoire web

Figure - Hiérarchie du répertoire web



* Dans la racine du répertoire web, nous retrouvons les fichiers PHP pour les diverses pages web
* Dans le répertoire images, nous retrouvons les images utilisées pour l'interface de l'éditeur et du visionneur de questionnaires
* Dans le répertoire JS, nous retrouvons du code javascript créé pour intégrer les différentes parties du module. Nous retrouvons aussi la librairie jQuery
* Dans le répertoire questionnaires\_xml, nous retrouvons les questionnaires sous forme de fichiers XML.
* Dans le répertoire style, nous retrouvons les feuilles de styles CSS
* Dans le répertoire XSLT, nous retrouvons des fichiers xsl pour créer la vue professeur et la vue étudiante
* Dans les répertoire ckeditor, niftyplayer, swfobject et uploadify, nous retrouvons les outils décrits dans la section 2 du document.
* Les images et fichiers mp3 envoyés par les professeurs sont stockés dans /uploadify/uploads/

# Travail futur

Ceci n'étant qu'un prototype, certaines améliorations seraient possible pour améliorer ce module de questionnaires vidéo. Voici quelques exemples d'améliorations possibles:

* Améliorer l’interface pour la gestion des questionnaires
* Améliorer la gestion des fichiers téléversés (images et fichiers mp3)
* Intégrer un module de correction automatique des questionnaires
* Ajouter la possibilité d'avoir des vidéos d'une autre source que YouTube

# Conclusion

Le travail accompli a été fait en trois grandes étapes soit une phase d'analyse, une phase d'exploration et d'apprentissage sur les diverses options disponibles et finalement une phase d'implémentation surtout axée sur l'intégration des outils choisis. Malgré certaines contraintes de temps et certaines difficultés techniques, le projet m'a permis de mettre en pratique plusieurs connaissances acquises lors des mes études en plus d'en apprendre sur des outils de programmation web. Le projet a donc été réalisé à l'aide de plusieurs technologies et libraires existantes. Le tout a finalement abouti à un produit ayant deux fonctionnalités principales soit l'édition et le visionnement de questionnaires.

# Remerciements

Un grand merci à Alain Charbonneau et Michal Iglewski pour m’avoir proposé ce sujet et ensuite guidé et supervisé. Leur support et compréhension m'a été fort utile dans la réalisation de ce projet.