



CI4 : Gestion de l'information / Structures matérielles et logicielles associées au traitement de l'information.

CI5 : Communication entre systèmes

### Tutoriel « Conception d'un site Web pour mobile »

- TP de découverte.
- Application, mise en œuvre de savoirs/savoir-faire
- Recherche et validation de solutions.
- Evaluation.



### CONNAISSANCES ABORDEES

Architecture client-serveur.

### PROBLEMATIQUE

Réaliser un site web pour mobile ou tablette, soigné et fonctionnel. L'installer sur un serveur http et effectuer les réglages permettant de le rendre accessible à un équipement Wifi.

### CONDITIONS DE DEROULEMENT DE L'ACTIVITE

Phases de travail	Objectifs	Activités
<b>A) Mise en situation</b>	Comprendre le cahier des charges.	Lecture de la mise en situation.
<b>B) Etude de la problématique</b>	Concevoir les pages du site.	Ecriture du code HTML des pages du site. Test du site avec un émulateur pour mobile.
<b>C) Synthèse</b>	Modifier le site existant et le rendre accessible à un équipement disposant du wifi.	Modification du site. Réglage du serveur et d'un équipement disposant du wifi.

### SIGLES UTILISES

: Durée conseillée	: Important	: Information	: Editeur de texte (voir prof)
--------------------	-------------	---------------	--------------------------------

### RESSOURCES DOCUMENTAIRES, LOGICIELS ET MATERIELS

PC + Emulateur Opéra Mobile + Navigateur (Chrome, Opéra, Mozilla etc..) + Editeur de texte  
Serveur Apache - Guide des balises HTML

Le site partiellement développé dans ce tutoriel est accessible à l'adresse :

<http://webgepem.byethost14.com>



## Table des matières

<b>1. Mise en situation</b>	<b>3</b>
1.1. Introduction	3
1.2. Cahier des charges	3
<b>2. Etude de la problématique</b>	<b>4</b>
2.1. Code minimum du " template" des pages d'un site pour mobile	4
2.2. Affichage du template des pages du site Web BD dans l'émulateur Opéra pour mobile	5
2.3. Analyse du fichier <i>template.html</i>	6
2.4. Construction du site	7
2.4.1. Préparation	7
2.4.2. Etape 1 : codage de la page « Accueil » (fichier <i>index.html</i> )	7
2.4.3. Etape 2 : codage de la page « Biographie » (fichier <i>biographie.html</i> )	8
2.4.4. Etape 3 : codage de la page « Œuvre » (fichier <i>oeuvre.html</i> )	8
2.4.5. Etape 4 : codage de la page « Les aventures de Tintin (1929) » (fichier <i>aventures.html</i> )	9
2.4.6. Etape 5 : codage de la page « Musée » (fichier <i>musee.html</i> )	10
2.4.7. Etape 6 : codage de la page « Contact » (fichier <i>contact.html</i> )	10
<b>3. Synthèse</b>	<b>11</b>
3.1. Analyse du site WebBD2 = WebBD + une visionneuse d'images	11
3.2. Installation du site WebBD2 en réseau	11
3.3. QCM (Voir prof)	11
<b>Annexe 1 : Gabarit (template) des pages du site à réaliser</b>	<b>12</b>
<b>Annexe 2 : Rendu attendu pour chacune des pages</b>	<b>13</b>
<b>Pour aller plus loin</b>	<b>14</b>
<b>Glossaire</b>	<b>15</b>
<b>Bibliographie - Webographie</b>	<b>15</b>
<b>Tutoriel</b>	<b>15</b>
<b>Table des illustrations</b>	<b>16</b>

## 1. Mise en situation

[🕒 10mn]

### 1.1. Introduction

« Impossible aujourd'hui de négliger l'internet mobile lors de la création d'un outil de communication ! Sa maîtrise est une plus-value, aussi bien pour se vendre que pour vendre ». Webdesign (2012) - Extrait.

Réaliser une application web pour mobile, c'est-à-dire : une application capable d'accéder à des pages web à partir d'un **écran tactile**, dont les dimensions peuvent varier dans de grandes proportions d'un type d'appareil à l'autre, serait un travail complexe sans l'existence de **kits de composants logiciels (frameworks)** intégrant des fonctionnalités écrites en **javascript**<sup>1</sup> et des styles de présentation **CSS**. Les frameworks **HTML5** pour mobiles ont pour nom : Sencha Touch, JQTouch, **JQuery Mobile**,...

**jQuery Mobile** est un **Touch-Optimized Web framework** permettant aux développeurs de réaliser simplement des **applications web attrayantes** pour mobiles (Smartphone, tablette). Il est basé sur JQuery et JQuery UI<sup>2</sup> (deux outils de référence pour le développement d'un site web).



A en croire certains sites spécialisés, jQuery Mobile permet **d'aborder facilement** le développement web mobile !

### 1.2. Cahier des charges



Figure 1 : Page "Accueil"

- ❖ La page « **Accueil** » (fig1) contient **trois liens** :
  - un lien vers la biographie de l'auteur,
  - un lien vers la présentation de son œuvre,
  - un lien vers les informations permettant d'accéder au site du musée.
- ❖ La page « **Biographie** » (figure 8) doit être organisée en zones de **texte pliables (collapsibles)**.
- ❖ La page « **Œuvre** » (figure 11) doit afficher la chronologie de l'œuvre de l'auteur dans des **listes (listview)**. La liste des albums de Tintin « **Les aventures de Tintin (1929)** » (fig12) doit apparaître sur une nouvelle page.
- ❖ La page « **Musée** » (figure 13) doit présenter des informations générales et géographiques dans des zones de texte pliables (**collapsibles**). Elle contient **deux liens** :
  - un lien vers le site du musée consacré à l'auteur,
  - un lien vers les moyens d'accès au musée (plan, mail).
- ❖ La page « **Contact** » (figure 14) contient **deux liens** :
  - un lien vers le plan d'accès au musée (**Google Maps**),
  - un lien permettant d'envoyer un mail au musée.

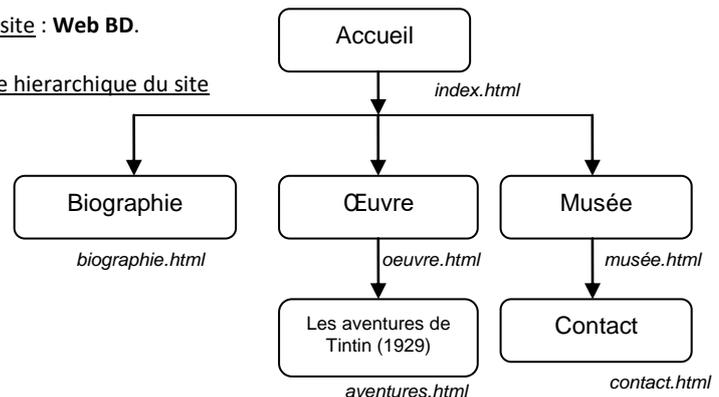


Figure 2: Géolocalisation

Le site à réaliser est destiné à des appareils mobiles (smartphones, tablettes). Il doit permettre de présenter rapidement l'auteur de bandes dessinées «Hergé» (sa jeunesse, son œuvre, la liste des albums de Tintin etc.). Il doit également permettre de géo localiser le musée qui lui est consacré.

Nom du site : **Web BD**.

Structure hiérarchique du site



Le rendu attendu pour les différentes pages est présenté en **annexe 2** à la fin de ce document.

<sup>1</sup> Javascript est le langage de programmation de scripts utilisé dans les pages web interactives (côté client). PHP est le langage de programmation de scripts utilisé côté serveur.

<sup>2</sup> UI : User interface (interface utilisateur).

Ce document va vous permettre de revoir les notions sur les langages **HTML** et **CSS** abordées précédemment (**balise, attribut, sélecteur, classe**). Vous appliquerez ces notions à la réalisation d'un **site web pour mobile**. Celui-ci sera déployé "en local" (votre PC) sur un **serveur Apache**. Il sera testé avec un **émulateur** de site pour mobile, un smartphone ou une tablette.

Aucune notion de programmation en Javascript n'est nécessaire dans ce TP.

Notre objectif étant de réaliser un site **soigné et fonctionnel**, les animations et le rendu des pages seront obtenus en faisant appel au **framework (jQuery Mobile)**.

Les **compétences** que vous allez acquérir seront réinvesties dans **les projets**. Elles vous permettront également de commencer à développer un **site mobile personnel**.

## 2. Etude de la problématique

[🕒 2h15]

*Objectif : Concevoir les pages du site.*

### 2.1. Code minimum du " template" des pages d'un site pour mobile

Dans cette première partie, vous allez étudier l'organisation de la **page de référence** du site puis vous serez guidé dans sa construction étape par étape.

Le site devant être constitué de **plusieurs pages**, nous allons utiliser un **gabarit (template)**. Il suffira de le dupliquer et de le compléter pour créer une page du site.

Le **template** le plus élémentaire, retenu pour les pages d'un site pour mobile, correspond au code ci-dessous :

Gabarit (template) disponible sur le site « JQuery Mobile »

```

<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <link rel="stylesheet" href="http://code.jquery.com/mobile/[version]/jquery.mobile-[version].min.css" />
  <script src="http://code.jquery.com/jquery-[version].min.js"></script>
  <script src="http://code.jquery.com/mobile/[version]/jquery.mobile-[version].min.js"></script>
</head>
<body>
  <div data-role="page">
    <div data-role="header">
      <h1>My Title</h1>
    </div><!-- /header -->

    <div data-role="content">
      <p>Page content goes here.</p>
    </div><!-- /content -->

    <div data-role="footer">
      <h1>Page Footer</h1>
    </div><!-- /footer -->
  </div><!-- /page -->
</body>
</html>

```

Template = Gabarit

En-tête de la page

Corps de la page

<meta indispensable pour l'affichage dans l'émulateur Opéra

L'**organisation** de cette page html est plutôt **classique**. Elle se compose d'un **en-tête** <head> et d'un **corps** <body> comme celles que vous avez écrites précédemment.

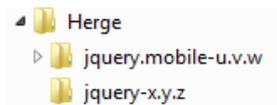
Outre la balise <title> que vous connaissez, l'en-tête contient une balise <meta>, une balise <link> et deux balises <script>.

- Sans trop entrer dans les détails, nous dirons que la **balise <meta>** donne des indications au navigateur. Dans le cas présent cette balise s'appelle « *viewport* ». Elle précise que la page web est optimisée pour les appareils mobiles ; l'attribut « *width* » affecté de la valeur « *device-width* » indique une largeur d'écran d'environ 320 pixels ; l'attribut « *initial-scale* » contrôle le zoom.
- La **balise <link>** permet de lier une feuille de style (.css), spécialement développée pour les appareils mobiles, à la page web.
- Les **balises <script>** permettent aux objets de la page d'avoir accès aux fonctions Javascript des bibliothèques JQuery et JQuery Mobile.

**Remarque :** Dans le code ci-dessus, ces balises font référence à des fichiers situés sur le site du projet JQuery. On dispose ainsi de la version actualisée et compressée des fichiers. Cette technique est principalement utilisée "en production".

Lors du **développement d'un site** il est préférable de télécharger les fichiers .css et .js des frameworks à partir du site JQuery Mobile pour les installer "**en local**" (sur son PC).

**C'est ce qui a été réalisé pour ce TP.** Le répertoire du projet « **Herge** » regroupe ces fichiers dans deux sous-répertoires comme sur la copie d'écran ci-contre. (Les chiffres correspondant à u, v, w et x, y, z représentent la version des frameworks)



Le fichier « *template.html* », situé dans le répertoire « **Herge** », contient le code du gabarit des pages du projet. Il a été construit à partir du code ci-dessus. L'en-tête a été modifié afin d'accéder aux fichiers des frameworks « **en local** ». Des éléments ont été rajoutés pour répondre à notre cahier des charges. Le code du template est disponible en **annexe 1** à la fin de ce document.



Le DOCTYPE précise au navigateur la **version** du code **HTML** utilisé. Son écriture, complexe avant , se limite aujourd'hui à : `<!DOCTYPE html>` sur chacune des pages du site.

*Appel prof*

Pour télécharger le répertoire du TP



Ouvrez le fichier « *template.html* », situé dans le répertoire « **Herge** » avec un éditeur de texte.

Affichez « *template.html* » dans un navigateur. **Attention** : certains navigateurs nécessitent d'avoir installé le projet sur un serveur !

Le navigateur du PC permet de vérifier rapidement le comportement du code mais l'affichage n'est pas représentatif du rendu sur un Smartphone. Pour cela, nous avons besoin d'un **émulateur**.

*Emulateur*

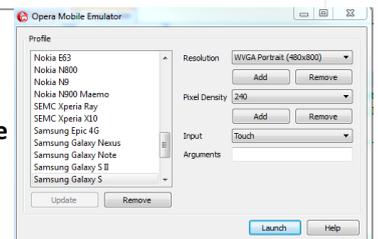


« En informatique, l'**émulation** consiste à substituer un élément de matériel – tel un terminal, un ordinateur, un mobile ou une console de jeux – par un logiciel ». Wikipédia.

Vous allez utiliser l'émulateur "**Opéra pour mobile**".

## 2.2. Affichage du template des pages du site Web BD dans l'émulateur Opéra pour mobile

- Démarez « *Opéra Mobile Emulator* » en double-cliquant sur l'icône placée sur le bureau du PC.
- Sélectionnez Samsung Galaxy S et cliquez sur « Launch » (Lancer).
- Copiez l'URL permettant d'accéder au template (*page* ouverte précédemment dans le navigateur du PC) et collez-la dans la **barre de recherche** de l'émulateur. Validez avec la touche `Entrée`.



L'émulateur doit afficher la page ci-contre.

Le **design par défaut** n'est pas exceptionnel mais à quoi ressemblerait cette page **sans** les styles CSS et les scripts du framework JQuery Mobile !



Figure 5: Template sans jQuery

Pour le savoir, il suffit de placer les balises `<meta>`, `<link>`, et `<script>` à l'intérieur de balises de commentaires.

- Entrez `<!--` avant `<link` et `-->` après le dernier `</script>`.

- **Sauvegardez** le fichier (`ctrl-s`) et actualisez l'émulateur (clic sur l'icône  ou touche F5 après l'avoir sélectionné avec la souris).



Figure 4 : Template avec jQuery

Vous devriez obtenir une page ressemblant à la copie d'écran ci-dessus (figure 5). Pas terrible !

On comprend mieux la devise du projet JQuery : « **write less, do more** ».

**En effet**, avec JQuery, nous avons un « **Design** » sans écrire une ligne de code CSS !

**Supprimer** les balises `<!-- -->` écrites précédemment et **sauvegardez** la page.

**Remarque** : Dans ce TP, nous utiliserons le « Design par défaut ». Sa personnalisation avec un outil graphique est présentée dans le pour "aller plus loin".



Figure 6 : Style personnalisé

### 2.3. Analyse du fichier *template.html*

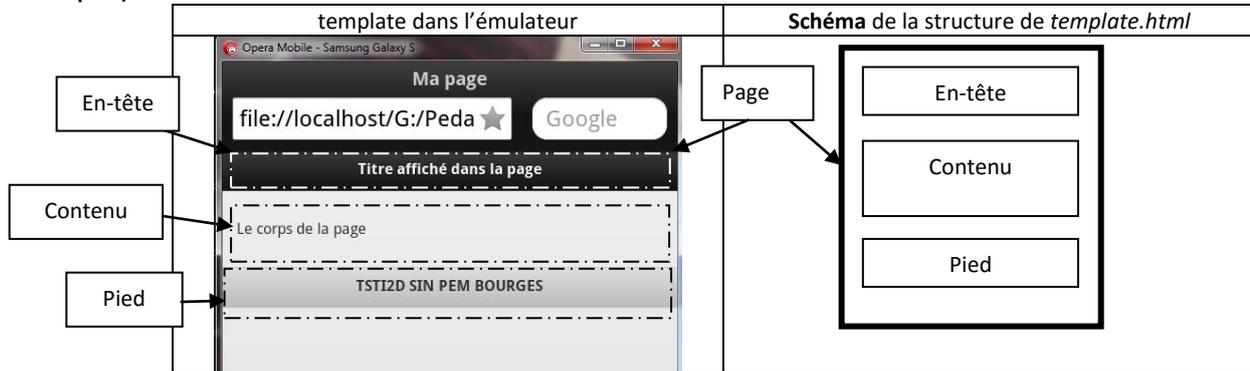
En observant le rendu de la page *template.html* dans l'émulateur, nous pouvons identifier **trois zones** :

- ❖ Un en-tête de page,
- ❖ Un contenu
- ❖ Un pied de page

Ces zones étant destinées à **contenir des objets** (boutons, texte etc.) on les appelle **boîte (ou conteneur)**.

La page *template.html* peut être représentée sous la forme d'un **schéma comprenant quatre boîtes (page, en-tête, contenu et pied)**.

Conteneur  
=  
boîte



**<div>  
data-\***

Le fichier *template.html* contient les **balises <div>** complétées par des attributs **data-\*** (ou \* est remplacé par un mot du framework JQuery Mobile).

Exemple : `<div data-role="page"> </div>` <!--Correspond à la définition de la boîte (Page) en trait gras ci-dessus -->

Les balises **<div>** créent les **différentes boîtes** (page, en-tête, contenu et pied). En analysant le contenu de la balise `<body>` dans le fichier *template.html*, on constate qu'une balise `<div>`, à laquelle on attribue le rôle de "page", **contient** trois autres `<div>` dont le rôle est **header** (en-tête de page), **content** (contenu) et **footer** (pied de page).



**<div>** : balise spécialement conçue pour **introduire une division (boîte ou conteneur sans passer par une balise de paragraphe)**. Cette balise est particulièrement utile pour la déclaration des feuilles de style.

**data-\*** : Cet attribut fait partie des « custom data attributs » introduits par **HTML5**. Ils permettent d'ajouter une donnée ou une information sur l'élément HTML sans qu'elle soit visible par l'utilisateur. Ce sont ces données que JQuery Mobile utilise pour transformer la page web en une application mobile.

Les termes "role", "header", "content" et "footer" sont reconnus par JQuery Mobile pour appliquer des styles à la page. Vous pouvez faire beaucoup avec JQuery Mobile : ajouter des listes, mettre des boutons, des formulaires, etc. Tout passe par les attributs **data-\***. Leur description est disponible sur le site de [JQuery Mobile](#).

#### Résumé et rappels

La réalisation du site va être grandement facilitée par l'utilisation des frameworks **JQuery** et **JQuery Mobile**. Les différents **comportements et rendus** attendus seront obtenus sans écrire une ligne de code **Javascript** puisqu'il suffira de "décorer" les balises `<div>` (division appelées blocs ou **conteneur**), assurant la structure de la page, avec des attributs **data-\***.

Seul le concept d'attribut, vu dans les précédents TP doit avoir été bien compris !

#### Le bon usage des balises (règles définies en xHTML et reprises par )

- **Règle 1** : Ecrire les balises en **minuscule**.
- **Règle 2** : Toutes les balises **ouvertes** par `<nom_balise>` doivent être **fermées** par `</nom_balise>` sauf s'il s'agit d'une balise unique `<balise_unique />`.
- **Règle 3** : Les balises doivent être correctement **imbriquées et indentées**.

Exemple : Dans le code ci-dessous, le `<a></a>` est **imbriquée** avec `<p></p>`, `<div>` et `<p>` sont **indentées**.

```

<div
  <p><a href="page2.html" data-role="button">Oeuvre</a></p>
</div>

```

- **Règle 4** : Les valeurs des attributs doivent toujours figurer entre des **guillemets**.

Exemple : `<a href="bibliographie.html" data-role="button">`

**Il est temps de commencer à coder...**

## 2.4. Construction du site

### 2.4.1. Préparation

Pour des raisons de sécurité, les navigateurs n'exécutent pas les liens gérés par JQuery si les pages qui les contiennent ne se situent pas sur un serveur. Aussi, comme nous allons travailler en "local", le répertoire du site "Herge" doit être copié dans le répertoire **www** du serveur Apache installé sur le PC.



1. Copiez le répertoire « **Herge** » dans le répertoire **www** du serveur (chemin **c:/wamp/www/**)
2. Démarrez le serveur HTTP en cliquant sur l'icône  ou à partir de la liste des programmes : Démarrer → Tous les programmes → Wampserver.



- 3 Dans l'éditeur, fermez le fichier « **template.html** », et ouvrez « **index.html** » situé dans le répertoire « **Herge** », copié sur le serveur.  
(⚠ **c:/wamp/www/Herge**)

Après le démarrage du serveur, un clic sur l'icône  dans la zone de notification doit vous donner la figure ci-contre.



- 4 Entrez **127.0.0.1/Herge** dans la barre de recherche de l'émulateur Opéra Mobile pour accéder à la page **index.html** du site.



Pour ne pas avoir à refaire cette manipulation lors de chaque accès à « Accueil », vous allez créer un **signet (raccourci)** dans l'émulateur Opera. (1) Clic sur 



Format agrandi disponible en annexe 2.



Figure 7 : Page "Accueil"

### 2.4.2. Etape 1 : codage de la page « Accueil » (fichier **index.html**)

Pour l'instant, le contenu de "**index.html**" est identique à celui de "**template.html**".

Vous allez modifier ce fichier afin d'obtenir l'affichage ci-contre.



**Revoyez** la description de la page « Accueil » dans le **cahier des charges**.

#### Indications

- La **taille du texte** se règle avec des balises **<hx></hx>** ( $x \in [1,6] \Rightarrow$  la valeur de  $x$  règle la taille du texte). Les paragraphes sont obtenus en utilisant des balises **<p></p>**.
- Pour transformer un lien en bouton, on utilise **data-role="button"** ou **class="ui-btn ui-corner-all ui-shadow"**.

Ex : `<a href="biographie.html" data-role="button">Biographie</a>`

lien vers la page « Biographie » décorée avec un aspect « bouton »

*Appel prof* Pour faire vérifier la justesse du code



**Activité 1** : Modifiez le fichier **index.html** pour que la page « Accueil » corresponde à la copie d'écran ci-dessus (figure 7). Testez-le dans l'émulateur Opéra Mobile. **Imprimez** le code source de la page et joignez-le à votre document réponse.

### 2.4.3. Etape 2 : codage de la page « Biographie » (fichier *biographie.html*)

La page « Biographie » présente **deux nouveautés** par rapport à la page « Accueil » :

- une icône  (home), située dans l'en-tête (un lien vers *index.html*), permet à l'utilisateur de revenir sur la page « Accueil »,
- trois zones de **texte pliables (collapsible)** optimisent la place occupée sur l'écran.



Ouvrez le fichier « *template.html* » dans l'éditeur ( `c:/wamp/www/Herge`), et **sauvegardez-le** sous le nom « *biographie.html* »



Figure 8 : Page "Biographie"

#### Indications

- Pour placer l'icône , il suffit d'entrer le code ci-dessous :

```
<a href="index.html" data-theme="d" data-icon="home" data-
iconpos="notext">home</a>
```

- Pour créer la première zone de **texte pliable**, il suffit d'entrer le code ci-dessous :

```
<div data-role="collapsible" data-content-theme="b">
  <h3>Une jeunesse bruxéloise</h3>
  <p>< Je me sentais médiocre et je vois ma jeunesse comme une
chose grise, grise.> Hergé </p>
</div><!-- /collapsible -->
```

**Activité 2 :** Modifiez le fichier *biographie.html* pour que la page « Biographie » corresponde à la **figure 8**. Testez-le dans l'émulateur Opéra Mobile. **Imprimez** le code source de la page et joignez-le à votre document réponse.

*Appel prof*

Pour faire vérifier la justesse du code

La **documentation** du framework est accessible sur le site **JQuery Mobile**. (<http://jquerymobile.com/>)

Sélectionnez « **Demo** » puis la dernière version stable.

En consultant la page associée au widget « Collapsible » vous devez retrouver le code proposé dans les indications ci-dessus.



Figure 9: Aide en ligne jQuery mobile

*Documentation*

 La documentation en ligne du framework vous donne toutes les informations utiles au codage de vos pages !

### 2.4.4. Etape 3 : codage de la page « Œuvre » (fichier *oeuvre.html*)

La page « Œuvre » est composée de **listes à puces**. La décoration de ces listes par le framework jQuery produit l'affichage ci-contre (fig10).

```
<ul>
<li>
```

Dans un premier temps, vous allez découvrir le comportement des listes à puce, et voir deux nouvelles balises : **<ul>** et **<li>**.

Vous les utiliserez ensuite avec JQuery Mobile pour coder la page « Œuvre ».

*Liste à puces*

	<p>Il est souvent nécessaire de <b>mettre en forme des listes d'éléments</b>. La <b>liste à puce</b> permet de faire une énumération.</p> <p>Deux balises sont nécessaires pour obtenir une liste à puce <code>&lt;ul&gt;&lt;/ul&gt;</code> et <code>&lt;li&gt;&lt;/li&gt;</code>. <code>&lt;ul&gt;</code> ouvre la liste et <code>&lt;/ul&gt;</code> la ferme.</p> <p>Chaque élément de la liste doit être encadré par <code>&lt;li&gt; &lt;/li&gt;</code></p>
---	---

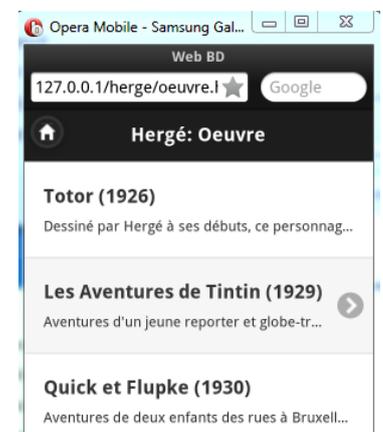
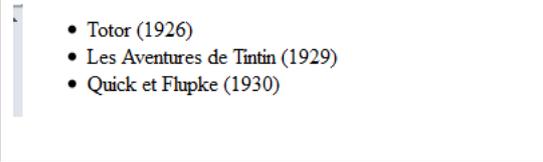


Figure 10 : Page "Œuvre" partielle

**Exemple** : La page *oeuvre.html* va être présentée sous la forme d'une **liste à puce** comme ci-dessous.

Code HTML	Résultat dans un navigateur <u>sans</u> jQuery
<pre>&lt;ul&gt;   &lt;li&gt;Totor (1926)&lt;/li&gt;   &lt;li&gt;Les Aventures de Tintin (1929)&lt;/li&gt;   &lt;li&gt;Quick et Flupke (1930)&lt;/li&gt; &lt;/ul&gt;</pre>	

Avec les balises HTML seules, le rendu est "minimum" ! Là encore, le framework JQuery Mobile assure un certain « **Design** » à la page.



Ouvrez le fichier « *template.html* » dans l'éditeur (⚠ *c:/wamp/www/Herge*), et **sauvegardez-le** sous le nom « *oeuvre.html* »

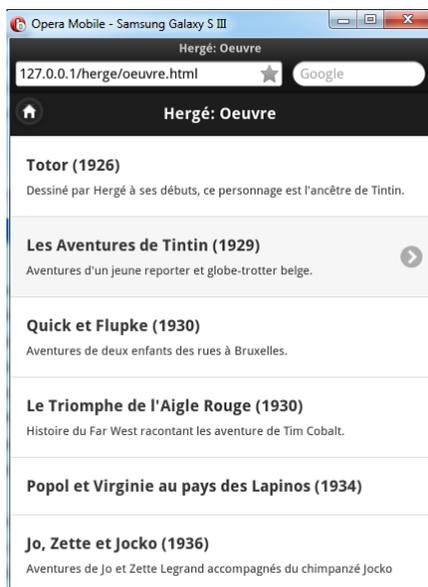


Figure 11: Page "Œuvre"

#### Indications

Pour présenter le premier élément sous la forme d'une liste, il suffit d'entrer le code ci-dessous :

```
<ul data-role="listview">
  <li>
    <h3>Totor (1926)</h3>
    <p>Dessiné par Hergé à ses débuts, ce personnage est
    l'ancêtre de Tintin.</p>
  </li>
</ul>
```

**Remarque** : Le deuxième élément doit contenir un lien vers la page *aventure.html*. Le bouton  est automatiquement créé. Consultez la documentation en ligne pour plus d'informations.

**Activité 3** : Modifiez le fichier "*oeuvre.html*" pour que la page « Œuvre » corresponde à la figure 11. **Testez-le** dans l'émulateur Opéra Mobile. **Imprimez** le code source de la page et joignez-le à votre document réponse. N'oubliez pas le lien  vers la page « Accueil » (fichier *index.html*).

#### 2.4.5. Etape 4 : codage de la page « Les aventures de Tintin (1929) » (fichier *aventures.html*)



Ouvrez le fichier « *template.html* » dans l'éditeur (⚠ *c:/wamp/www/Herge*), et **sauvegardez-le** sous le nom « *aventures.html* ».



Figure 12: Page "Les aventures de Tintin (1929) "

La réalisation de la page « Les aventures de Tintin (1929) » nécessite de **reprendre la démarche adoptée pour une partie du travail fait précédemment**.

**Remarque** : La conception de cette page nécessite d'avoir bien compris les activités précédentes !

**Activité 4** : Modifiez le fichier "*aventures.html*" pour que la page « Les aventures de Tintin (1929) » corresponde à la figure 12. **Testez-la** dans l'émulateur Opéra Mobile. **Imprimez** le code source de la page et joignez-le à votre dossier réponse.

*Appel prof*

Pour faire vérifier la justesse du code

#### 2.4.6. Etape 5 : codage de la page « Musée » (fichier *musee.html*)

La réalisation de la page « Musée » nécessite de **reprendre la démarche adoptée pour une partie du travail réalisé précédemment**. En effet, il s'agit maintenant de présenter des **listes à puces** à l'intérieur de zones pliables et d'assurer les liens vers le site du musée et vers la page « Contact » (fichier *contact.html*).



Ouvrez le fichier "*template.html*" dans l'éditeur (⚠ `c:/wamp/www/Herge`) et sauvegardez-le sous le nom "*musee.html*"

Liste à puce  
contenu dans  
une zone  
pliable

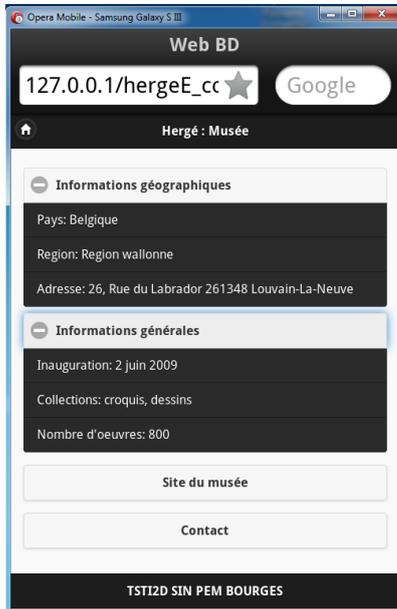


Figure 13: Page "Musée"



Revoyez la description de la page « Musée » dans le cahier des charges.

#### Indications

L'url du musée Hergé est : <http://www.museeherge.com>

Pour réaliser le lien vers la page contact, il faut entrer le code ci-dessous :

```
<a href="contact.html" data-role="button" data-rel="dialog" data-transition="pop">Contact</a>
```

**Activité 5 :** Modifiez le fichier "*musee.html*" pour que la page « Musée » corresponde à la figure 13. Testez-le dans Chrome !

N'oubliez pas le lien  vers la page « Accueil ». Imprimez le code source de la page et joignez-le à votre document réponse.

Appel prof

Pour faire vérifier la justesse du code.

#### 2.4.7. Etape 6 : codage de la page « Contact » (fichier *contact.html*)

Dans cette dernière étape, vous allez mettre en œuvre un **outil graphique "en ligne"**. Vous **comparerez** ensuite le code obtenu avec cet outil à celui que l'on aurait pu écrire « à la main ».

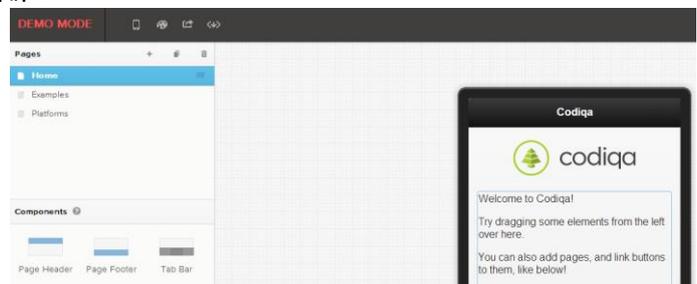
Cet outil est disponible sur le site **Codiqa**.

À entrer dans le navigateur

<https://codiqa.com/demo>

En mode **DEMO**, l'outil graphique se présente comme sur la copie d'écran ci-contre.

Si ce genre d'outil accélère le processus de création, il **ne supprime pas complètement la phase de codage !**



Vous allez générer le code de la page « Contact » (fichier *contact.html*) figure 14 ci-contre avec Codiqa.

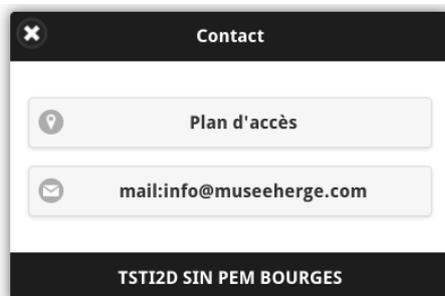


Figure 14: Page "Contact"

Pour cela :

- supprimez les objets présents dans l'émulateur à l'ouverture de la page,
- placez les différents éléments en effectuant des *glisser-déposer* et paramétrez les boîtes de code.

Vous obtiendrez le code HTML en cliquant sur le bouton **Code** en bas et à droite de la page.

#### Indications

- Liens pour le Plan d'accès : <http://goo.gl/maps/9IcSh>
- Le mail : `href="mailto:info@museeherge.com?subject=Demande d'informations"`



**Activité 6 :** Etablissez le code de la page « Contact » avec l'éditeur graphique **Codiqa** et **comparer-le** à celui du fichier contenu dans le répertoire « Herge » (fichier *contact.html*). Précisez si ce code est plus complet, moins complet, l'utilité de ce qui a été rajouté etc.). Répondez sur le document réponse.

### 3. Synthèse

[🕒 15mn]

**Objectif :** Rendre le site accessible à un équipement mobile disposant du wifi.

#### 3.1. Analyse du site WebBD2 = WebBD + une visionneuse d'images

Une visionneuse d'images a été intégrée à WebBD en utilisant la bibliothèque [PHOTOSWIPE](#).

**Copiez** le répertoire « Herge2 » (situé dans le répertoire du TP) dans le répertoire www du serveur (chemin `c:/wamp/www/`)



**Activité 7 :** Analysez le code de la page `index.html` et complétez l'arborescence de ce nouveau site sur le document réponse.

#### 3.2. Installation du site WebBD2 en réseau



Votre PC est relié à une "box" par le réseau du lycée. Il est donc accessible à un équipement disposant du Wifi.

*Appel prof*

Pour avoir le code d'accès au Wifi de la "box".



**Activité 8 :** Réglez les paramètres de votre PC (Serveur http) **et de** votre Smartphone (ou une tablette du lycée) pour accéder à votre site.

**Notez** les principaux réglages sur le document réponse.

#### 3.3. QCM (Voir prof)

## Annexe 1 : Gabarit (template) des pages du site à réaliser

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Web BD</title>
5   <meta name="viewport" content="width=device-width, initial-scale=1 /">
6
7   <!-- Liens pour le développement en local -->
8   <link rel="stylesheet" href="jquery.mobile-1.4.3/jquery.mobile-1.4.3.min.css" />
9   <script src="jquery-1.11.1/jquery-1.11.1.min.js"></script>
10  <script src="jquery.mobile-1.4.3/jquery.mobile-1.4.3.min.js"></script>
11
12  <!-- Liens avec une liaison internet -->
13  <!--<link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.3/jquery.mobile-1.4.3.min.css" />
14  <script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>
15  <script src="http://code.jquery.com/mobile/1.4.3/jquery.mobile-1.4.3.min.js"></script></script-->
16 </head>
17
18 <body>
19 <div data-role="page">
20 <div data-role="header" data-theme="b">
21   <h1>Titre affiché dans la page</h1>
22 </div><!-- /header -->
23
24 <div data-role="content">
25   <p>Le corps de la page</p>
26 </div><!-- /content -->
27
28 <div data-role="footer" data-theme="b">
29   <h1>TSTI2D SIN PEM BOURGES</h1>
30 </div><!-- /footer -->
31 </div><!-- /page -->
32 </body>
33 </html>
```

Annexe 2 : Rendu attendu pour chacune des pages



Figure 1 et Figure 7



Figure 8

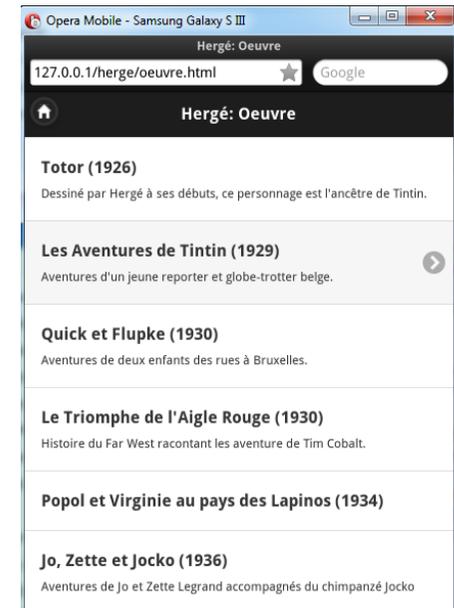


Figure 11

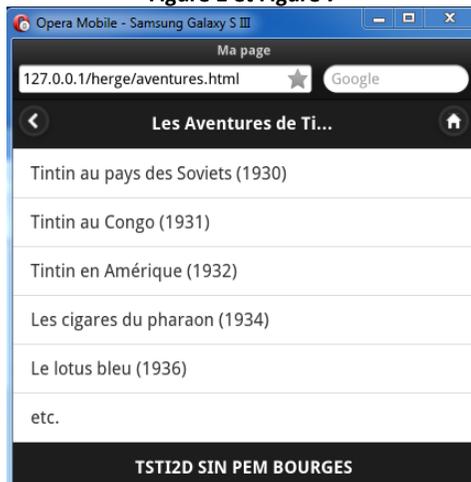


Figure 12



Figure 14

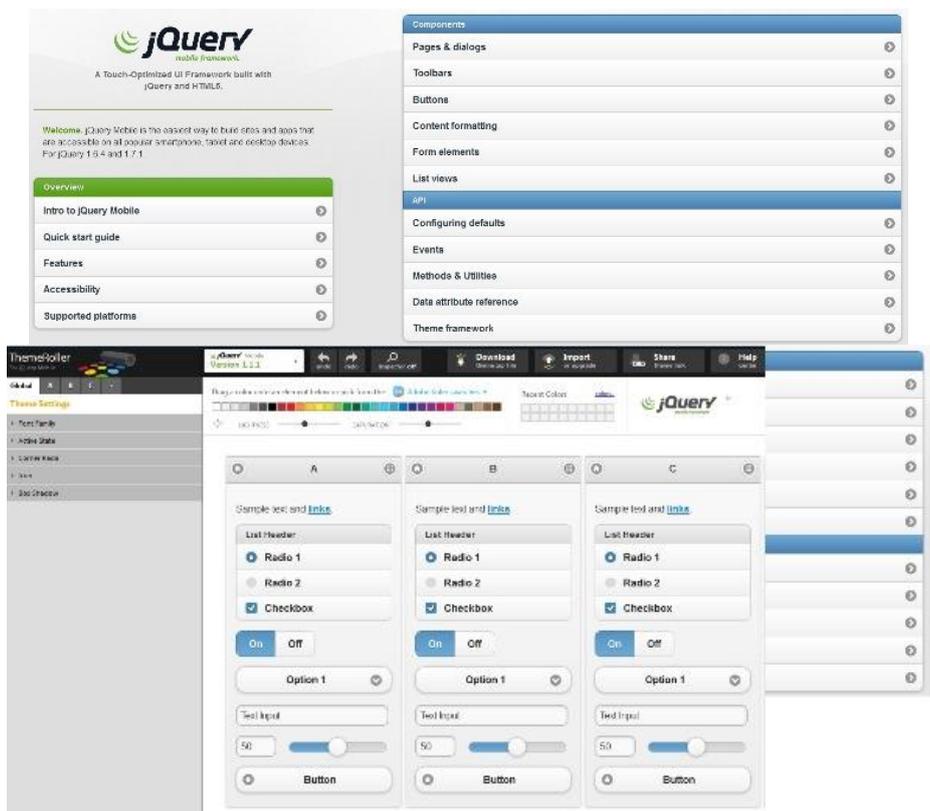


Pour aller plus loin....

### Informations complémentaires (pour améliorer et personnaliser le projet)

La documentation JQuery mobile en français est disponible à l'adresse :

<http://mobile.jquery-fr.com/>



Le générateur de styles JQuery mobile permet de personnaliser les différents thèmes d'un site mobile à l'aide d'une interface graphique.

<http://jquerymobile.com/themeroller/>

**PhotoSwipe** est une librairie JavaScript permettant de mettre en place **une galerie d'images**. Elle cible spécifiquement les terminaux mobiles, offre aux visiteurs une interface familière et intuitive, et utilise la technique du « **responsive design** » pour s'adapter automatiquement à la résolution d'écran.

<http://www.photoswipe.com/>



Informations sur PhotoSwipe (en français)

<http://goo.gl/v7dTu5>

## Glossaire

### Framework :

En programmation informatique, un **framework** est un *kit* de [composants logiciels](#)...(Wikipédia)

### Template

Un **gabarit**, souvent nommé à tort avec les anglicismes **template** ou **layout**, est un patron de mise en page où l'on place images et textes. Aussi, il est souvent utilisé de manière répétitive pour créer des documents présentant une même structure. (Wikipédia)

**UTF-8** (*UCS transformation format 8 bits*) est un [codage de caractères informatique](#) conçu pour coder l'ensemble des caractères internationaux d'[Unicode](#) en restant compatible avec la norme [ASCII](#) limitée à l'anglais, mais très largement répandue depuis des décennies. (Wikipédia)

## Bibliographie - Webographie

### Périodiques

WebDesign Masterclass Vol.6

Article : « Réaliser un site pour mobile »

WebDesign n°41 Juillet-Août 2012

Article : « Construire un diaporama en JQuery »

### Livres

JQuery Mobile (Créez des applications web mobiles)

John Reid

PEARSON

### Liens

Hergé

<https://goo.gl/dh7oiP>

Musée Hergé

<https://goo.gl/y0920N>

### Documentation en français

JQuery

<http://www.jquery-fr.com/>

JQuery mobile

<http://mobile.jquery-fr.com/demos/>

## Tutoriel



Un site Web dynamique avec JQuery : Le framework JQuery mobile

<https://goo.gl/TqzgwX>

## Table des illustrations

<i>Figure 1 : Page "Accueil"</i> .....	3
<i>Figure 2: Géolocalisation</i> .....	3
<i>Figure 3 : Choix de la cible</i> .....	5
<i>Figure 4 : Template avec jQuery</i> .....	5
<i>Figure 5: Template sans jQuery</i> .....	5
<i>Figure 6 : Style personnalisé</i> .....	5
<i>Figure 7 : Page "Accueil"</i> .....	7
<i>Figure 8 : Page "Biographie"</i> .....	8
<i>Figure 9: Aide en ligne jQuery mobile</i> .....	8
<i>Figure 10 : Page "Œuvre" partielle</i> .....	8
<i>Figure 11: Page "Œuvre"</i> .....	9
<i>Figure 12: Page "Les aventures de Tintin (1929) "</i> .....	9
<i>Figure 13: Page "Musée"</i> .....	10
<i>Figure 14: Page "Contact"</i> .....	10