

Prise en main de NVU et Notepad++

(conception d'application web avec PHP et MySQL)

Propriétés	Description
Intitulé long	Conception de pages web dynamiques à l'aide de NVU et Notepad++
Formation concernée	Classe terminale de la série Sciences et technologies de la gestion (STG) Spécialité « Gestion des systèmes d'information » (GSI)
Matière	Gestion des systèmes d'information
Présentation	Après une présentation succincte des spécificités de NVU, le lecteur est invité à suivre l'élaboration de pages HTML et PHP. La présentation des pages et l'accès aux données sont gérés de façon indépendante par des feuilles de style et des fonctions PHP
Notions	B1.1 Définition, interrogation et mise à jour des données B3.2 Architecture des applications C2. L'adaptation de l'application
Outils	NVU, Notepad++ (ou le bloc-note), MySQL, PhpMyAdmin.
Mots-clés	HTML, PHP, cas d'utilisation, maquettage, programmation
Pré-requis	connaissances en HTML, PHP et SQL
Lien(s)	Document présentant la programmation PHP (cas Lafleur de Pierre Loisel) : http://www.reseaucerta.org/cotecours/cotecours.php?num=95
Auteur(es)	Jean-Philippe Pujol
Version	1.0
Date de publication	octobre 2006

Ce document présente succinctement NVU, recommandé par le réseau Certa (<http://www.reseaucerta.org/outils/internet.php>), puis décrit son utilisation conjointement avec Notepad++ pour l'élaboration d'un petit site web dynamique : quelques pages publiées sur un site http accèdent à des informations stockées dans une base de données

Nvu est un éditeur de pages web What-You-See-Is-What-You-Get (WYSIWYG) (Ce que vous voyez est ce que vous aurez). Il est basé sur Gecko, le moteur de rendu de Mozilla. Nvu est totalement libre et peut-être téléchargé sur le site NVU : <http://www.nvu.com>

Notepad++ est un outil alternatif au bloc-note qui fait de la coloration syntaxique ; il fait l'objet d'une licence GNU General Public License et est publié par la Free Software Foundation. Il peut être téléchargé ici : <http://notepad-plus.sourceforge.net>

L'installation de ces deux outils se fait simplement en exécutant le fichier téléchargé.

NVU permet de construire une page en mode graphique dans un onglet, tandis que son code peut être observé ou modifié dans un autre onglet.

Plutôt orienté vers la réalisation de pages statiques, cet outil est ici mis en œuvre pour réaliser une petite application qui puise ses données dans une base MySQL à l'aide d'instructions PHP.

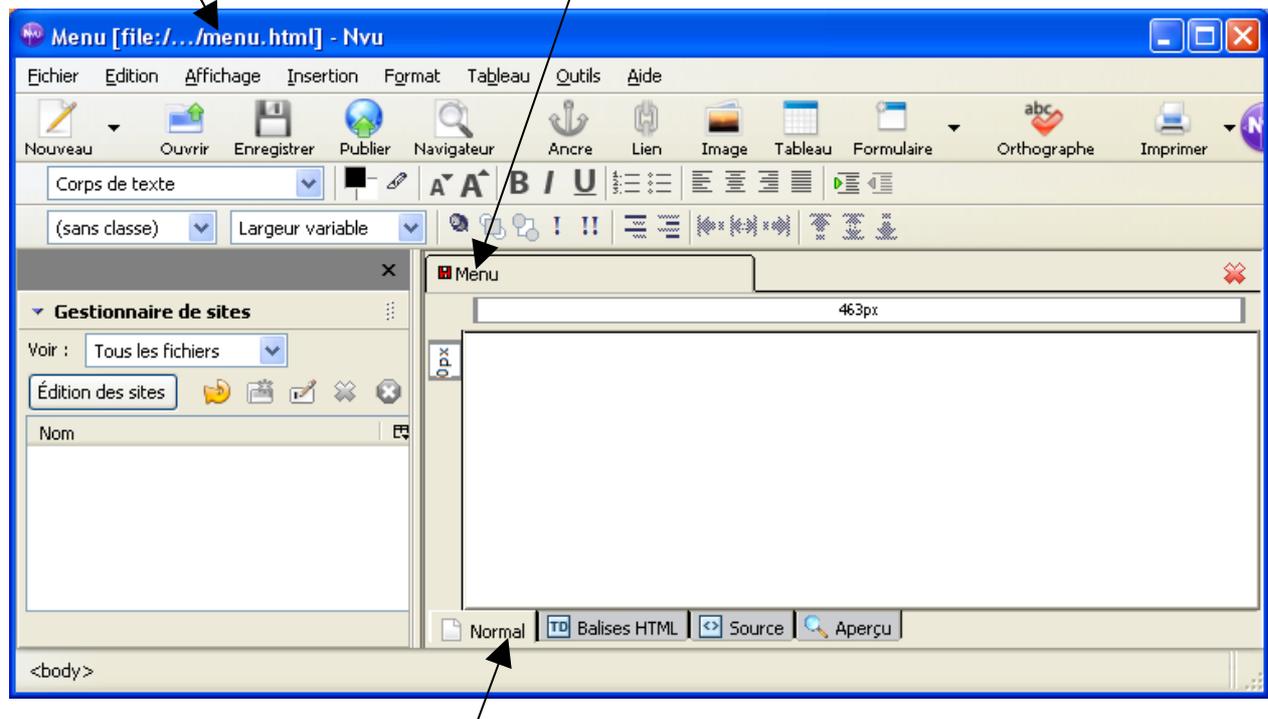
La technique de séparation en couches utilisée ici isole l'interface utilisateur des données ; cette méthode fait l'objet d'une utilisation conjointe de NVU et de Notepad++

Présentation de NVU

Les différentes icônes et éléments du menu permettent de composer la page web.

Il est indispensable d'enregistrer la page dès le début du travail car certaines fonctionnalités ne sont accessibles qu'à cette condition.

Le nom du fichier enregistré apparaît ici et la valeur de l'attribut <title> ici



Les onglets permettent d'observer la page sous ses différents aspects :

- « Normal » : mode de conception graphique où les balises invisibles sont schématisées par des icônes ;
- « Balises HTML » : aperçu graphique des balises HTML et de leurs imbrications. Bien utile pour observer la structure de tableaux complexes ;
- « Source » : le code HTML pur qui peut éventuellement être modifié. Utile parfois pour positionner le pointeur d'écriture à un emplacement précis ;
- « Aperçu » : le rendu de la page (en mode statique).

La fenêtre de gauche permet la gestion de la publication des documents sur un site dont on fournit les paramètres. La rubrique d'aide explique dans le détail ce fonctionnement.

Le menu « Insertion » permet entre autres d'insérer du code PHP qui sera automatiquement placé entre les balises spécifiques <?php ... ?>

Dans le menu « Outils », on trouve un éditeur de feuille de styles css ainsi qu'une console javascript.

Un avantage -ou un inconvénient- de NVU est son obstination à régénérer automatiquement les balises HTML comme <body> ou <head>. Bien adaptée à la conception de pages statiques, cette méthode devient gênante en développement PHP.

Il est dès lors obligatoire de passer par un éditeur de texte (Notepad++ par exemple) pour pouvoir effacer les balises inutiles d'un document qui doit être inclus dans un autre par un appel PHP spécifique.

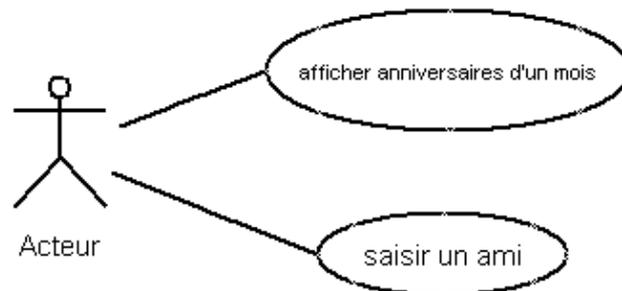
Exemple de mise en œuvre

Après avoir décrit le contexte servant d'exemple, nous réaliserons dans l'ordre les opérations suivantes :

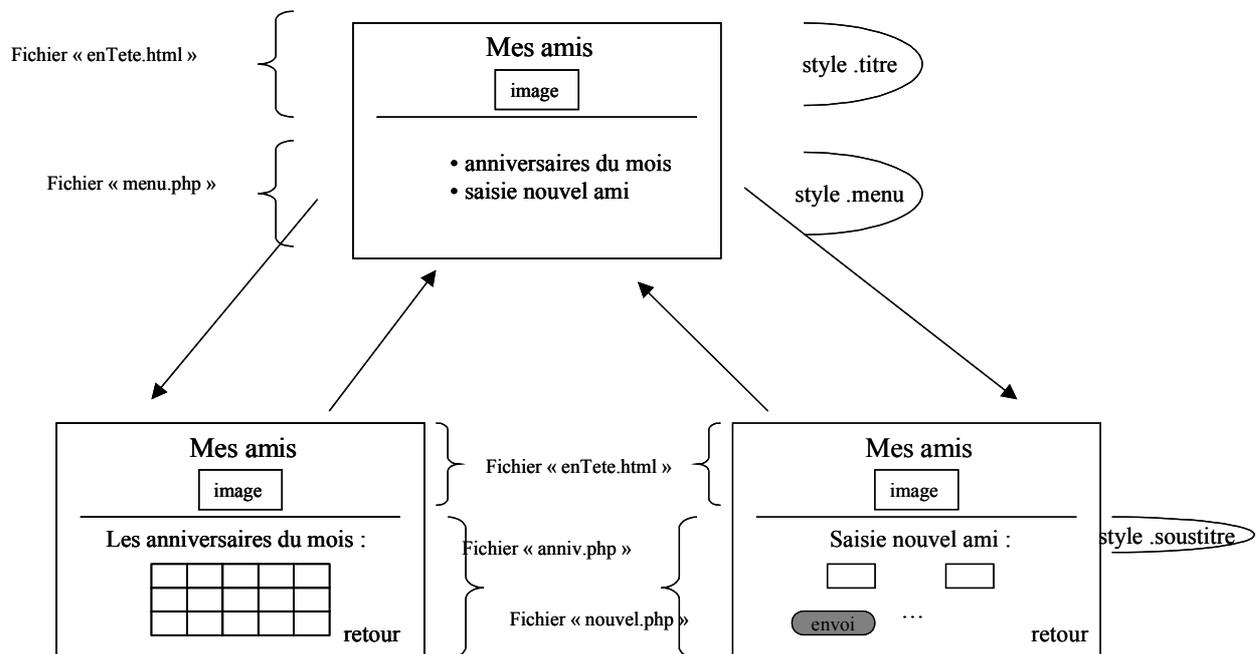
- maquettes des pages web ;
- base de données nécessaire ;
- conception des requêtes d'accès SQL ;
- création de l'en-tête de chaque page ;
- conception de la feuille de styles externe ;
- création des interfaces et programmes PHP.

Contexte

On souhaite assurer une gestion simplifiée de ses amis. On se limite à deux cas utilisation :



Maquettes de l'application



L'application est composée de trois interfaces web.

Chaque page comporte un même en-tête dont le code est enregistré dans un fichier *enTete.html*. Trois autres fichiers permettent la réalisation du corps des pages¹ : *menu.php*, *anniv.php* et *nouvel.php*. Un fichier *enregAmi.php* assurera l'enregistrement dans la base des données saisies.

Trois styles nommés sont prévus : *.titre*, *.menu* et *.soustitre*.

Les balises `<body>`, `<table>` et `<td>` bénéficieront aussi d'un style personnalisé.

¹ l'extension *.php* est requise pour ces pages qui contiennent du code php

Base de données

On crée une base de données « CARNET » comportant une seule table « Ami » dont le schéma relationnel est :

Ami (id, nom, prenom, dateNaissance, telephone, adresseCourriel)
clé primaire : *id*
(la clé *id* est un numéro automatique géré par MySql)

Cette base de données est créée avec PhpMyAdmin dans MySql à l'aide du script (fourni) « *scriptAmi.txt* »

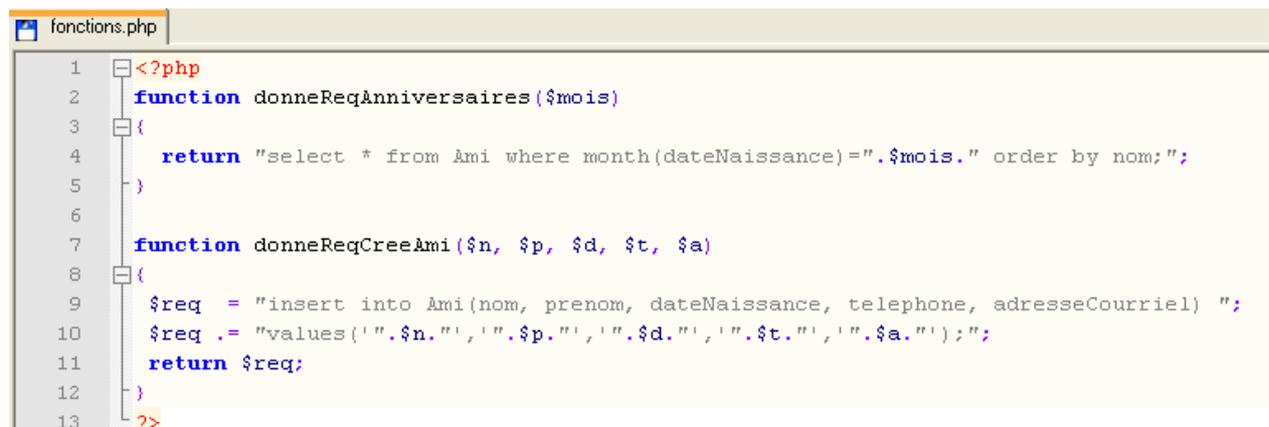
Espace de travail

Les fichiers HTML, PHP, CSS, JPG... nécessaires à l'application web à construire vont être stockés dans un espace (un répertoire) qu'on crée préalablement sur son disque.

Requêtes SQL d'accès aux données

Afin d'assurer une séparation entre les interfaces et l'accès aux données, les requêtes SQL nécessaires aux traitements envisagés vont être enregistrées dans un fichier spécifique. Elles seront accessibles via des fonctions PHP.

À l'aide de NotePad++ on crée le fichier *fonctions.php* sauvegardé dans le répertoire de travail :



```
1 <?php
2 function donneReqAnniversaires($mois)
3 {
4     return "select * from Ami where month(dateNaissance)='".$mois.'" order by nom;";
5 }
6
7 function donneReqCreeAmi($n, $p, $d, $t, $a)
8 {
9     $req = "insert into Ami(nom, prenom, dateNaissance, telephone, adresseCourriel) ";
10    $req .= "values ('".$n."', '".$p."', '".$d."', '".$t."', '".$a."')";
11    return $req;
12 }
13 ?>
```

La fonction *donneReqAnniversaire()* attend un numéro de mois passé en paramètre. Dans cette version simplifiée, le numéro de mois est extrait de la date courante Elle retourne la requête SQL nécessaire à la sélection.

La fonction *donneReqCreeAmi()* attend cinq paramètres servant à créer puis insérer un nouveau tuple dans la table (la clé est un numéro automatique).

Pour ne pas surcharger l'exemple, la connexion à la base de données n'est pas incluse dans ce fichier mais est traitée dans chaque page : c'est un inconvénient qui peut faire l'objet d'une évolution.

En-tête et feuille de styles

Avec NVU, une feuille de style est obligatoirement créée en association avec une page HTML déjà enregistrée.

On va donc d'abord créer la page d'en-tête puis la feuille de styles qui sera utilisée ensuite par toutes les autres pages.

Création de l'en-tête de chaque page

Avec NVU on crée un nouveau document² : « Fichier / nouveau / document vide ».

On enregistre immédiatement ce document vide dans l'espace de travail précédemment créé : « Fichier / enregistrer sous »

On fournit un titre (propriété <title>), par exemple « Mes Amis », puis le nom du fichier HTML. Par défaut, NVU propose le même nom ; on saisira ici *enTete.html*.

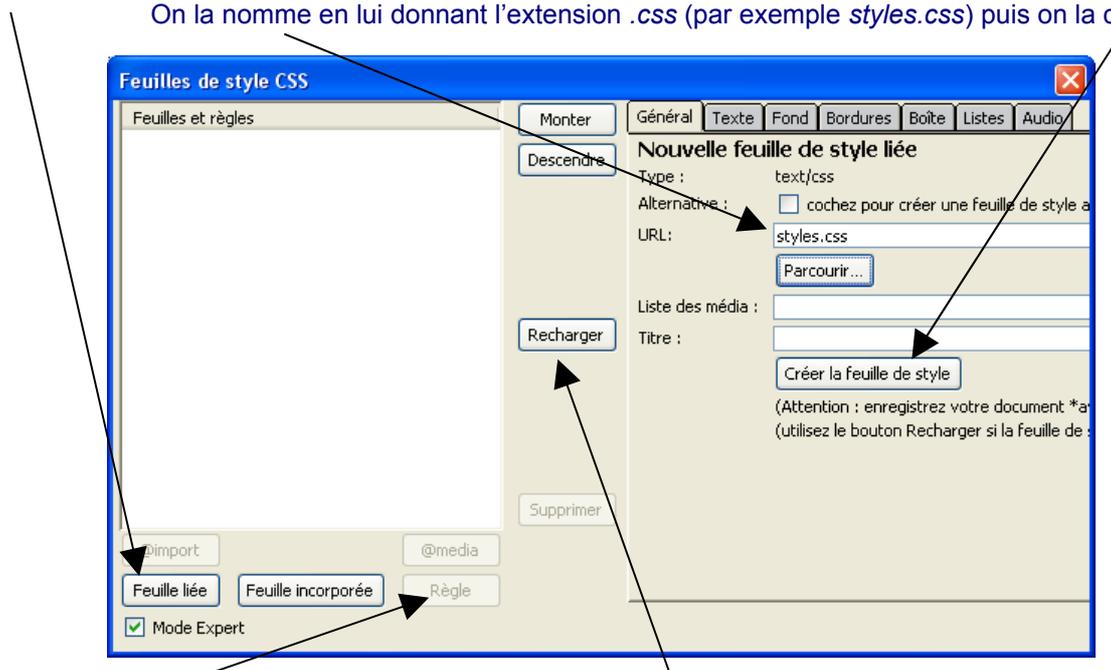
Un squelette comportant les balises <head> et <body> est observable dans l'onglet « Source ».

Création des styles

On va maintenant créer les styles associés à ce document : « Outils, Editeur CSS »

On veut une feuille de styles externe³ : « Feuille liée »

On la nomme en lui donnant l'extension .css (par exemple *styles.css*) puis on la crée.



Si le bouton Règle n'est pas accessible, on clique sur « Recharger » (ne fonctionne que si la page HTML précédente est déjà enregistrée).

Pour créer chaque règle, on clique sur Règle puis on sélectionne le type de style.

Par exemple pour le style *.soustitre* :



² au démarrage de NVU, un nouveau document vide est systématiquement créé

³ une feuille externe permet de gérer les styles de façon indépendante

Autre exemple, pour le style attaché aux balises <body> :

Nouvelle règle de style

style nommé (entrez un nom de classe ci-dessous)

Créer un nouveau : style appliqué à tous les éléments d'un type (entrez le type ci-dessous)

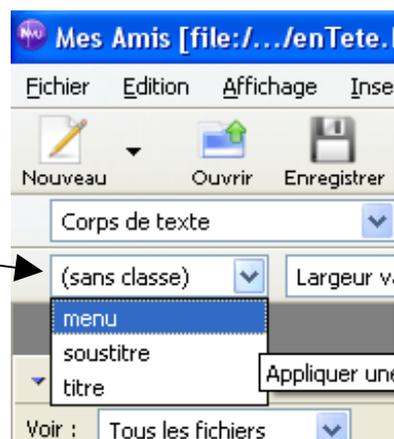
style appliqué à tous les éléments correspondant au sélecteur ci-dessous

body

Créer la règle de style

Pour chacun de ces styles, on définit bien évidemment ses caractéristiques (couleurs, tailles, alignements...) à l'aide des onglets Texte, Fond... Ces caractéristiques sont résumées dans l'onglet Général.

Quand tous les styles ont été définis, les styles nommés apparaissent dans la liste adéquate. Les styles attachés aux balises HTML sont automatiquement pris en compte.



Conception de l'en-tête

La feuille de style est créée ; le fichier d'en-tête contient maintenant un lien vers cette feuille, visible dans l'onglet Source (balise <link>).

On va maintenant concevoir le contenu de ce fichier en-tête.

Dans l'onglet Normal, on saisit le titre ; on veille à ce qu'il soit placé dans une balise <p> : après avoir sélectionné le texte, « Format / Paragraphe / Paragraphe ».

On place une image (icône Image, et on choisit une URL relative) suivie d'une ligne horizontale (Insertion, Ligne horizontale).

Pour appliquer le style « titre » au titre saisi on le sélectionne puis on choisit le style dans la liste.

La page se présente dès lors ainsi :



On enregistre enfin la page (son nom a déjà été fourni : *entete.html*)

Conseil : un passage fréquent par l'onglet « Source » permet de détecter et de réparer les erreurs de manipulations réalisées ; c'est aussi la possibilité de placer précisément le pointeur d'insertion.

Mise en forme du fichier d'en-tête

Techniquement, cette page HTML ne doit être qu'un fragment de code qui sera inclus dans chaque page par une instruction PHP « require() » : il faut donc la débarrasser du code HTML inutile.

Pour cela, à l'aide de Notepad++, on efface les deux dernières lignes (</body> et </html>) puis on enregistre le fichier.

Interface web *menu.php*

Conception de la page

On crée une nouvelle page avec NVU, que l'on enregistre sous le nom de *menu.php*

Il n'est pas nécessaire de donner une valeur à la balise <title> puisque ce sera celle de l'en-tête qui sera utilisée.

Cette page devra utiliser les styles déjà définis. En mode exécution (sous le contrôle du serveur web) l'en-tête qui va être inclus se chargera d'appeler la feuille de style. Mais en mode conception, il faut utiliser une astuce, c'est à dire lier la feuille de styles afin de pouvoir les utiliser. En d'autres termes, il s'agit de créer une ligne <link> dans la page.

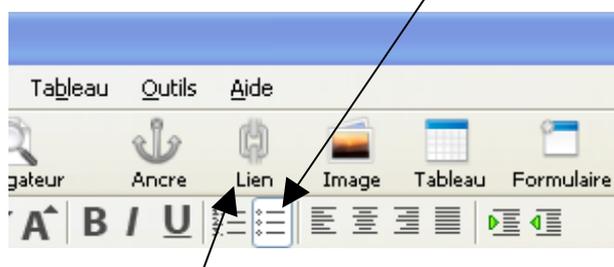
On va dans « Outils / Editeur CSS / Feuille liée / Parcourir », on sélectionne la feuille et on clique sur le bouton « Créer la feuille de style ».

On saisit ensuite le code php permettant l'insertion de l'en-tête : « Insertion / Code PHP », puis on saisit dans la nouvelle fenêtre le code :

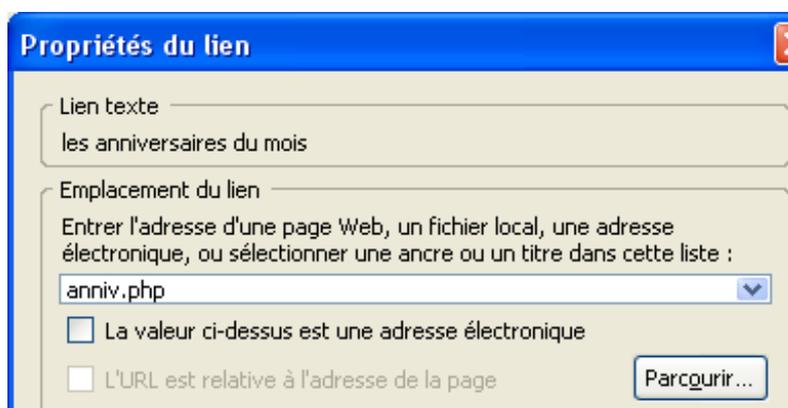
```
require (\"entete.html\");
```

Une balise « PHP » apparaît alors sur la page.

On saisit le texte du menu puis après l'avoir sélectionné, on lui applique le style prédéfini « liste à puces » puis le style *.menu*



Après avoir sélectionné chacune des lignes du menu, on crée les liens vers les pages correspondantes.



Mise en forme du fichier

À l'aide de Notepad++, on édite le fichier *menu.php*

Ce fichier doit maintenant être débarrassé du code HTML inutile, car déjà présent dans le fichier inclus *entete.html*.

Après effacement du début, le fichier doit avoir cet aspect :

```
menu.php
1  <?php require ("entete.html");?><br>
2
3  <br>
4
5  <ul class="menu">
6
7      <li><a href="anniv.php">les anniversaires du mois</a></li>
8
9      <li><a href="nouvel.php">saisie d'un nouvel ami</a></li>
10
11 </ul>
12
13 </body>
14 </html>
```

Remarque : dans cette conception, nous avons utilisé NVU pour incorporer le code PHP dans la page. Nous verrons par la suite que cette solution, si elle facilite un peu la saisie, n'est pas toujours possible : il faudra donc généralement passer par un éditeur de texte pur (Notepad++) pour réaliser cette action.

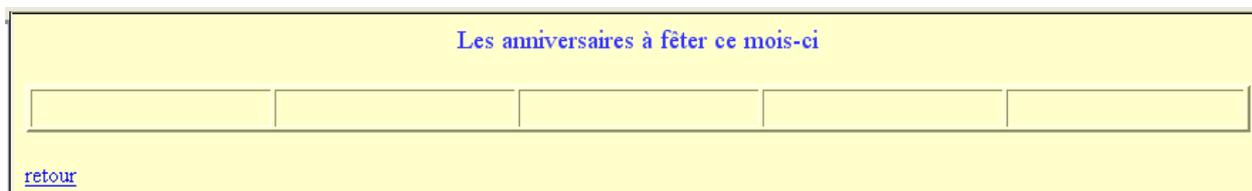
Interface web *anniv.php*

Comme précédemment, on crée une nouvelle page et on lie provisoirement la feuille de style. On saisit le sous-titre de la page puis on lui applique son style.

Nous souhaitons ensuite afficher les données concernant les amis dans un tableau. Le nombre de lignes de ce tableau sera géré par une itérative php, le nombre de colonnes correspond aux données à afficher (ici cinq).

On crée donc un tableau de une ligne et cinq colonnes puis le message « retour » avec un lien vers la page *menu.php*.

L'interface doit avoir cet aspect :



Il nous faut maintenant écrire le code PHP dont l'emplacement doit être convenablement choisi. Il s'avère qu'avec NVU en mode graphique il n'est pas possible de fixer le point d'insertion juste après la balise `<table>`. La modification du « Source » étant sujette à des réparations automatiques de code HTML, nous allons donc utiliser Notepad++ pour réaliser les modifications sur le code.

Le code PHP à écrire peut avoir cet aspect (on se reportera aux séquences proposées concernant l'apprentissage de PHP dans la description du thème) :

```
1 require ("entete.html");
2 require ("fonctions.php");
3 $connexion = mysql_connect("localhost","root","");
4 if ($connexion)
5 {
6     mysql_select_db("CARNET",$connexion);
7     $requete = donneReqAnniversaires(date("m"));
8     $resultat= mysql_query($requete,$connexion);
9     $ligne = mysql_fetch_assoc($resultat);
10    while ($ligne)
11    {
12        echo $ligne["nom"];
13        echo $ligne["prenom"];
14        echo $ligne["dateNaissance"];
15        echo $ligne["telephone"];
16        echo $ligne["adresseCourriel"];
17        $ligne=mysql_fetch_assoc($resultat);
18    }
19 }
20 else
21 {
22     echo "problème de connexion <br />";
23 }
24 mysql_close($connexion);
```

Le code généré automatiquement par NVU se présente ainsi :

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2 <html>
3 <head>
4     <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
5     <title></title>
6     <link rel="stylesheet" href="file:///E:/Documents%20and%20Settings/JPh/Mes%20documents/r%E
7 </head>
8 <body>
9     <p class="soustitre">Les anniversaires &agrave; f&ecirc;ter ce mois-ci</p>
10    <br>
11    <table style="text-align: left; width: 100%;" border="1" cellpadding="2" cellspacing="2">
12    <tbody>
13    <tr>
14        <td></td>
15        <td></td>
16        <td></td>
17        <td></td>
18        <td></td>
19    </tr>
20    </tbody>
21    </table>
22    <br>
23    <a href="menu.php">retour</a><br>
24 </body>
25 </html>
```

NVU a construit le squelette HTML de l'application en plaçant les bonnes balises aux bons endroits ; on doit écrire le code PHP, le placer aux bons emplacements et supprimer les balises HTML inutiles. En utilisant Notepad++, on arrive à la fusion suivante :

```

1  <?php require ("entete.html");?>
2  <p class="soustitre">Les anniversaires &agrave; f&ecirc;ter ce mois-ci</p><br>
3  <table>
4  <?php
5      require ("fonctions.php");
6      $connexion = mysql_connect("localhost","root","");
7      if ($connexion)
8      {
9          mysql_select_db("CARNET", $connexion);
10         $requete = donneReqAnniversaires(date("m"));
11         $resultat= mysql_query($requete,$connexion);
12         $ligne = mysql_fetch_assoc($resultat);
13         while ($ligne)
14         {
15             ?>
16             <tr>
17                 <td><?php echo $ligne["nom"];?></td>
18                 <td><?php echo $ligne["prenom"];?></td>
19                 <td><?php echo $ligne["dateNaissance"];?></td>
20                 <td><?php echo $ligne["telephone"];?></td>
21                 <td><?php echo $ligne["adresseCourriel"];?></td>
22             </tr>
23         <?php
24             $ligne=mysql_fetch_assoc($resultat);
25         }
26     }
27     else
28     {
29         echo "problème de connexion <br />";
30     }
31     mysql_close($connexion);
32     ?>
33 </table>
34 <br>
35 <a href="menu.php">retour</a><br>
36 </body>
37 </html>

```

Interface web *nouvel.php*

Conception de l'interface de saisie

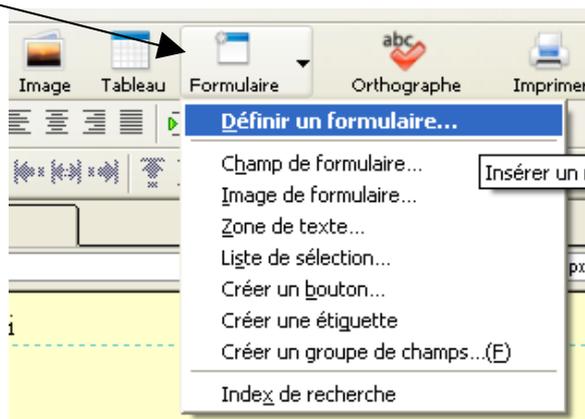
Nous allons maintenant utiliser la même technique pour concevoir les pages :

- utilisation de NVU pour la gestion des balises HTML (fort utile en cas de tableaux, de formulaires ou de structures complexes) et des feuilles de style ;
- recours à Notepad++ pour :
 - o écrire le programme PHP ;
 - o simplifier le code HTML produit par NVU ;
 - o incorporer le code PHP dans la page HTML.

On utilise NVU pour créer une nouvelle page et la lier provisoirement à la feuille de style. On saisit le sous-titre de la page puis on lui applique son style.

On va maintenant construire un formulaire pour permettre la saisie d'un nouvel ami :
« icône Formulaire / Définir un formulaire »

On définit ensuite ses propriétés (*enregAmi.php* est le fichier chargé de l'enregistrement des données dans la base) :



On place ensuite les cinq zones de saisies permettant de caractériser un nouvel ami : « icône Formulaire / Champ de formulaire / Texte » puis on donne un nom à chacun de ces champs.

Chaque zone de saisie est précédée d'un texte explicatif.

On place enfin le bouton permettant l'envoi des données saisies : « icône Formulaire / Champ de formulaire / Bouton de validation ». On donne une valeur au bouton (ce qui est affiché dedans)

À l'état brut, sans avoir défini les longueurs des champs de saisie (adaptées aux dimensions définies dans la base de données) ni les avoir positionnés correctement sur la page (avec un tableau par exemple), on arrive à une interface :

Après reprise du code HTML à l'aide de Notepad++, on obtient un fichier *nouvel.php* :

```

1  <?php require ("entete.html");?>
2  <p class="soustitre">Saisie d'un nouvel ami</p>
3
4  <form method="post" action="enregAmi.php" name="saisieAmi">
5      Nom : <input name="nom"><br>
6          <br>
7      Prénom : <input name="prenom"><br>
8          <br>
9      Date de naissance : <input name="dateNaissance"><br>
10         <br>
11      Téléphone : <input name="telephone"><br>
12         <br>
13      Adresse courriel : <input name="adresseCourriel"><br>
14         <br>
15         <input value="Envoyer" type="submit">
16     </form>
17     <br>
18     <a href="menu.php">retour</a><br>
19 </body>
20 </html>

```

Conception du programme d'enregistrement

On se référera utilement aux applications PHP citées dans la description du thème pour construire un programme enregistrant les données reçues depuis le formulaire.

Ce programme renvoie ensuite sur un nouvel interface de saisie

```
1 <?
2 $connexion = mysql_connect("localhost","root","");
3 if ($connexion)
4 {
5     require ("fonctions.php");
6     mysql_select_db("CARNET", $connexion);
7     $requete=donneReqCreeAmi($_POST["nom"], $_POST["prenom"], $_POST["dateNaissance"], $_POST["telephone"], $_POST["adresseCourriel"]);
8     mysql_query($requete, $connexion);
9     mysql_close($connexion);
10    require ("nouvel.php");
11 }
12 else
13 {
14     echo "problème à la connexion <br />";
15 }
16 ?>
```

Conclusion

Le logiciel NVU utilisé ici présente un certain nombre d'avantages mais aussi d'inconvénients :

- c'est un logiciel libre dont l'utilisation en classe de terminale est gratuite ;
- c'est un logiciel qui respecte les standards du web ;
- il propose différentes visions d'une même page web (graphique, source...) ;
- il gère correctement les feuilles de style CSS ;
- il permet des inclusions de code PHP même si cette fonctionnalité n'est pas très utile ;
- il surveille et reconstitue systématiquement les balises HTML d'une page, ce qui est un inconvénient lors de la construction de pages dynamiques ; l'utilisation conjointe de Notepad++ permet de passer outre cet inconvénient.

L'élève est placé dans une situation de conception assistée : les balises HTML les plus délicates (par exemple des tableaux imbriqués) sont construites automatiquement, ce qui n'empêche pas qu'il doive comprendre la logique mise en œuvre afin de pouvoir insérer à bon escient le code PHP nécessaire.

Pour reprendre les conseils formulés dans la dernière partie, on peut proposer une démarche :

- utilisation de NVU pour la gestion des balises HTML et des feuilles de style ;
- recours à Notepad++ pour :
 - o écrire le programme PHP ;
 - o simplifier le code HTML produit par NVU ;
 - o incorporer le code PHP dans la page HTML.