



DOUCHETTE PORTABLE

Cazenave Ketzia / Jean-Baptiste Jonathan /
Thomas Cédric



Introduction

A ce jour, les entreprises n'autorisent aucune erreur de gestion, elle constitue de plus en plus une préoccupation auprès des dirigeants d'entreprises.

Les stocks constituent des valeurs d'exploitation à gérer. La gestion des stocks consiste à planifier, organiser, diriger et contrôler les activités relatives à tous les stocks de marchandises gardées dans l'entreprise. Toutefois, la gestion de stocks est tout un processus allant de l'achat des marchandises, à leur entrée en stock, à leur sortie en stock et à leur comptabilisation. Il urge pour toute entreprise d'instituer impérativement une gestion saine des stocks dont elle a la possession.

Et c'est en considération de l'importance de cette discipline qu'est la gestion des stocks que nous avons décidé d'étudier cette notion à travers une action très simple, qu'est l'inventaire.

Tout d'abord, nous allons présenter brièvement l'entreprise, puis nous passerons à la présentation du système proposé. Ensuite, nous allons étudier comment peut-on visualiser le contenu du stock dans une base de données ; deuxièmement, nous allons démontrer comment obtenir l'autonomie de la solution proposée, et en dernier lieu, nous étudierons l'envoi des données à la base de données.

SUIVI D'INVENTAIRE DE STOCK

L'inventaire physique est l'opération qui consiste à relever tous les éléments du patrimoine de l'entreprise en mentionnant la nature, la quantité et la valeur de chacun d'eux à la date d'inventaire.

Problèmes liés à la gestion de stock

Certaines entreprises utilisent toujours la méthode traditionnelle, c'est-à-dire, la méthode manuelle. Muni d'une feuille, l'employé devait rentrer à la main toute la marchandise existant dans le stock. Dans ce cas, les risques d'erreurs sont très importants, et cette opération peut prendre énormément de temps.

SOMMAIRE

Présentation de l'entreprise.....	4
Fiche d'identification.....	4
Douchette portable.....	5
Présentation du système.....	5
Etudiant 1.....	7
Visualiser les données saisies dans la base, les rapprochements en cours.....	8
Objet : IHM permettant la connexion à la base.....	9
Fiche de test n°1.....	12
Fiche de test n°2.....	13
Fiche de test n°3.....	14
Visualisation des données dans la base.....	15
Fiche de test n°4.....	21
Fiche de test n°5.....	22
Etudiant 2.....	23
Etudiant 3.....	24
Cahier des charges.....	25
Présentation.....	25
Critères de fonctionnement.....	25
Critères techniques.....	26
Bête à cornes.....	26
Pieuvre.....	27
Contexte.....	27
Conception.....	28
Programmation du microcontrôleur Beaglebone.....	29
Etudiant 1,2,3.....	30
Rédaction d'un cahier de mission.....	31
Annexes.....	32

Annexe 1.....	33
Annexe 2.....	34
Annexe 3.....	35
Annexe 4.....	36
Annexe 5.....	38
Annexe 6.....	40
Annexe 7.....	41
Annexe 8.....	42
Annexe 9.....	43

PRESENTATION DE L'ENTREPRISE

DIFFUSION GENERALE QUINCAILLERIE (DIGEQ), société par actions simplifiée est active depuis 40 ans.

Domiciliée à BAIE MAHAULT (97122), elle est spécialisée dans le secteur d'activité du commerce de gros (commerce inter-entreprises) de quincaillerie. Son effectif est compris entre 20 et 49 salariés.

Sur l'année 2010 elle réalise un chiffre d'affaires de 16 566 900,00 €.

Francois BROT, est président de l'entreprise DIFFUSION GENERALE QUINCAILLERIE.

FICHE D'IDENTIFICATION

Dénomination	SOCIETE DIFFUSION GENERALE DE QUINCAILLERIE
Adresse	7 B IMP JEAN MARIE JACQUARD 97122 BAIE MAHAULT
Pays	France
SIREN	312 398 118
SIRET (siège)	31239811800017
Activité (Code NAF ou APE)	Commerce de gros (commerce inter-entreprises) de quincaillerie
Forme juridique	Société par actions simplifiée
Tranche d'effectif	20 à 49 salariés
Capital social	762 500,00 €
Date création entreprise	01-01-1978

DOUCHETTE PORTABLE

Qu'est-ce que la douchette portable ? Pourquoi l'utiliser ?

La douchette portable est un outil permettant de scanner des codes barres. Son utilisation est simple et rapide car une fois le code barre scanner, les informations que renferme celui-ci sont directement envoyées à la base de données. Ce processus minimise les risques d'erreurs et est beaucoup plus rapide.

Présentation du système

Matériels utilisés :

- Douchette portable USB
- Écran LCD
- Carte graphique BeagleBone Black
- Serveur NAS

Comment la connexion est établie ?



La douchette est branchée en USB, au dispositif BBB (Beagle Bone Black) + écran, cet ensemble est connecté au serveur NAS.

Un écran LCD, pourquoi ?

La douchette n'est pas équipée d'une durée de vie illimitée. C'est pour cela que nous avons eu l'idée d'y associer un écran permettant à l'utilisateur de connaître le niveau de batterie de celle-ci.

Q'est-ce qu'un serveur NAS ?

Un serveur NAS (Network Attached Storage) est un petit ordinateur entièrement tourné autour d'un système de stockage.

PRINCIPE DU CODE BARRE

- Un code-barres, ou code à barres, est la représentation d'une donnée numérique ou alphanumérique sous forme d'un symbole constitué de barres et d'espaces dont l'épaisseur varie en fonction de la symbologie utilisée et des données ainsi codées.

On distingue deux types généraux de codes barre :

- Unidimensionnel (1D) : ces codes sont ceux représentés par une série de lignes parallèles d'épaisseur variable. Leur lecture est unidimensionnelle. Selon la technologie de lecture utilisée, le décodage pourra se faire de façon unidirectionnelle ou bidirectionnelle afin de confirmer le premier décodage.

- Bidimensionnel (2D) : ces codes utilisent une variété de symboles (rectangles, points, hexagones et autres formes géométriques). Cette forme matricielle permet d'enregistrer davantage d'informations.

On distingue deux familles de codes 2D :

- Les codes empilés :

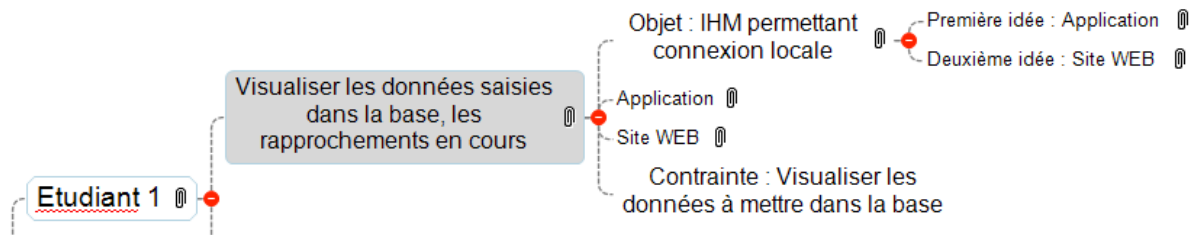
Il s'agit de codes 1D empilés (Code 16k, PDF417, etc.). Ces codes peuvent aussi être lus par les lecteurs 1D en faisant un balayage du code.

- Les codes bi-dimensionnels :

Il s'agit de codes dont les motifs constituent une forme souvent rectangulaire ou carrée qui ne peuvent être lus que par des technologies de prise de photos.

ETUDIANT 1

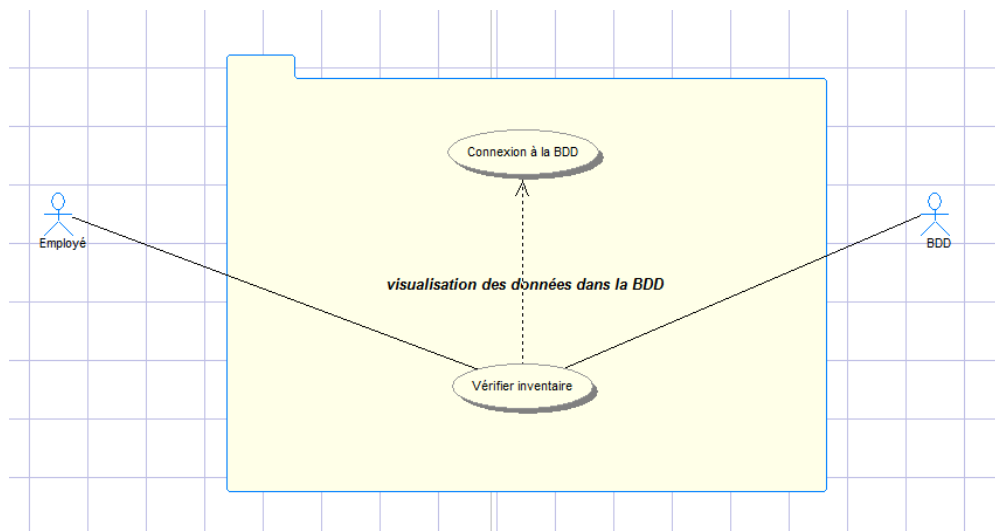
Cazenave Ketzia



Pour toute bonne gestion de stock, une vérification est nécessaire. Dans notre cas, la vérification ne se fait non pas sur papier, mais est belle et bien numérique.

I. VISUALISER LES DONNEES SAISIES DANS LA BASE, LES RAPPROCHEMENTS EN COURS

- Diagramme de cas d'utilisation

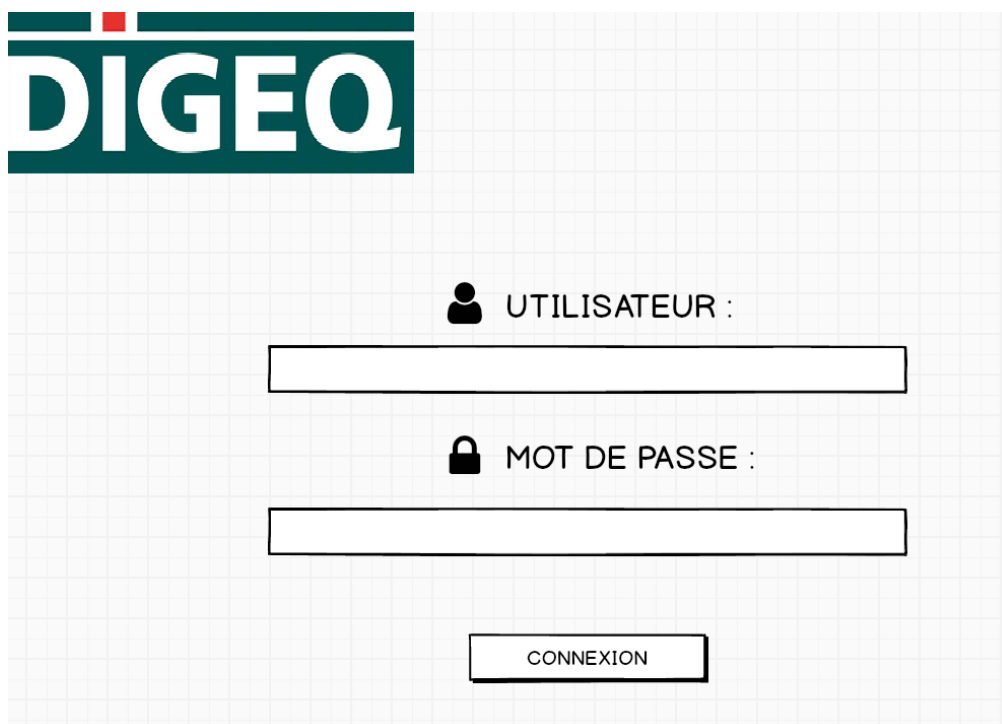


1. OBJET : IHM PERMETTANT CONNEXION

Ce n'est pas n'importe quel employé qui pourra visualiser le contenu de la base de données. Les employés désignés, habilités seront munis d'un identifiant et d'un mot de passe, pour effectuer une connexion à la base.

- **PREMIERE IDEE : APPLICATION**

Comme je l'ai précisé au début, ce n'est pas tous les employés de l'entreprise, qui pourront accéder à la plate-forme de gestion de stocks. Les employés habilités seront munis d'un identifiant et d'un mot de passe pour pouvoir se connecter.



The image shows a login interface for a system named DIGEO. The logo 'DIGEO' is in the top left corner, with 'DI' in white on a dark green background and 'GEO' in white on a dark green background. Below the logo, there are two input fields. The first field is labeled 'UTILISATEUR :' with a user icon. The second field is labeled 'MOT DE PASSE :' with a lock icon. Below these fields is a button labeled 'CONNEXION'.

- **DEUXIEME IDEE : SITE WEB (VOIR ANNEXE 1,2)**

L'idée suivante est plus simple que la première. Les employés iraient simplement sur un site WEB, pour se connecter et accéder à la base de données, et pour éventuellement s'inscrire s'il n'y sont pas déjà.



Pour pouvoir afficher cette page web, j'ai généré le code Html suivant :

- `<h2 align= »center »><u><i>Votre identifiant : </u></i></h2> <h2 align= »center »><input type= »text » name= »user » ></h2>
`
- `<h2 align= »center »><u><i>Votre mot de passe : </u></i></h2> <h2 align= »center »><input type= »password » name= »passwd » ></h2>
`

Ces deux lignes de codes représentent les deux zones de saisie.

- `<h2 align= »center »><input type= »submit » value= »connexion »></h2>
`

Celle-ci représente le bouton connexion.

- `<h2 align= »center »><input type= »reset » value= »Annuler »></h2>
`
Et celle-là le bouton Annuler, qui une fois activé, efface tout ce qui est marqué dans les deux zones de saisie.

Le lien « Inscription » renvoie à la page « PageInscription.php » :

```
<h3 align= »center »><a href= »PageInscription.php »>Inscription</a></h3>
```

L'action « Connexion », établie une connexion avec la base de données :

```
mysql_connect($host, $user,$passwd, $database) or die(« erreur de  
connexion au serveur ») ;  
mysql_select_db($database) or die(« erreur de connexion a la base de  
donnees ») ;
```

Nous utilisons ces deux lignes de code pour pouvoir nous connecter à la base.

Ici, les zones de saisies sont générées de la même manière que pour la connexion, même processus pour le bouton « M'inscrire » que pour « Connexion » par exemple.

L'action « M'inscrire », envoie les informations saisies dans les champs sur la base de données dans la table dédiée à l'enregistrement des employés habilités à accéder à la base de données.

La ligne de code utilisée est la suivante :

```
$sql = $connect->query("INSERT INTO agent VALUES(", '$identifiant', '$mdp')");
```

FICHE DE TEST N°1

Présentation du test :

Nom du projet : Douchette portable

Nature du test : Unitaire

Déroulement du test :

Nom(s) du (des) fichier(s) de test :

- Pagehtml1.html

-PageConnexion.php

Procédure de test :

-Connexion à la base de données

Rapport de Test :

Résultat du test : Refusé

Compte rendu : La connexion à la base de données n'a pu être établie, car la méthode `mysql_connect()` / `mysql_select_database()`, sont des méthodes trop anciennes.

Solution :

Les méthodes citées précédemment sont trop obsolètes, je l'ai donc remplacé par la méthode : PDO.

Ce qui donne :

```
$Cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
```

```
$connect= new PDO($Cible, $user, $passwd);
```

FICHE DE TEST N°2

Présentation du test :

Nom du projet : Douchette portable

Nature du test : Unitaire

Déroulement du test :

Nom(s) du (des) fichier(s) de test :

- Pagehtml1.html

-PageConnexion.php

Procédure de test :

-Connexion à la base de données

Rapport de Test :

Résultat du test : Accepté

Compte rendu : La connexion à la base de données a été établi.

FICHE DE TEST N°3

Présentation du test :

Nom du projet : Douchette portable

Nature du test : Unitaire

Déroulement du test :

Nom(s) du (des) fichier(s) de test :

- Pagehtml1.html
- PageConnexion.php
- PageInscriptions.php

Procédure de test :

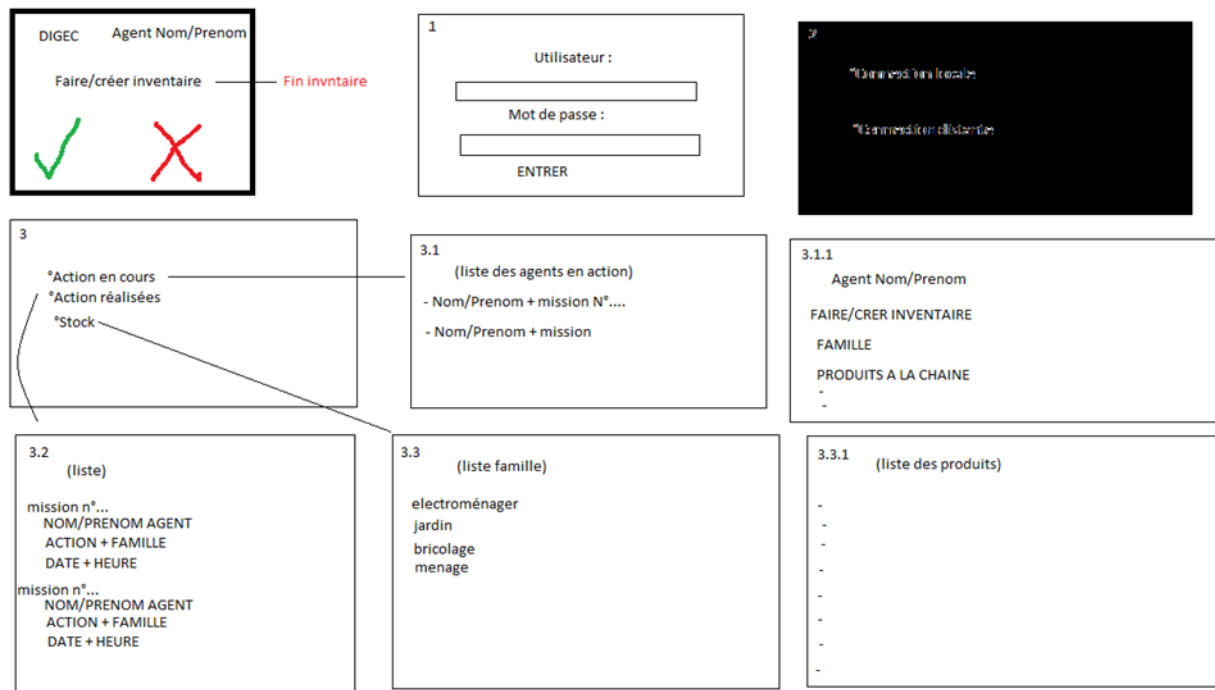
- Envoie des données d'inscription à la base

Rapport de Test :

Résultat du test : Refusé

Compte rendu : La connexion à la base de données est toujours active, mais les données rentrées dans les zones de saisies n'arrivent pas jusqu'à la base.

2. VISUALISATION DES DONNEES DANS LA BASE



• AVEC L'APPLICATION

Au départ, j'ai pensé à utiliser l'écran LCD pour que l'agent ne voit pas seulement autonomie de la batterie de la douchette, mais qu'il puisse également, à travers cet écran constaté en temps réel son travail.

C'est-à-dire :

Tout d'abord, les agents seraient munis d'un code barre qui leur est propre. Pour commencer à travailler, il scannerai leur code barre, leur nom s'afficherai en haut à droite sur l'écran.

A chaque code barre scanné, l'action s'afficherai sur l'écran (Inventaire/Débuter/Pause/Reprendre/Terminer).

Trois icônes y sont également : une verte pour affirmer que la lecture a bien été effectuée, le code barre est bon, une rouge pour un mauvais code barre, une jaune pour savoir si le code barre à déjà été lu.

Et puis en bas à gauche, le niveau de batterie.



Une fois connecter, ils pourront choisir entre trois actions différentes :

- Les actions en cours (Voir en temps réel le travail des agents avec la douchette)
- Les actions déjà réalisées
- Le stock

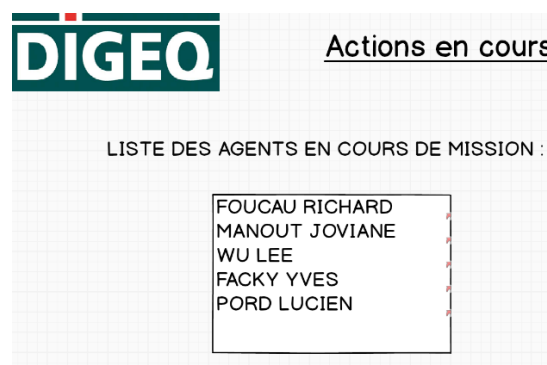
Ces trois possibilités seront affichées sur l'écran sous forme de lien.



- **Actions en cours :**

La fonction "Actions en cours", s'affiche en haut à droite de l'écran, de manière à nous situer.

Il peut y avoir plusieurs agents travaillant sur différents inventaires, dans ce cas l'employé peut choisir de vérifier le travail d'un agent spécifique



Une fois qu'un agent a été sélectionné, le numéro de la mission en cours
Son nom s'affiche, la famille dans laquelle il travail, et la liste des produits scanné en temps réel.

DIGEQ Actions en cours

FOUCAU RICHARD

- N° de mission
- Faire / créer inventaire
- Famille

- PRODUIT
- PRODUIT
- PRODUIT

Les actions réalisées sont des missions déjà effectuées, elles s'affichent sous forme de liste.

DIGEQ Actions réalisées

- Mission n°1259
- Mission n°5986
- Mission n°5214
- Mission n°8546
- Mission n°4856
- Mission n°1259

Le numéro de la mission s'affiche en haut à droite de l'écran.
Le nom de l'agent qui l'a réalisé, la date et l'heure à laquelle elle a été effectué, la famille et la liste des produits étudiés.

DIGEQ Mission n°1259

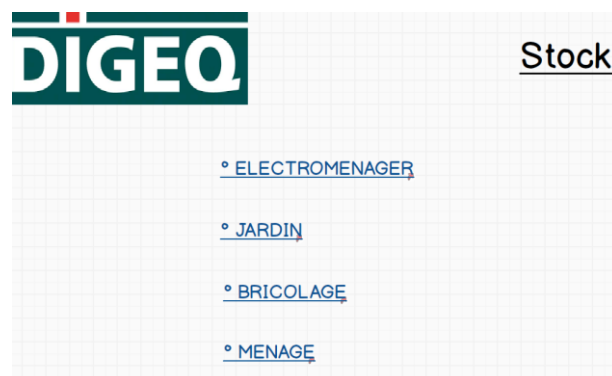
Agent : NOM PRENOM

DATE et HEURE

Familles des produits

- produit
- produit
- produit
- produit

Pour la vérification du stock, il y a quatre familles que l'employé peut sélectionner.



- **AVEC LE SITE WEB (VOIR ANNEXES 1,2,3,4,5,6,7)**

Une fois la connexion établie, la base de données s'affiche sous forme de tableau :

[Deconnexion](#)

[Electromenager](#)

[Bricolage](#)

[Jardin](#)

[Menage](#)

Nom	Reference	Marque	Categorie
scie	SCIE789	rapide	Electroménager
Nom	Reference	Marque	Categorie
engrais rose	ROSE459	fertiligene	Jardin
Nom	Reference	Marque	Categorie
serpillère	SERPBA489	balais	Menage
Nom	Reference	Marque	Categorie
marteau piqueur	MARTPI7895	Casse	Bricolage

Lorsque l'employé rentre son identifiant et son mot de passe, la connexion à la base est établie. Une fois cette connexion établie, une requête y est envoyée, de façon à afficher les éléments de la base de données dans un tableau :

```
$reponse = $connect->query('SELECT * FROM digeq');
```

Sur cette même page, cinq liens apparaissent, Pourquoi ?

- Le premier lien « Deconnexion », renvoie à la première page de connexion :

```
<h3 align="right"><a href="PageHtml1.html">Deconnexion</a></h3>
```

- Le lien « Electromenager » renvoie sur une page où s'affiche uniquement un tableau contenant les éléments de la base de la catégorie Electroménager :

```
o <h3 align="right"><a href="PageElectromenager.php">Electromenager</a></h3>
```

Sur la page : PageElectromenager.php, la requête utilisée pour obtenir le tableau :

```
$reponse = $connect->query('SELECT Nom,Reference,Marque FROM digeq WHERE Categorie = Electromenager');
```

- Le lien « Bricolage » renvoie sur une page où s'affiche uniquement les éléments de la catégorie bricolage :

```
o <h3 align="right"><a href="PageBricolage.php">Bricolage</a></h3>
```

Sur la page : PageBricolage.php, la requête utilisée pour obtenir le tableau :

```
$reponse = $connect->query('SELECT Nom,Reference,Marque FROM digeq WHERE  
Categorie = Bricolage');
```

- Le lien « Jardin » renvoie sur une page où s'affiche uniquement les éléments de la catégorie jardin :

- `<h3 align="right">Jardin</h3>`

Sur la page : PageJardin.php, la requête utilisée pour obtenir le tableau :

```
$reponse = $connect->query('SELECT Nom,Reference,Marque FROM digeq WHERE  
Categorie = Jardin');
```

- Le lien « Menage » renvoie sur une page où s'affiche uniquement les éléments de la catégorie menage :

- `<h3 align="right">Menage</h3>`

Sur la page : PageMenage.php, la requête utilisée pour obtenir le tableau :

```
$reponse = $connect->query('SELECT Nom,Reference,Marque FROM digeq WHERE  
Categorie = Menage');
```

FICHE DE TEST N°4

Présentation du test :

Nom du projet : Douchette portable

Nature du test : Unitaire

Déroulement du test :

Nom(s) du (des) fichier(s) de test :

- Pagehtml1.html

-PageConnexion.php

Procédure de test :

-Connexion à la base de données

-affichage des données de la base dans un tableau

Rapport de Test :

Résultat du test : Accepté

Compte rendu : La connexion à la base de données a été établie. Le contenu de notre base de données, s'affiche dans un tableau.

FICHE DE TEST N°5

Présentation du test :

Nom du projet : Douchette portable

Nature du test : Unitaire

Déroulement du test :

Nom(s) du (des) fichier(s) de test :

- Pagehtml1.html
- PageConnexion.php
- PageElectromenager.php
- PageBricolage.php
- PageJardin.php
- PageMenage.php

Procédure de test :

- Connexion à la base de données
- affichage des données selon les catégories

Rapport de Test :

Résultat du test : Refusé

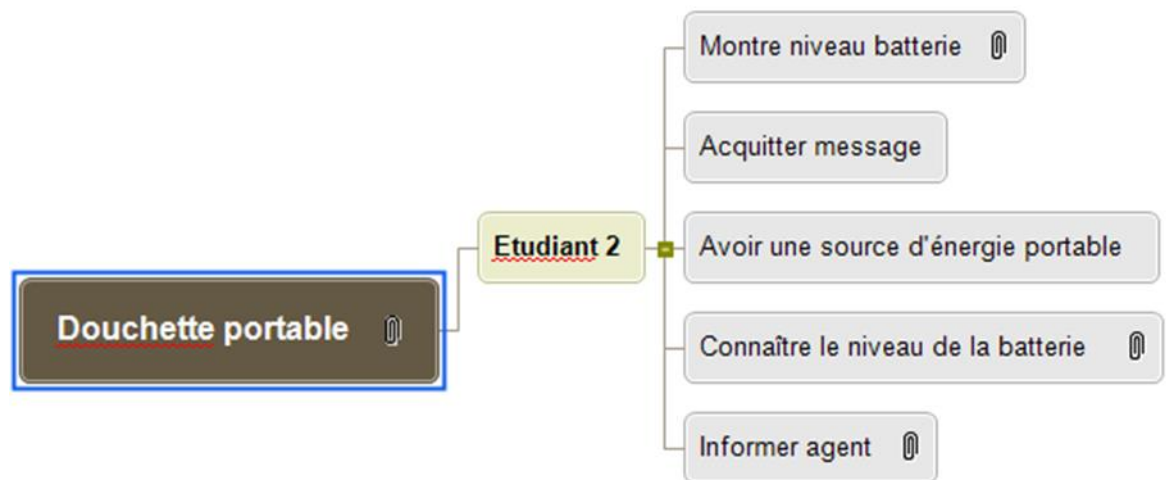
Compte rendu : La connexion à la base de données est toujours active, mais les liens renvoie une page vierge.

ETUDIANT 2

Thomas Cédric

ETUDIANT 3

Jean-Baptiste Jonathan



CAHIER DES CHARGES :

- Tout d'abord il faut programmer et indiquer au microcontrôleur ce qu'il doit faire
- Dans le cadre de ce projet on utilise une Beaglebone black afin de connaître et montrer le niveau de la batterie.

PRESENTATION :

- Il est important de savoir comment tester la batterie. Ces tests vont nous servir à connaître l'état de marche ainsi que son état de charge.
- Être capable de mesurer en temps réel la charge de la batterie.
- Afficher sur un écran LCD le pourcentage des informations de la batterie.

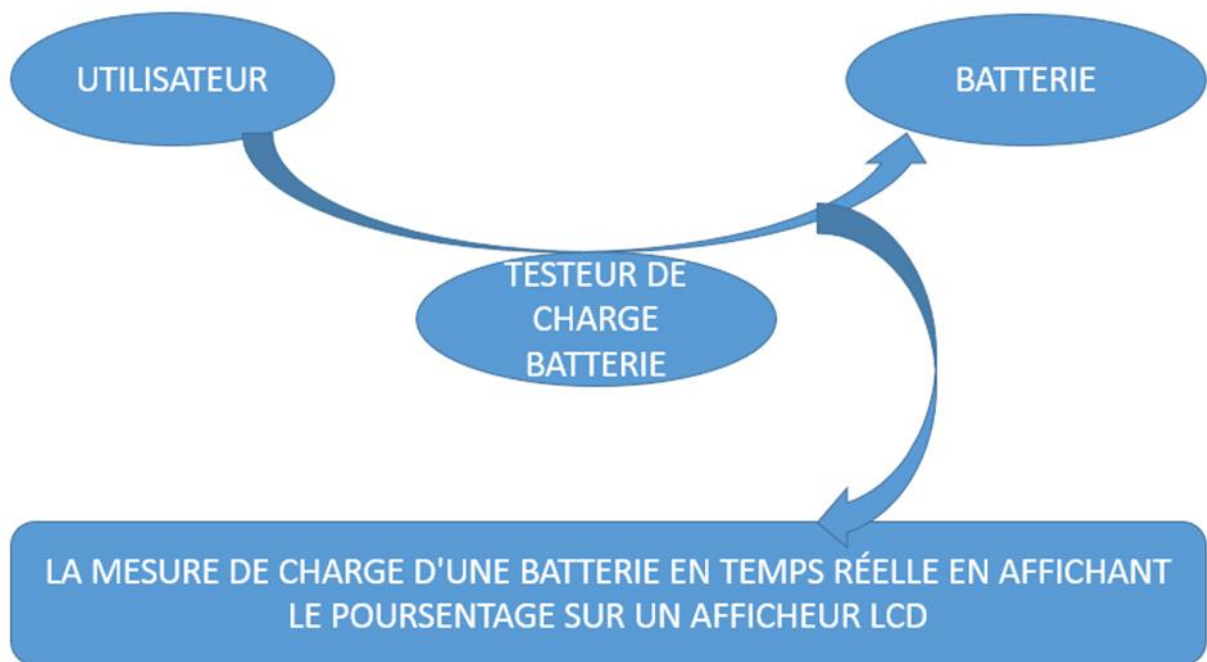
CRITERES DE FONCTIONNEMENT :

- Afin d'utiliser la carte sans l'endommager la tension doit être abaissée à l'aide d'un diviseur de tension à une valeur de 5V.
- La Beaglebone doit être reliée à un câble USB pour lui fournir l'alimentation nécessaire ainsi que le logiciel utilisé.
- Un afficheur LCD est nécessaire, pour pouvoir afficher toutes les informations sur la batterie.

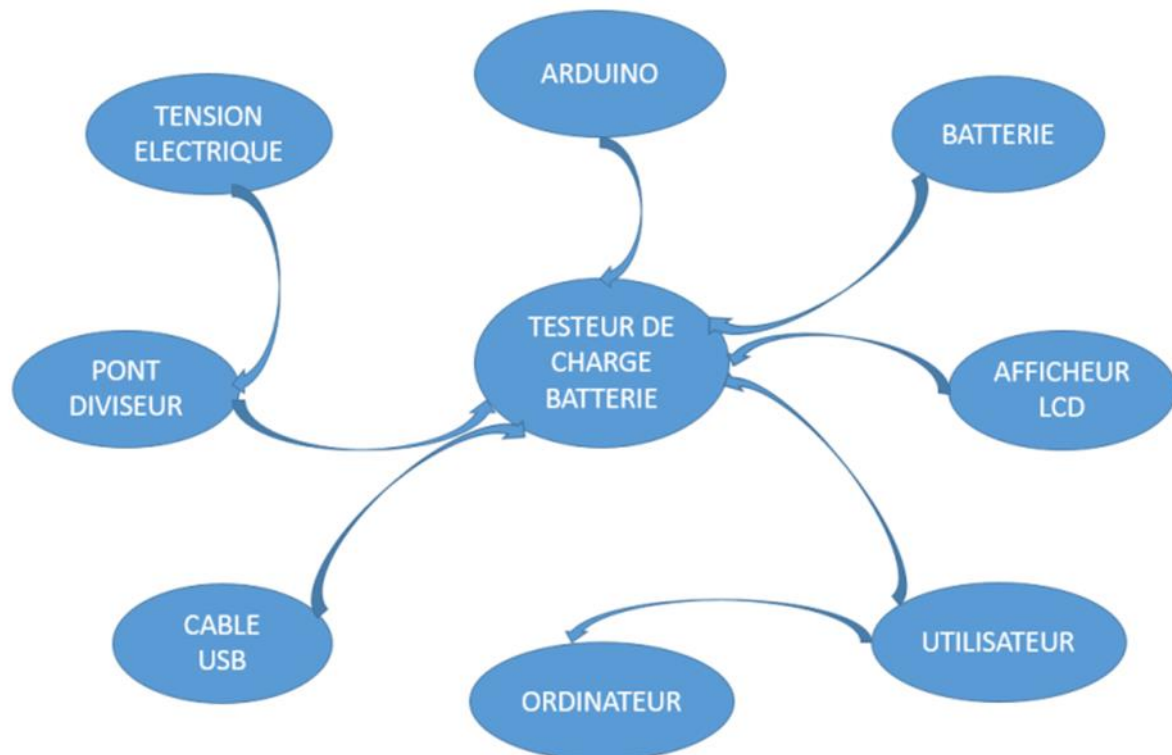
CRITERES TECHNIQUES :

- Câblages de l'afficheur LCD.
- Maitriser un peu l'électronique pour réaliser un diviseur de tension.

BETES A CORNES :

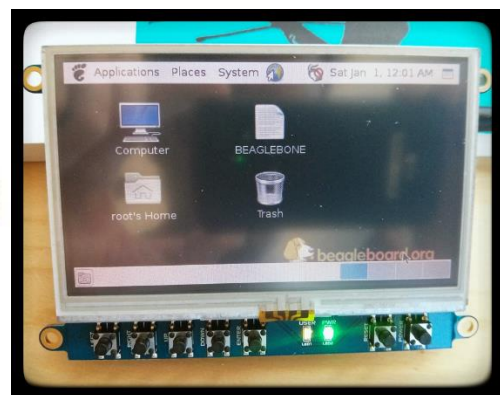
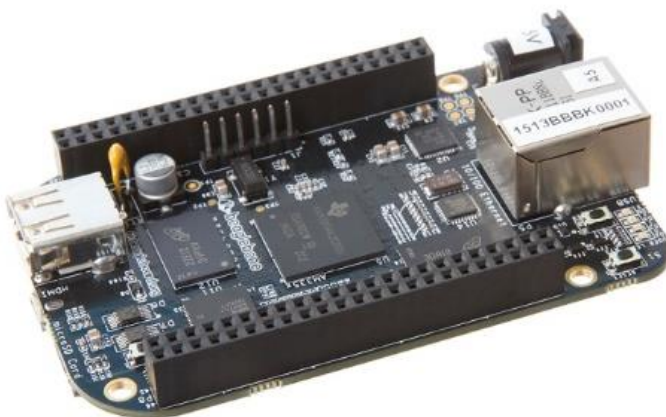


PIEUVRE :



CONTEXTE :

Pour réaliser ce projet j'ai à ma disposition un microcontrôleur Beaglebone black qui va permettre la mesure de la charge de la batterie, d'un écran LCD pour afficher les informations concernant la batterie et pour communiquer avec l'utilisateur.



CONCEPTION

Adaptation de la tension :

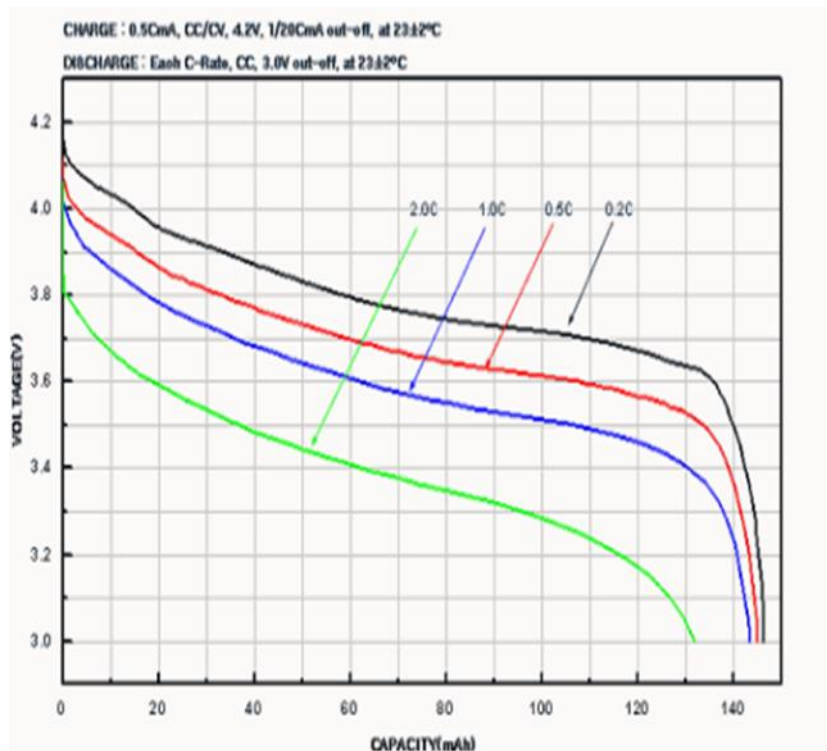
La première tâche effectuée a été l'adaptation de la tension de la batterie, pour pouvoir mesurer sa charge à l'aide de la Beaglebone black parce que cette dernière ne peut pas recevoir de tensions supérieures à 5V en entrée.

J'ai réalisé un diviseur de tension afin d'adapter les 12V délivrer par la batterie quand elle charge à 100% à une tension de 4.8V. J'ai réalisé ce diviseur de tension avec 2 résistances, la première est de 10KOhms et la deuxième de 15KOhms.

Câblage de l'afficheur LCD :

Dans un second temps j'ai réalisé le câblage de l'afficheur LCD sur la beaglebone black, de la batterie ainsi que le diviseur de tension.

En fonction de la tension délivrer la charge de la batterie peut vérifier à l'aide d'une courbe et d'un tableau représentant les tensions.



0%	3,00V
5%	3,30V
10%	3,60V
20%	3,70V
30%	3,75V
40%	3,79V
50%	3,83V
60%	3,87V
70%	3,92V
80%	3,97V
90%	4,10V
100%	4,20V

PROGRAMMATION DU MICROCONTROLEUR BEAGLEBONE :

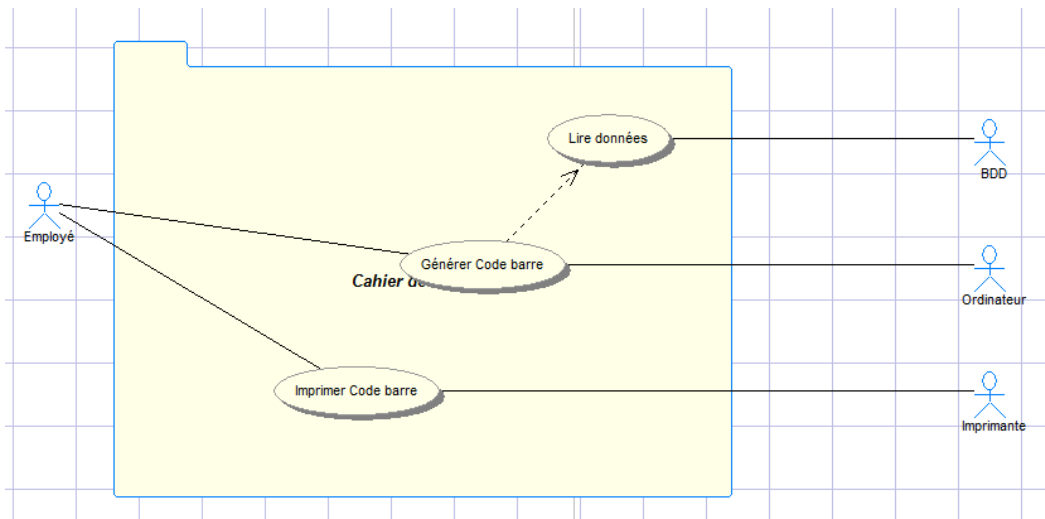
Ce programme consiste à tester plusieurs valeurs arrivant sur la patte Ain0.

En déterminant les valeurs à tester tout en sachant que la beaglebone travail sur du 12 bits et que la tension maximale est de 5V, alors l'entrée analogique A0 reçoit 5V elle correspondra à 4095 dans la beaglebone black

ETUDIANT 1,2,3

A. REDACTION D'UN CAHIER DE MISSION

- Diagramme de cas d'utilisation



Le cahier de mission est un ensemble de page où sont répertorié, les codes barre de chaque mission.

Qu'est-ce qu'une mission ?

Une mission comporte un numéro (exemple : Mission 1627). Elle est caractérisée par la famille dans lequel est fait l'inventaire (électroménager, jardin, bricolage ou ménage).

Les codes barre générés sont différents pour chaque mission.

MISSION 1246



INVENTAIRE



CATEGORIE



FIN INVENTAIRE

ANNEXES

A. ANNEXE 1 : FICHE D'INSTALLATION

Matériels fournis :

- Douchette portable USB
- Manuel de configuration de code barre
- Ecran LCD
- Support portable pour l'écran
- Serveur NAS
- Un câble Ethernet



-
- Un câble USB



- Un câble d'alimentation

Installation :

Branchez la douchette portable à l'écran LCD via le port USB.

Branchez le serveur NAS, à une prise avec le câble d'alimentation, utilisez le câble Ethernet pour relier le NAS à un port ethernet libre de votre Freebox, ou sur un port ethernet d'un switch ethernet branché sur le Server.

Le câble USB peut être utilisé pour permettre de brancher des disques durs supplémentaires et augmenter ainsi la capacité de stockage du NAS.

B. ANNEXE 2 : FICHE D'UTILISATION

Pour utiliser la douchette portable, il vous faut configurer la lecture des codes barre.

C'est-à-dire :

Sur le manuel de configuration, vous devrez scanner le code barre de début de configuration, une fois entrer en mode config, vous pourrez choisir un dès code barre correspondant au code barre de vos produits (EAN-13 ou EAN-8, par exemple), après avoir choisit vous devrez scanner le code barre de fin de configuration.

Vous pouvez commencer à travailler !

Pour commencer un inventaire, sortez une feuille de mission, scanner le code barre « Inventaire », puis le code barre « Categorie », commencez à scanner chaque produit correspondant à cette famille, une fois l'inventaire terminé, scannez le code barre « Fin inventaire », votre inventaire est terminé.

Tout au long de votre inventaire, des signaux sonores vous indiquerons la bonne comme la mauvaise lecture des codes barre.

Pour la vérification de l'inventaire :

Aller sur le site « dp-2018 ».

Si vous êtes habilité à faire ce travail connectez-vous, si vous n'êtes pas encore inscrit, le lien « Inscription » vous permet de le faire.

Une fois la connexion établie votre base de données s'affichera sous forme de tableau, donnant le nom, la marque, la référence, la catégorie et la quantité de chaque produit inventorié.

Vous pouvez choisir d'afficher que les produits de la catégorie souhaitée.

ANNEXE 3 : CODE SOURCE PAGEHTML1.HTML

```
<!DOCTYPE>
<html>
  <head>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <p><strong><u><font color="teal"><h1 align="center"><t>Digeq Entreprise</t></h1></font></u></strong></p><hr/>

    <form method="post" action="PageConnexion1.2.php">
      <h2 align="center"><u><i>Votre identifiant : </u></i></h2> <h2 align="center"><input type="text" name="user" ></h2><br/>
      <h2 align="center"><u><i>Votre mot de passe : </u></i></h2> <h2 align="center"><input type="password" name="passwd" ></h2><br/>

      <h2 align="center"><input type="submit" value="connexion"></h2><br/>

      <h2 align="center"><input type="reset" value="Annuler"></h2><br/>

      <h3 align="center"><a href="PageInscription.php">Inscription</a></h3>
    </form>

  </body>
</html>
```

C. ANNEXE 4 : CODE SOURCE PageConnexion.php

```
<h3 align="right"><a href="PageHtml1.html">Deconnexion</a></h3>
<h3 align="right"><a href="PageElectromenager.php">Electromenager</a></h3>
<h3 align="right"><a href="PageBricolage.php">Bricolage</a></h3>
<h3 align="right"><a href="PageJardin.php">Jardin</a></h3>
<h3 align="right"><a href="PageMenage.php">Menage</a></h3>

<?php

$host="127.0.0.1";
$user="root";
$password = "";
$database="mydb";

$cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
$conect= new PDO($cible, $user, $password);

//mysql_connect($host, $user,$password, $database) or die("erreur de connexion au serveur");
//mysql_select_db($database) or die("erreur de connexion a la base de donnees");

try
{
    $conect;
}
catch (Exception $a)
{
    die('Erreur :'.$a->getMessage());
}
```

```
//on récupère le contenu de la table inventaire
$reponse = $connect->query('SELECT * FROM digeq');

//on affiche chaque entrée une à une
while ($donnees = $reponse->fetch())
{
    ?>

<meta http-equiv="refresh" content="10">
<table border="1" cellpadding="10" cellspacing="1" width="100%">
    <tr>
        <th width="20%" align="center">Nom</th>
        <th width="10%" align="center">Reference</th>
        <th width="10%" align="left">Marque</th>
        <th width="10%" align="left">Categorie</th>
    </tr>
    <tr>
        <td align="center" valign="top"><?php echo $donnees['Nom']; ?></td>
        <td align="center" valign="top"><?php echo $donnees['Reference']; ?></td>
        <td align="left" valign="top"><?php echo $donnees['Marque']; ?></td>
        <td align="left" valign="top"><?php echo $donnees['Categorie']; ?></td>
    </tr>
</table>

<?php
}
$reponse->closeCursor(); //termine le traitement de la requête
```

D. ANNEXE 5 : CODE SOURCE PageInscription.php

```
<?php
if(empty($_POST['identifiant']))
{
    echo "remplissez le formulaire !";
}

elseif(!empty($_POST['identifiant'])) //il faut vérifier si le formulaire a été rempli
{

    $identifiant=$_POST['identifiant'];
    $mdp=$_POST['mdp'];

    // D'abord, je me connecte à la base de données.
    $host="127.0.0.1";
    $user="root";
    $passwd = "";
    $database="mydb";

    $Cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
    $connect= new PDO($Cible, $user, $passwd);

    $sql = $connect->query("INSERT INTO agent VALUES('', '$identifiant', '$mdp')");

    try
    {
        $connect;
        $mdp = sha1($mdp);//crypter les mot des passe
    }
}
```

```
catch (Exception $a)
{
    die('Erreur : '.$a->getMessage());
}

if ($connect == true)
{
    $sql;
    //$connect->exec($sql); // exécute la requête sql
    if ( $sql == true)
    {
        echo "Inscription terminée !";
    }
}

}

?>
```

```
<html>
<head>
    <link rel="stylesheet" href="style.css" />
</head>
<body>
    <p><strong><u><font color="teal"><h1 align="center"><t>Digeq Entreprise</t></h1></font></u></strong></p><hr/>

    <form method="post" action="PageInscription.php">
    <h2 align="center"><label>Identifiant: <input type="text" name="identifiant"/></label></h2><br/>
    <h2 align="center"><label>Mot de passe: <input type="password" name="mdp"/></label></h2><br/>

    <h2 align="center"><input type="submit" value="M'inscrire"/></h2>
    </form>
</body>
</html>
```


E. ANNEXE 6 : CODE SOURCE PageElectromennager.php

```
<?php

$host="127.0.0.1";
$user="root";
$password = "";
$dbase="mydb";

$cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
$conn = new PDO($cible, $user, $password);

//mysql_connect($host, $user,$password, $dbase) or die("erreur de connexion au serveur");
//mysql_select_db($dbase) or die("erreur de connexion a la base de donnees");

try
{
    $conn;
}
catch (Exception $a)
{
    die('Erreur : '.$a->getMessage());
}

//on récupère le contenu de la table inventaire
$reponse = $conn->query('SELECT Nom,Reference,Marque FROM digeq WHERE Categorie = Electromennager');
//on affiche chaque entrée une à une
while ($donnees = $reponse->fetch())
{
    ?>

<meta http-equiv="refresh" content="10">
<table border="1" cellpadding="10" cellspacing="1" width="100%">
    <tr>
        <th width="20%" align="center">Nom</th>
        <th width="10%" align="center">Reference</th>
        <th width="10%" align="left">Marque</th>

    </tr>
    <tr>
        <td align="center" valign="top"><?php echo $donnees['nom']; ?></td>
        <td align="center" valign="top"> <?php echo $donnees['reference']; ?></td>
        <td align="left" valign="top"><?php echo $donnees['marque']; ?></td>

    </tr>
</table>

<?php
}
$reponse->closeCursor(); //termine le traitement de la requête

?>
```

F. ANNEXE 7 : CODE SOURCE PageBricolage.php

```
<?php

$host="127.0.0.1";
$user="root";
$passwd = "";
$database="mydb";

$Cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
$conect= new PDO($Cible, $user, $passwd);
//mysql_connect($host, $user,$passwd, $database) or die("erreur de connexion au serveur");
//mysql_select_db($database) or die("erreur de connexion a la base de donnees");

try
{
    $conect;
}
catch (Exception $a)
{
    die('Erreur : '.$a->getMessage());
}

//on récupère le contenu de la table inventaire
$reponse = $conect->query('SELECT Nom,Reference,Marque FROM digeq WHERE Categorie = Bricolage');

//on affiche chaque entrée une à une
while ($donnees = $reponse->fetch())
{
    ?>
    |
    <meta http-equiv="refresh" content="10">
    <table border="1" cellpadding="10" cellspacing="1" width="100%">
        <tr>
            <th width="20%" align="center">Nom</th>
            <th width="10%" align="center">Reference</th>
            <th width="10%" align="left">Marque</th>

        </tr>
        <tr>
            <td align="center" valign="top"><?php echo $donnees['nom']; ?></td>
            <td align="center" valign="top"> <?php echo $donnees['reference']; ?></td>
            <td align="left" valign="top"><?php echo $donnees['marque']; ?></td>

        </tr>
    </table>

    <?php
    }
    $reponse->closeCursor(); //termine le traitement de la requête

    ?>
```

G. ANNEXE 8 : CODE SOURCE PageJardin.php

```
<?php

$host="127.0.0.1";
$user="root";
$passwd = "";
$database="mydb";

$Cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
$conect= new PDO($Cible, $user, $passwd);
//mysql_connect($host, $user,$passwd, $database) or die("erreur de connexion au serveur");
//mysql_select_db($database) or die("erreur de connexion a la base de donnees");

try
{
    $conect;
}
catch (Exception $a)
{
    die('Erreur :'.$a->getMessage());
}

//on récupère le contenu de la table inventaire
$reponse = $conect->query('SELECT Nom,Reference,Marque FROM digeq WHERE Categorie = Jardin');

//on affiche chaque entrée une à une
while ($donnees = $reponse->fetch())
{
    ?>
    <meta http-equiv="refresh" content="10">
    <table border="1" cellpadding="10" cellspacing="1" width="100%">
        <tr>
            <th width="20%" align="center">Nom</th>
            <th width="10%" align="center">Reference</th>
            <th width="10%" align="left">Marque</th>
        </tr>
        <tr>
            <td align="center" valign="top"><?php echo $donnees['nom']; ?></td>
            <td align="center" valign="top"> <?php echo $donnees['reference']; ?></td>
            <td align="left" valign="top"><?php echo $donnees['marque']; ?></td>
        </tr>
    </table>

    <?php
    }
    $reponse->closeCursor(); //termine le traitement de la requête

    ?>
```

H. ANNEXE 9 : CODE SOURCE PageMenage.php

```
<?php

$host="127.0.0.1";
$user="root";
$password = "";
$database="mydb";

$cible = 'mysql:host='.$host.';dbname=mydb;charset=utf8';
$conect= new PDO($cible, $user, $password);
//mysql_connect($host, $user,$password, $database) or die("erreur de connexion au serveur");
//mysql_select_db($database) or die("erreur de connexion a la base de donnees");

try
{
    $conect;
}
catch (Exception $a)
{
    die('Erreur :'.$a->getMessage());
}

//on récupère le contenu de la table inventaire
$reponse = $conect->query('SELECT Nom,Reference,Marque FROM digeq WHERE Categorie = Menage');

//on affiche chaque entrée une à une
while ($donnees = $reponse->fetch())
{
    ?>
    |
    <meta http-equiv="refresh" content="10">
    <table border="1" cellpadding="10" cellspacing="1" width="100%">
        <tr>
            <th width="20%" align="center">Nom</th>
            <th width="10%" align="center">Reference</th>
            <th width="10%" align="left">Marque</th>

        </tr>
        <tr>
            <td align="center" valign="top"><?php echo $donnees['nom']; ?></td>
            <td align="center" valign="top"> <?php echo $donnees['reference']; ?></td>
            <td align="left" valign="top"><?php echo $donnees['marque']; ?></td>

        </tr>
    </table>

    <?php
    }
    $reponse->closeCursor(); //termine le traitement de la requête

    ?>
```

