

LA BIBLIOTHEQUE

ISSET BIZERTE

DEPARTEMENT

TI

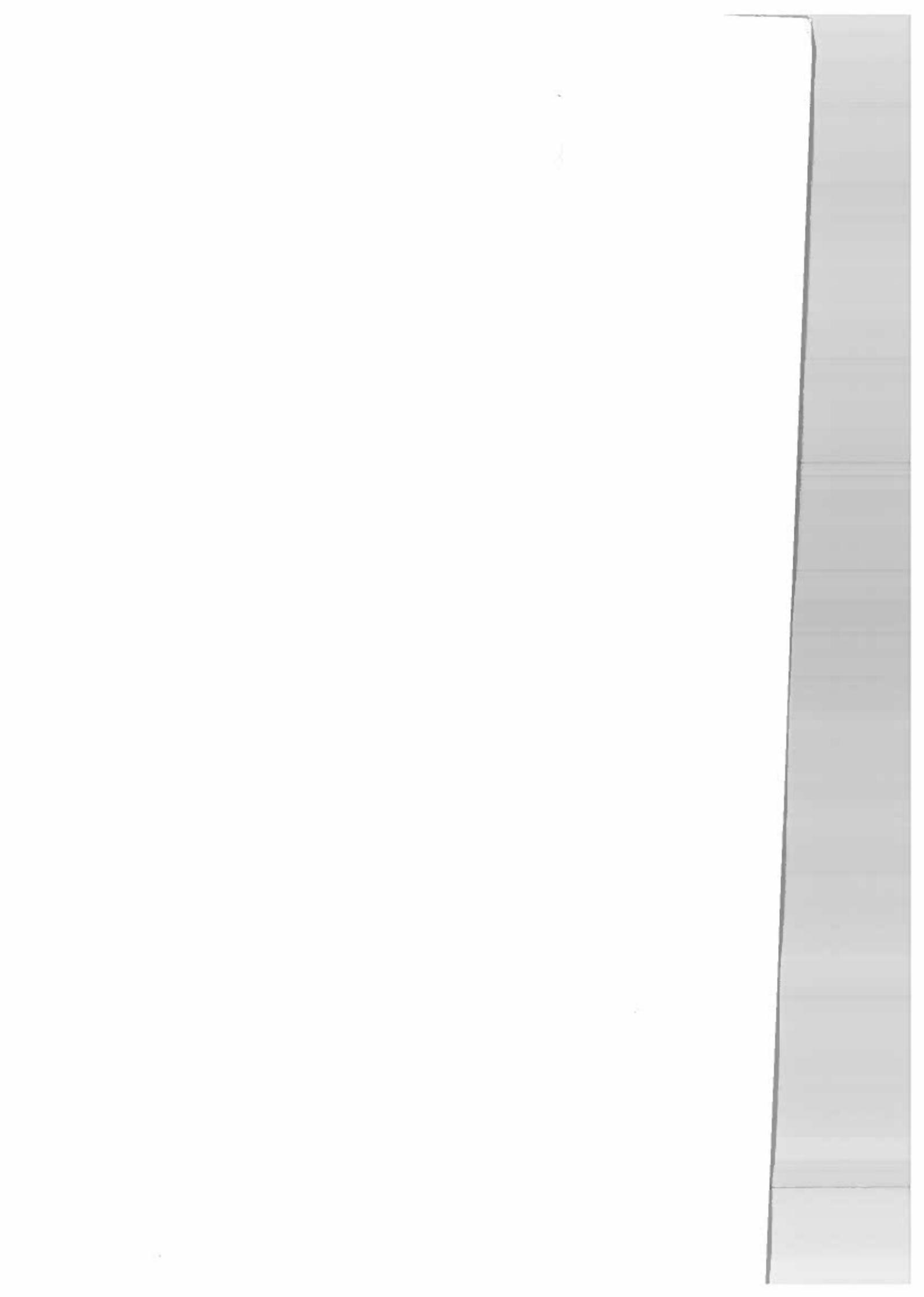
EXAMENS JUIN

2EME

SEM

ANNEE UNIVERSITAIRE

2020/2021



<b>Examen</b>	<b>Circuits Numériques</b>	
	Classe : SEM 2.1	Nb pages : 2
Documents : non autorisés	Enseignante : M <sup>me</sup> Wided Maghraoui	Durée : 1h30

**Exercice n°1 :**

1. Donner les équations des entrées des bascules D d'un registre à décalage circulaire à droite sur 4 bits :  $Q_3 Q_2 Q_1 Q_0$ . (1.5 pt)

2. Donner les équations des entrées des bascules D d'un registre à décalage circulaire à gauche sur 4 bits :  $Q_3 Q_2 Q_1 Q_0$ . (1.5pt)

3. Soit le compteur dit "rampant" dont les états sont :

**0000 → 0001 → 0011 → 0111 → 1111 → 1110 → 1100 → 1000 → 0000.**

Donner les équations des entrées des bascules D d'un circuit réalisant ce compteur. (2.5pt)

4. Donner les équations générales ainsi que le logigramme du circuit qui remplit les 3 fonctions précédentes. (3 pt)

**Exercice n°2 :**

1. Dessiner les formes d'onde demandées pour donner suite à l'analyse de la figure (1)

(Utiliser la page suivante pour répondre ). (2.5pt)

2. Quels sont le type et le modulo de ce compteur ? (1pt)

3. Modifier le schéma de la figure (1) pour avoir un décompteur (1 pt)

4. A partir des questions précédentes concevoir un compteur/décompteur (2pt)

5. Modifier le schéma de la figure (1) pour avoir un compteur réalisant le cycle suivant :

**5 – 6 – 7 – 8 – 9 – 5.** (2pt)

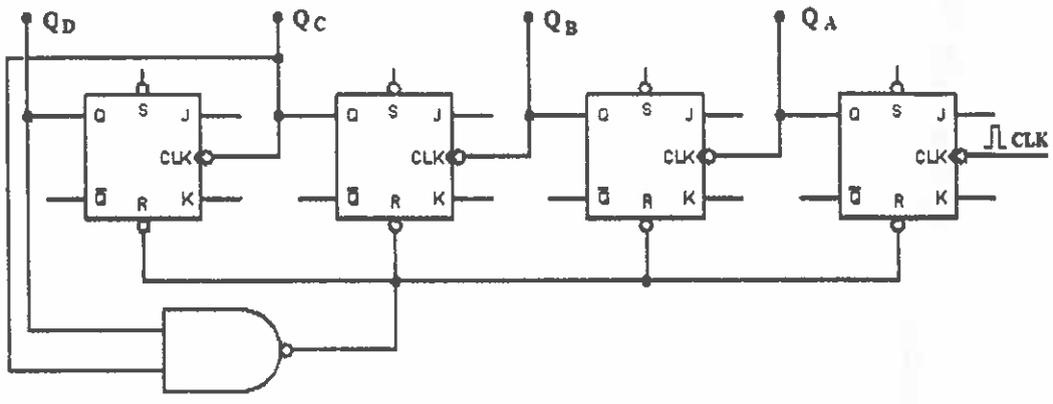
6. Réaliser un décompteur synchrone modulo 10 à base des bascules JK. (3pt)



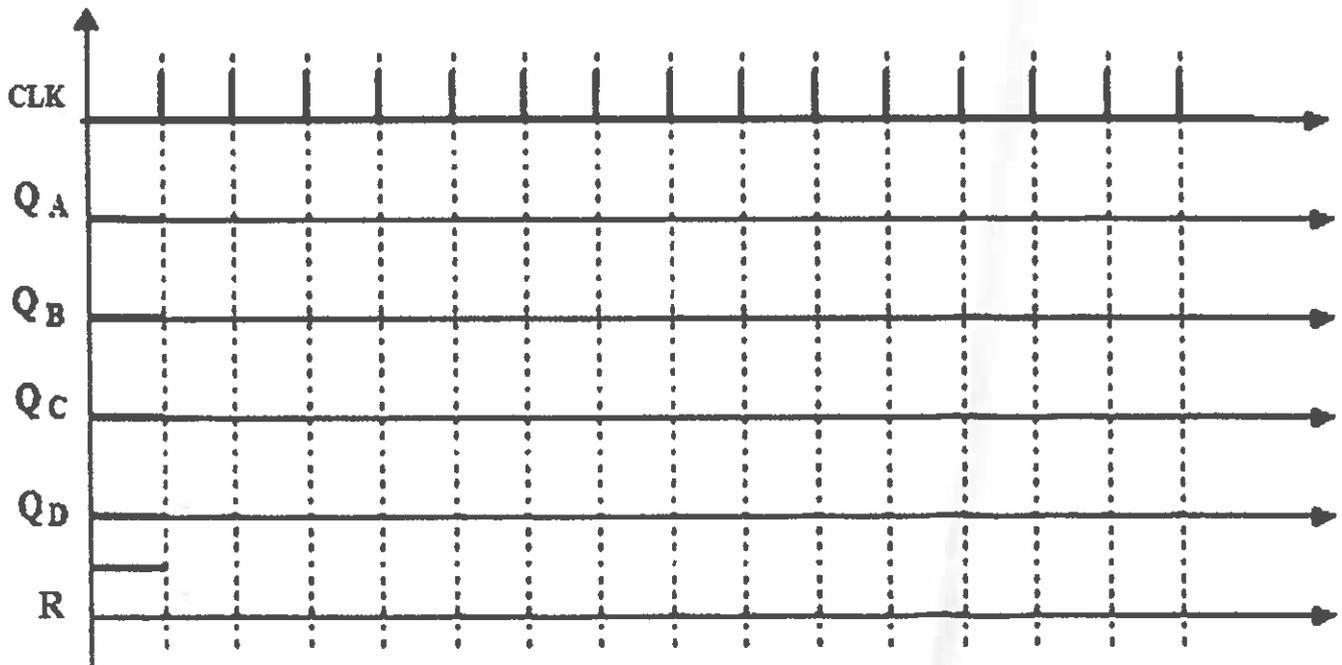
<b>Examen</b>	<b>Circuits Numérique</b>	
	Classe : SEM 2.1	Nb pages : 2
Documents : non autorisés	Enseignante : M <sup>me</sup> Wided Maghraoui	Durée : 1h30

Nom : ..... Prénom : ..... Classe : ..... N° C.I.N : .....

----- ✂ -----



*Figure (1)*





### Examen

<b>Matière</b>	: Développement mobile	<b>Classe (s)</b>	: SEM2
<b>Enseignant (s)</b>	: Mme Mounira Zouaghi	<b>Durée</b>	: 01h30
<b>Date</b>	: Juin 2021	<b>Documents</b>	: non autorisés
<b>Nbre pages</b>	: 9 pages	<b>Calculatrice</b>	: non autorisée

**Nom** : ..... **Prénom** .....

#### Exercice 1: ListView (5,25pts)

Soit la l'activité MyListViewActivity.java qui affiche la liste d'objets



1. Donner les étapes de création d'une l'actictivité ci-dessus en bref. (2pts)

Étape1:.....  
 .....

# NE RIEN ECRIRE ICI

Etape2:.....  
.....  
.....

Etape3:.....  
.....

Etape4:.....  
.....

2. En quoi consiste le rôle de la classe Adapter pour une Listview (1pts)

.....  
.....  
.....

3. Compléter le code suivant dans MyListActivity.java (2,25pts)

```
public class MyListActivity extends AppCompatActivity {
    Integer[] photos={R.drawable.a_avator,R.drawable.b_avator,R.drawable.c_avato
r,R.drawable.d_avator,R.drawable.e_avator};
    String[] title={"Formation PHP","Formation J2EE","Formation
Android","Formation C#","HTML/CSS »};

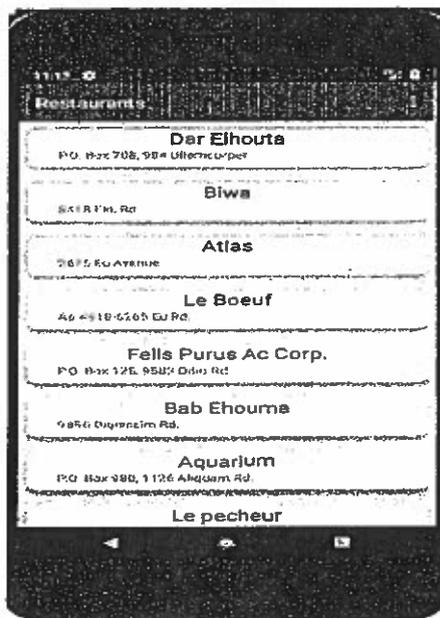
    ListView lv;
    ImageAdapter imga;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        lv=..... (R.id.lv);
        imga=new..... (this,photos,title);

        lv.....(imga);
    }
}
```

# NE RIEN ECRIRE ICI

## Exercice 2 : Affichage de la liste de restaurants (7,5pts)



Soit la liste de restaurants affichée sous forme d'un RecyclerView. Chaque ligne contient le nom de restaurant et l'adresse sous forme de TextView.

1) Quelle sont les différences entre un RecyclerView et un ListView.(1,5pt)

.....

.....

.....

1) On prépare aussi la classe Restaurant.java ayant comme attribut Nom,Adresse,Etoiles et les getter. Déclarer cette classe.(1,5pt)

**NE RIEN ECRIRE ICI**

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

2) Pour Réaliser le RecyclerView on doit préparer la classe **CustomAdapter.java** qui hérite de la classe **RecyclerView.Adapter**

a) La première étape sera d'implémenter la classe **MyViewHolder**. Compléter cette classe. **.(1,5pt)**

```
public class MyViewHolder extends .....  
{ TextView textViewName;  
  TextView textViewadresse;  
  public MyViewHolder(View itemView) {  
    super(itemView);  
    this.textViewName=.....findViewById(R.id.name);  
    this.textViewadresse=.....findViewById(R.id.adresse);  
  }  
}
```

# NE RIEN ECRIRE ICI

- b) La deuxième étape est d'implémenter les méthodes de la classe CustomAdapter.java. La première méthode est **onCreateViewHolder**. Compléter la méthode sachant que le layout des lignes est **card\_layout.xml** (1,5pt)

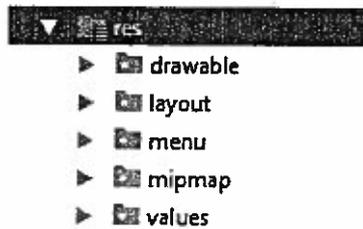
```
@Override
public ..... onCreateViewHolder(ViewGroup parent, int
viewType) {
    View view=
    LayoutInflater.from(parent.getContext()).inflate(R.....
    .....,parent,false);
    return new .....(view);
}
```

- c) La method **onBindViewHolder** permet de remplir chaque ligne comme suit. Compléter la méthode. (1,5pt)  
ArrayList<Restaurant> restaurants;

```
@Override
public void onBindViewHolder(MyViewHolder holder, int position) {
    TextView textView=.....;
    TextView textView2=holder. ....;
    textView.setText(restaurants.get(position).getNom());
    textView2.setText(restaurants.get(position).getAdresse());
}
```

**NE RIEN ECRIRE ICI**

**Exercice 3: Layouts et Ressources (7,25pts)**



1. Donnez que peut contenir chaque dossier ressource : type de fichier et rôle? (2,5pts)

Drawble : .....

Layout : .....

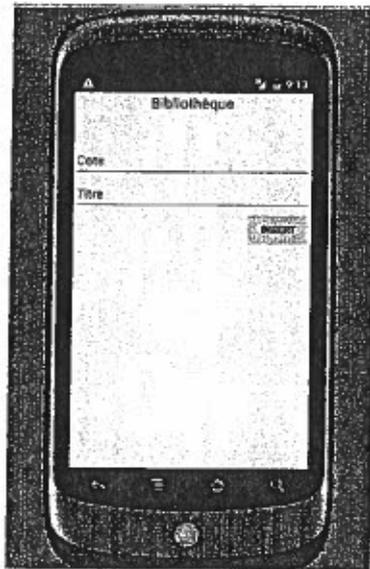
Menu : .....

Mipmap : .....

values : .....

2. Soit l'activité suivante :

**NE RIEN ECRIRE ICI**



1. Proposer un code xml pour cette interface en utilisant un LinearLayout. Toutes les chaînes de caractères sont définies dans le fichier string.xml (2pts)

```
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  android:layout_width="....."
  android:layout_height="....."
  android:.....="vertical">
```

.....

.....

.....

.....

.....

.....



**3. Le click sur le bouton fait appel à une fonction `insert_bd`. (0,75pts)**

Dans cette fonction, on affiche un toast « l'insertion est effectuée avec succès » et on passe à une nouvelle activité `MyListActivity.java`.

Ecrire le code de cette fonction.

.....

.....

.....

.....

.....

.....

.....

.....



## Examen final

<b>Matière</b>	: Conception SysMI	<b>Classe (s)</b>	: SEM 2
<b>Enseignant (s)</b>	: B.Zemzem	<b>Durée</b>	: 01 h 30
<b>Date</b>	: Juin 2021	<b>Documents</b>	: non autorisés
<b>Nbre pages</b>	: 2		

### Exercice n°1 (6Pts+6Pts)

Il s'agit d'informatiser l'étude des demandes de financement d'un projet.

Un client doit déposer sa demande de financement en présentant toutes les pièces jointes demandées. Il peut suivre via son espace l'avancement de l'étude de cette demande et la réponse.

Une fois la commission a accepté le financement, le dossier de projet est donc soumis simultanément au responsable de l'agence concernée et au siège, qui doit donner un avis très rapidement.

Si l'avis du siège est défavorable, le projet est abandonné. Les différents acteurs concernés seront prévenus (client et agence).

Si le siège accepte de co-financer le projet, on attend la réponse de l'agence pour décider de poursuivre les procédures de financement (si c'est un accord favorable) ou de rejeter le dossier (si c'est un accord défavorable).

Quand toutes les réponses sont positives, un dossier de financement définitif est établi puis le projet est lancé sinon le refus doit être présenté avec les justificatifs.

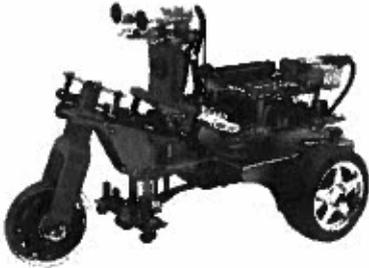
#### Travail à faire

*1) Donnez le diagramme de cas d'utilisation*

*2) Donnez le diagramme d'activités correspondant au processus de financement d'un projet.*

## **Exercice n°2 (8Pts)**

*L'objectif est de réaliser un robot tricyclé et de développer une application qui l'exploite.*



Ce robot doit répondre aux besoins suivants :

1. Pouvoir des déplacer dans toutes les directions. Le déplacement sera téléguidé (en suivant les ordres de l'utilisateur via une page web) ou bien tout seul pour balayer un espace. Dans le dernier cas le robot doit être capable de détecter les obstacles et de tourner à droite (ou bien à gauche)
2. Prendre des photos (toutes les secondes) et les envoyer à un serveur de données
3. Etre autonome en énergie pendant 30mn.

### **Travail à faire**

*Donnez le diagramme d'exigence pour ce système*

**Bon Travail**



Ne rien écrire dans cette zone

---

**✍ Exercice 1 (3 points)**

Cocher la(les) bonne(s) réponse(s).

**Q1. A quoi peuvent servir les sessions ?**

- a) Sauvegarder des informations
- b) Se connecter à une base de données
- c) A faire référence à une session en cours

**Q2. A quoi sert la clause 'where' en SQL ?**

- a) Trier les données en sortie
- b) Rechercher certains enregistrements parmi dans une table ou un ensemble de tables
- c) Regrouper des enregistrements en sortie

**Q3. Quelle sont les expressions valides ?**

- a) `<?php if(isset($nom)) echo $nom ?>`
- b) `<?= if(isset($nom)) echo $nom ?>`
- c) `<?php echo $nom ?>`
- d) `<?= $nom ?>`

**Q4. En php, comment instancier un objet à partir d'une classe ?**

- a) `instanceof`
- b) `->`
- c) `New`
- d) `this`

**Q5. Comment créer une session qui permettra d'utiliser des variables de session ?**

- a) `session_start()`
- b) `start_session()`
- c) `init_session()`

**Q6. Comment détruire une session ?**

- a) `session_destroy()`
- b) `destroy_session()`
- c) `delete_session()`

Ne rien écrire dans cette zone

---

### Etude de cas (17 points)

---

Le document suivant permet un simple insérer, sélectionner, modifier et supprimer (CRUD) d'images en utilisant le PDO.

Une image sera uploadée, insérée et modifiée avec MySQL.

#### **La base de données**

---

La BD qu'on utilisera est nommée "inscription".

On va travailler sur la machine locale sans modifier la configuration par défaut de l'environnement comme dans les ateliers. La table "users" est la suivante :

users (userId : integer auto-increment, userName, userProfession, userInformations : chaînes de caractères)

#### **connexion.php**

---

1. Compléter le code suivant permettant de faire la connexion avec notre BD en utilisant PDO.

```
<?php
try {
    ..... =
    ..... ("mysql:host=.....;dbname=.....",
    .....);
} catch(PDOException $e){
    echo $e->getMessage();
}
```

#### **ajoutNouveau.php**

---

Soit le formulaire HTML suivant (créé avec Bootstrap) permettant l'ajout d'un nouvel utilisateur :

```
<form method="post" action="....."
class="form-signin">
    <h2 class="form-signin-heading">Ajouter un utilisateur</h2><hr />
```

Ne rien écrire dans cette zone

```
<div class="form-group">
    <span class="text-danger"><?php if(isset($nomErr)) echo
$nomErr;?></span>
    <input type="text" class="form-control" name="uname" value="<?php
if(isset($uname)) echo $uname?>" />
</div>
<div class="form-group">
    <span class="text-danger"><?php if(isset($professionErr)) echo
$professionErr;?></span>
    <input type="text" class="form-control" name="uprofession"
value="<?php if(isset($uprofession)) echo $uprofession?>" />
</div>
<div class="form-group">
    <span class="text-danger"><?php if(isset($professionErr)) echo
$professionErr;?></span>
    <textarea class="form-control" name="uinformations" id="uinformations"
cols="30" rows="5" placeholder="Saisir Informations"><?php
if(isset($uinformations)) echo $uinformations ?></textarea>
</div>
<div class="form-group">
    <button type="submit" class="btn btn-primary">Ajouter</button>
</div>
</form>
```

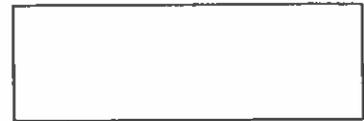
Nom : ..... Prénom : ..... Classe : .....

N° C.I.N. : [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] Salle : ..... N° place.....

Signature de l'étudiant

Epreuve de : Développement côté serveur

Date : Juin 2021



Le résultat obtenu par ce code est le suivant :

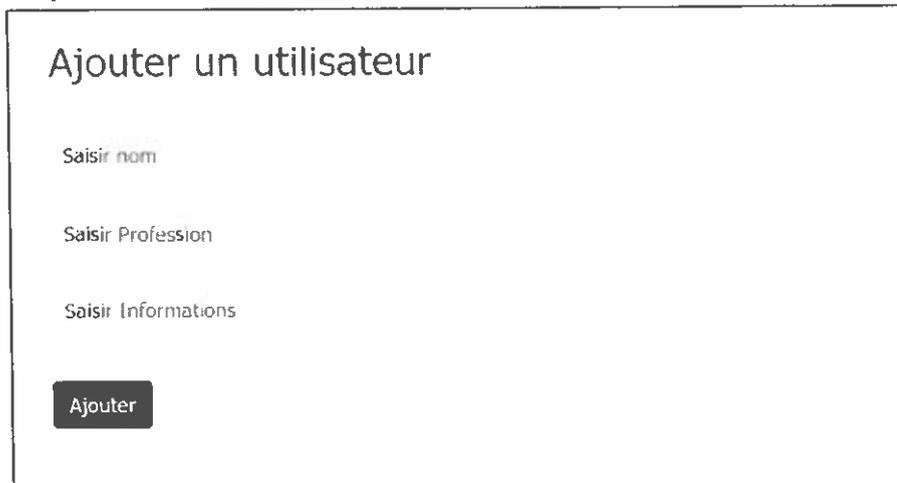


Figure 1: ajoutNouveau.php

On souhaite insérer dans la page ajoutNouveau.php le code PHP suivant, permettant d'insérer un utilisateur. Des tests seront faits sur les champs, en cas d'erreur un message sera affiché avec un design Bootstrap.

2. Compléter le code suivant permettant l'insertion d'un nouvel utilisateur à la BD en sachant que :

- Le nom est obligatoire et ne contient que des lettres et des espaces.
- Le nom ne doit pas exister dans la base de données (on suppose qu'il existe une fonction exist\_user(\$uname) qui permet de le vérifier).
- Les champs profession et informations sont obligatoires.

```
<?php
if (!..... (.....)) {
    //Récupération du contenu du formulaire
    $uname = ..... ;
    $uprofession = ..... ;
    $uinformations = ..... ;
```

Ne rien écrire dans cette zone

```
$valid=.....; // true si le formulaire est valide (tous
les champs sont remplis)

// Test username

if (..... ($uname)) {

    $.....= "Le nom est obligatoire";

    $valid=.....;

} else if (!preg_match("/^[a-zA-Z ]*$/", $uname)) { //la vérification
est faite en utilisant la notion d'expression régulière

    $.....= "Seuls les lettres et l'espace sont permis";

    $valid=.....;

} else if (exist_user($uname)) {

    $.....= "Username existe déjà";

    $.....=.....;

}

// Test user profession

if (..... ($uprofession)) {

    $professionErr = "Profession obligatoire";

    $valid=.....;

}

// Test user informations

if (..... ($.....)) {

    $informationsErr= "Informations obligatoires";

    $valid=.....;

}
```

Ne rien écrire dans cette zone

```
//Si tout est valide
if (.....) {
    // Insertion dans la Base de données
    require '.....';
    $req = "INSERT .....";
    .....";
    $rep = $con->..... ($.....);
    if ($rep) {
        echo '<div class="alert alert-success" role="alert">';
        echo 'Insertion effectuée avec succès';
        echo '</div>';
    } else {
        echo '<div class="alert alert-danger" role="alert">';
        echo 'Insertion non effectuée';
        echo '</div>';
    }
} } ?>
```

## index.php

L'affichage des utilisateurs dans la page d'accueil sera fait aussi avec bootstrap, en utilisant un div avec la class="row".

Ne rien écrire dans cette zone

The screenshot shows a web page titled "Liste des utilisateurs". At the top left, there is a button labeled "Add User". Below this, three user profiles are displayed in a table-like format. Each profile includes a name, a profession, a detailed biography, a user ID, and two buttons labeled "Edit" and "Delete".

userName	userProfession	userInformations	userID
Morgan Freeman	Actor	Est un acteur, réalisateur et producteur de cinéma américain, né le 1er juin 1937 à Memphis (Tennessee).	# 2
Michael Jackson	Singer	Est un auteur-compositeur-interprète, danseur-chorégraphe et acteur américain né le 29 août 1958 à Gary (Indiana)	# 3
Lionel Messi	Footballeur	né le 24 juin 1987 à Rosario en Argentine, est un footballeur international argentin évoluant au poste attaquant au FC Barcelone.	# 4

Figure 2: index.php

3. Compléter le code PHP suivant permettant d'afficher les profils des utilisateurs à partir de la BD.

```
<div class="container">
  <h2 class="mt-5">Liste des utilisateurs <a class="btn btn-success"
href="addUser.php">Add User</a></h2>
  <hr>
  <div class="row">
  <?php
  // Sélection de la liste des utilisateurs
  $sql = 'SELECT .....';
  $reponse = $con->..... ();
  $users = $reponse->..... (PDO::FETCH_ASSOC);
  // Affichage de la liste des utilisateurs
```

Ne rien écrire dans cette zone

```
</ul>
</div></div>
<?php endforeach; ?>
</div> </div>
```

### delete.php

4. Compléter le code php suivant permettant de supprimer un utilisateur de la BD.

```
<?php
require ".....php";

$id = $_GET['id'];

// Suppression de l'utilisateur

$sql = "DELETE .....";

$response = $.....->..... ($sql);

if (.....) {

    echo "Suppression effectuée avec succès";

    ..... ("location:index.php");

} else {

    echo "Echec de la suppression";

}

?>
```

Nom : ..... Prénom : ..... Classe : .....

N° C.I.N. :         Salle : ..... N° place .....

Signature de l'étudiant

Epreuve de : Développement côté serveur  
Date : Juin 2021

```
<?php foreach ($..... as $.....):?>

<div class="elem">

  <div class="card" style="width: 18rem;">

    <div class="card-body">

      <h5 class="card-title"><strong><?= ..... ?></h5>

      <p class="card-text"><em><?= ..... ?></em></p>

      <p class="card-text"><em><?= ..... ?></em></p>

    </div>

    <ul class="list-group list-group-flush">

      <li class="list-group-item"> #<?= $user['.....'] ?></li>

      <li class="list-group-item">

        <div class="boutons">

          <a class="btn btn-info" href="update.php?id=<?= .....?>"
onclick="return confirm('\Etes vous sûr de vouloir modifier cet
utilisateur ?\')">Edit</a>

          echo '<a class="btn btn-danger" href="delete.php?id=<?=
.....?>" onclick="return confirm('\Etes vous sûr de vouloir
supprimer cet utilisateur ?\')"> Delete</a>

        </div>

      </li>

    </ul>

  </div>

</div>
```



## update.php

Le formulaire d'Édition du profil d'un utilisateur est aussi simple que celui de l'ajout.

Mettre à jour des informations d'un utilisateur

Nom:  
Morgan Freeman

Email:  
Actor

Informations:  
Est un acteur, réalisateur et producteur de cinéma américain, né le 1er juin 1937 à Memphis (Tennessee).

Modifier Annuler

Figure 3: update.php

Lorsqu'un le profil d'un utilisateur est modifié, on doit chercher cet enregistrement bien précis dans la BD.

5. Compléter le code PHP suivant permettant de modifier le profil d'un utilisateur dans la BD.

```
<?php
require '.....php';
$id = $_..... ['id'];
if (!empty(.....)) {
//Récupération du contenu du formulaire de mise à jour
$username = $_POST['.....'];
$uprofession = $_..... ['uprofession'];
$uinformations = $_..... ['uinformations'];
```

Ne rien écrire dans cette zone

```
// Modifier les informations de l'utilisateur dans la BD
$sql = "UPDATE .....
.....";

$response = $con->..... (.....);
if ($response) {
    echo '<div class="alert alert-success" role="alert">';
    echo "Modification effectuée avec succès";
    echo '</div>';
} else {
    echo '<div class="alert alert-danger" role="alert">';
    echo ".....";
    echo '</div>';
} }

// Récupérer les infos de l'utilisateur afin de les afficher
// dans le formulaire
$req = "SELECT * FROM ..... WHERE userID = .....";
$rep = $con->..... ($req);
$user = $rep->..... (); ?>

<h2 class="mt-5">Mettre à jour des informations d'un utilisateur</h2>
<hr>
```



**Ne rien écrire dans cette zone**