

École Centrale des Logiciels Libres et de Télécommunications



Institut Supérieur de Technologie de l'Université de Bangui (IST/UB)



Premiers pas avec HTMX pour des applications monopages

**Enseignant :
Pr. SAMUEL OUYA**

Premiers pas avec HTMX pour des applications monopages

Prof Samuel OUYA

Objectif

Comment intégrer dans un code html Ajax afin de créer une application monopage qui appelle des applications derrière sans qu'on ne bouge de la page de son application principale

Avant, on utilisait des technologies lourdes telle que react mais aujourd'hui, on utilise **htmx** qui est léger

La bibliothèque [HTMX](#) permet d'ajouter des fonctionnalités de navigateur modernes sans utiliser JavaScript. Il vous donne accès aux **transitions CSS**, **AJAX**, **WebSockets** et **événements envoyés par le serveur** directement en HTML, en utilisant des attributs pour créer rapidement des interfaces utilisateur modernes. Serait-il possible de développer des applications Web côté client sans JavaScript ? Dans cette séquence, nous allons explorer toutes les fonctionnalités époustouflantes de HTMX. On va envoyer des requêtes Ajax, télécharger des fichiers, valider des entrées et à déclencher des transitions CSS sans écrire de code JavaScript.

Le créateur de HTMX dit qu'il tente de tirer parti de la puissance de HTML en utilisant le modèle original du Web pour créer des applications Web. Avec cette approche, les développeurs n'ont pas besoin d'écrire du JavaScript pour obtenir des fonctionnalités similaires. Au lieu de cela, ils utilisent des attributs HTML supplémentaires pour obtenir un contenu et des mises à jour dynamiques. Selon Gross, ce sont les motivations derrière htmx :

- Pourquoi ne devrait-on **<a>** et ne **<form>** peut-on que faire des requêtes HTTP ?
- Pourquoi seuls les événements **click** devraient-ils **submit** les déclencher ?
- Pourquoi seules les méthodes **GET** devraient-elles être disponibles ? **POST**

« En supprimant les contraintes arbitraires, htmx complète le HTML en tant qu'hypertexte », conclut-il.

Avec HTMX, les développeurs peuvent implémenter plusieurs fonctionnalités d'interface utilisateur et modèles UX avec un minimum de code HTML et de style, tels que la [barre de progression](#), le [chargement différé](#), le [défilement infini](#), la [validation en ligne](#), etc.

Configurer HTMX

Pour configurer HTMX, vous pouvez apparemment le charger via un CDN et l'ajouter à votre balise principale comme ceci ci-dessous :

```
<script src="https://unpkg.com/htmx.org@1.7.0"
integrity="sha384-EzBXYPt0/T6gxNp0nuPtLkmRpmDBbjg6WmCUZRLXBBwYYmwAUxzLSGej0ARHX0
Bo" crossorigin="anonymous"></script>
```

Exemple

Contenu de htmx/premier.html

```
<html>
<head>
</head>

<body>
<script src="https://unpkg.com/htmx.org@1.7.0"></script>
<div
  hx-get="http://localhost/bootstrap/tableauboot.html"
>EC2LT</div>
</body>
```

Résultat

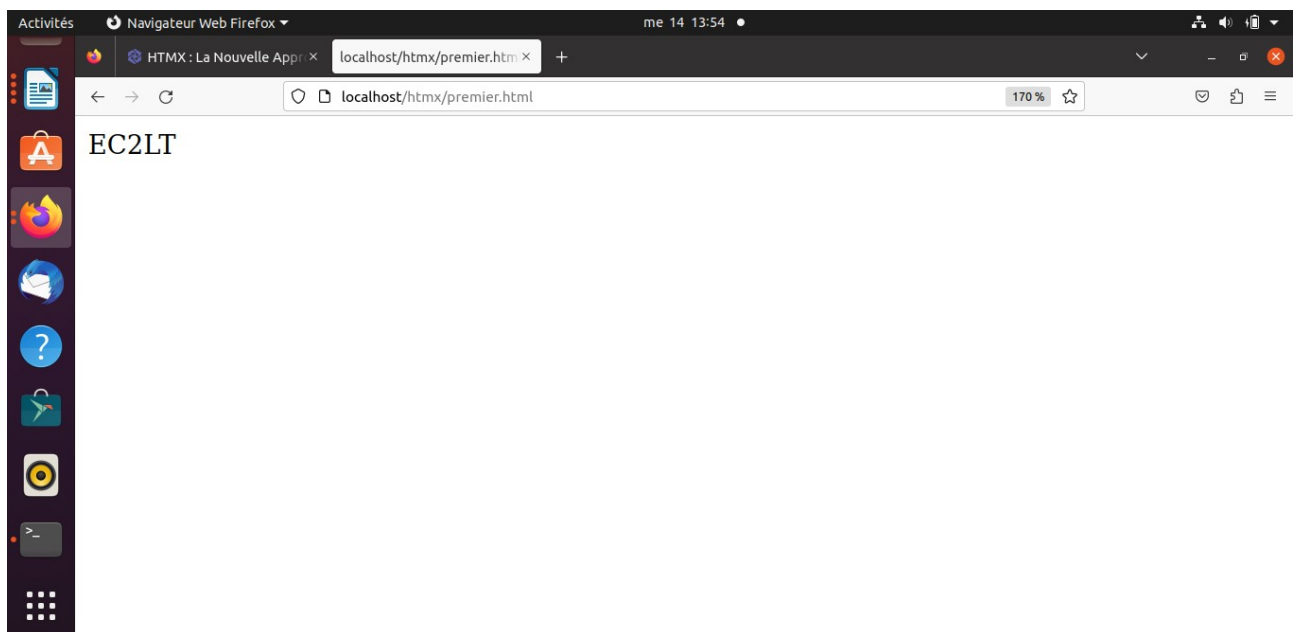


fig1 :



ID	Prenom	Nom	Numero de compte	Codeclient	Solde
1	Macky	Sall	1001	1111	14500000
2	Denis	Sassou	1002	2222	80000000
3	Faustin	Archange Touadera	1003	3333	700000000
4	Ali	Bongo Odimba	1004	4444	550000000
5	Paul	Biya	1005	5555	680000000

fig2 : Quand on clique sur EC2LT, on obtient le tableau fourni par le `tableauboot.html` sans voir l'url menant vers `tableauboot.html`

Une autre méthode consiste à télécharger le [fichier source htmx.min.js](#) et à l'ajouter dans le répertoire approprié de votre projet, puis à l'insérer si nécessaire avec une balise de script comme celle-ci :

```
<script src="/path/to/htmx.min.js"></script>
```

Vous pouvez également installer HTMX via **npm** comme ceci ci-dessous :

```
root@im:/var/www/html/htmx# npm install htmx.org
```

```
added 1 package in 3s
```

```
npm notice
```

```
npm notice New patch version of npm available! 9.6.4 -> 9.6.6
```

```
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.6.6
```

```
npm notice Run npm install -g npm@9.6.6 to update!
```

```
npm notice
```

```
root@im:/var/www/html/htmx#
```

```
root@im:/var/www/html/htmx/node_modules/htmx.org# cd dist
```

```
root@im:/var/www/html/htmx/node_modules/htmx.org/dist# ls
```

```
ext htmx.d.ts htmx.js htmx.min.js htmx.min.js.gz
```

```
root@im:/var/www/html/htmx/node_modules/htmx.org/dist# mkdir
```

```
/var/www/html/htmx/sourcehtmx
```

```
root@im:/var/www/html/htmx/node_modules/htmx.org/dist# cp htmx.min.js
```

```
/var/www/html/htmx/sourcehtmx/
```

On modifie la source de htmx dans `premier.html` et on obtient les mêmes résultats

```

<html>
<head>
</head>

<body>
<script src="sourcehtmlx/htmx.min.js"></script>
<div
  hx-get="http://localhost/bootstrap/tableauboot.html"
>EC2LT</div>
</body>

```

Envoi de requêtes Ajax à l'aide de HTML

HTMX propose un ensemble d'attributs qui vous permettent de faire des requêtes AJAX directement à partir de HTML.

- **hx-post**- envoie une requête POST à l'URL donnée.
- **hx-get**- envoie une requête GET à l'URL donnée.
- **hx-put**- envoie une requête PUT à l'URL donnée.
- **hx-patch**- envoie une requête PATCH à l'URL donnée.
- **hx-delete**- envoie une requête DELETE à l'URL donnée.

Chacun de ces attributs ci-dessus accepte une URL à laquelle envoyer une requête AJAX. Ainsi, chaque fois que l'élément est déclenché, il envoie le type de requête spécifié à l'URL donnée. Considérez l'exemple ci-dessous :

Déclencher des requêtes avec HTMX

L'événement "naturel" d'un élément lance automatiquement les requêtes Ajax par défaut. Par exemple, **onchange** déclencheurs d'événement **textarea**, **input** et **select**. Pendant que l'**onsubmit** événement se déclenche **form**. L' **click** événement déclenche toutes les autres requêtes. HTMX offre un "hx-trigger" unique si vous avez besoin de spécifier quel événement déclenchera la requête.

Interrogation avec l' htmx-trigger attribut

En utilisant l'attribut de déclencheur HTMX, nous pouvons également spécifier que l'élément interroge l'URL donnée toutes les **n** secondes au lieu d'attendre qu'un événement se produise.

Code de trio.html

```

<html>
<head>
</head>

```

```

<body>
<script src="sourcehtmx/htmx.min.js"></script>
<button class="btn" hx-get="http://localhost/bootstrap/navigation.html" hx-target="#result" >
EC2LT
</button>

<div id="result"></div>

</body>

</html>

```

Résultats

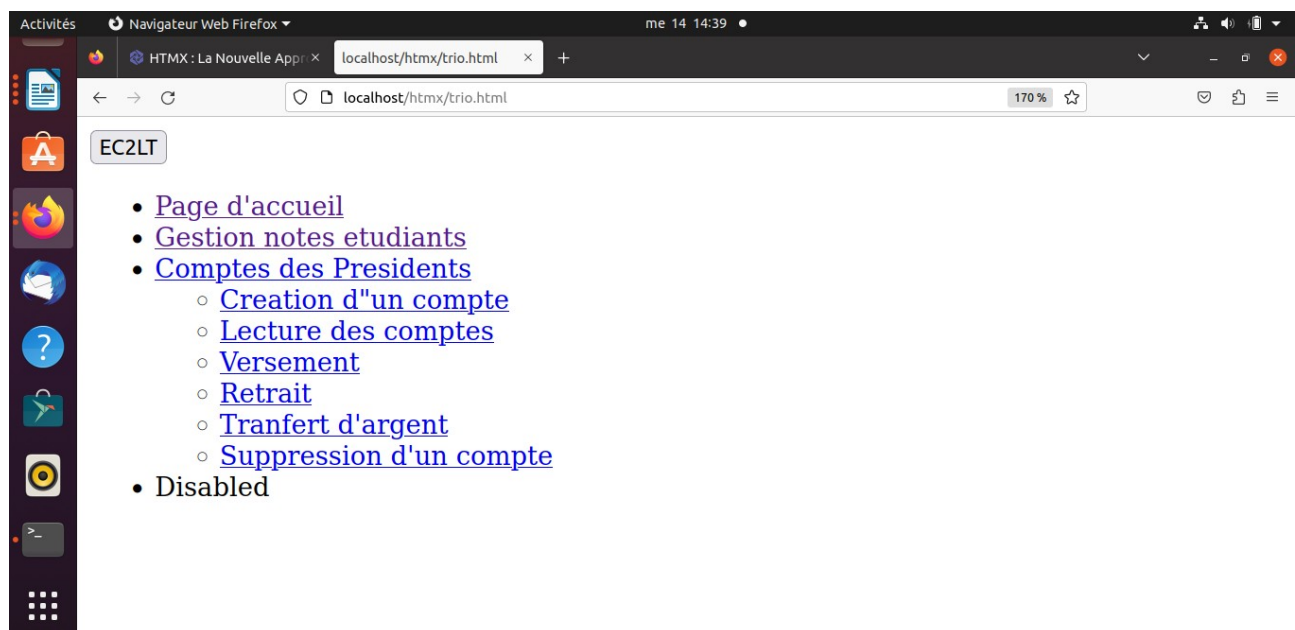


fig3 : appel à une page de navigation

Mise en œuvre

Mettons en place une application PHP CRUD avec PDO et requêtes préparées

tpnoir/comptepresi .php

```

?php
$choix=$_REQUEST['choix'];
$conn = new PDO('mysql:host=localhost;dbname=banquepresi1','bouki1',"passer");

```

```

function lecture($numcompte)
{
global $conn;

```

```

$sql="select * from client where numcompte=:numcompte";
$smt= $conn->prepare($sql);
$smt->bindParam(':numcompte',$numcompte);
$smt->execute();
if ($smt->rowCount()>0) {
$message="voici les infos demandees";
$result=['message'=>$message,'client'=>$smt->fetch()];

}
else {

    $message="desolé client introuvable";
    $result=['message'=>$message,'client'=>[]];
}

return( json_encode($result));

}

function lectures()
{
global $conn;

$sql="select * from client";
$smt= $conn->prepare($sql);
$smt->execute();
if ($smt->rowCount()>0) {
$message="voici les infos demandees";
$tab=$smt->fetchAll();
$result=['message'=>$message,'client'=>$tab];

    $chaine="<table border='1px'><tr> <td>Prenom</td> <td>Nom</td> <td>Numcompte</td>
<td>Codeclient</td> <td>Solde</td></tr>";
    foreach ( $tab as $line ) {
        $prenom=$line['prenom'];$nom=$line['nom'];$numcompte=$line['numcompte'];
        $codeclient=$line['codeclient'];$solde=$line['solde'];
        $chaine=$chaine."<tr> <td>$prenom</td> <td>$nom</td> <td>$numcompte</td>
<td>$codeclient</td> <td>$solde</td></tr>";
    }
    $chaine=$chaine."</table>";
echo $chaine;

}
else {

    $message="desolé client introuvable";
    $result=['message'=>$message,'client'=>[]];
}
return( json_encode($message));

```

```

}

function suppression($numcompte)
{
global $conn;

$sql="delete from client where numcompte=:numcompte";
$stmt= $conn->prepare($sql);
$stmt->bindParam(':numcompte',$numcompte);
$stmt->execute();
if ($stmt->rowCount()>0) {
$message="Compte supprime avec succes";
}
else {
$message="Compte introuvable";
}

return json_encode($message);
}

function creation($prenom,$nom,$numcompte,$codeclient,$solde)
{
global $conn;

$sql="insert into client(prenom,nom,numcompte,codeclient,solde) values
(:prenom,:nom,:numcompte,:codeclient,:solde)";
$stmt= $conn->prepare($sql);
$stmt->bindParam(':prenom',$prenom);
$stmt->bindParam(':nom',$nom);
$stmt->bindParam(':numcompte',$numcompte);
$stmt->bindParam(':codeclient',$codeclient);
$stmt->bindParam(':solde',$solde);
$stmt->execute();
$message="compte cree avec succes";
return(json_encode($message));
}

case "2":
    print_r(lectures());
    break;
case "1":
    $prenom=$_REQUEST["prenom"];
    $nom=$_REQUEST["nom"];
    $numcompte=$_REQUEST["numcompte"];
    $codeclient=$_REQUEST["codeclient"];
    $solde=$_REQUEST["solde"];
    print_r(creation($prenom,$nom,$numcompte,$codeclient,$solde));
    break;
case "3":
    $numcompte=$_REQUEST['numcompte'];
    $somme=$_REQUEST['somme'];
    print_r(versement($numcompte,$somme));
    break;

```



```

    case "4":
        $numcompte=$_REQUEST['numcompte'];
        print_r(suppression($numcompte));
        break;

    case "5":
        $numcompte=$_REQUEST['numcompte'];
        print_r(lecture($numcompte));
        break;

}

?>

```

Mettons nos interfaces [htmx](#)
htmx/creation.html

```

<html>
<head>
    <meta charset="UTF-8">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
    rel="stylesheet" integrity="sha384-
    1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
    crossorigin="anonymous">

</head>

<body>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>
<script src="sourcehtmx/htmx.min.js"></script>

<form hx-post="http://localhost/tpnoir/comptepresi.php">

<div class="row mb-3">
    <div class="col-3">
        <label for="prenom">Prenom</label>
    </div>
    <div class="col-6">
        <input id="prenom" name="prenom" type="text" class="form-control">
    </div>
</div>

<div class="row mb-3">
    <div class="col-3">
        <label for="nom">Nom</label>
    </div>
    <div class="col-6">
        <input id="nom" name="nom" type="text" class="form-control">
    </div>
</div>

```

```

</div>

<div class="row mb-3">
  <div class="col-3">
    <label for="numcompte">Numcompte</label>
  </div>
  <div class="col-6">
    <input id="numcompte" name="numcompte" type="text" class="form-control">
  </div>
</div>

div class="row mb-3">
  <div class="col-3">
    <label for="codeclient">Codeclient</label>
  </div>
  <div class="col-6">
    <input id="codeclient" name="codeclient" type="text" class="form-control">
  </div>
</div>

<div class="row mb-3">
  <div class="col-3">
    <label for="solde">Solde</label>
  </div>
  <div class="col-6">
    <input id="solde" name="solde" type="text" class="form-control">
  </div>
</div>
  <input type="hidden" id="choix" name="choix" value="1">
  <button type="submit">Sauvegarder</button>
</form>

</body>

</html>

```

code de htmx/lecture.html

```

<html>
<head>
  <meta charset="UTF-8">
</head>

<body>
<script src="sourcehtmx/htmx.min.js"></script>

<form hx-get="http://localhost/tpnoir/comptepresi.php">
  <button type="submit">Afficher</button>
  <input type="hidden" id="choix" name="choix" value="2"><br>

```

</form>

</body>

</html>

code de versement.html

<html>

<head>

<meta charset="UTF-8">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

</head>

<body>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p" crossorigin="anonymous"></script>

<script src="sourcehtmlx/htmlx.min.js"></script>

<form hx-post="http://localhost/tpnoir/comptepresi.php">

<div class="row mb-3">

<div class="col-2">

<label for="numcompte">Numcompte</label>

</div>

<div class="col-6">

<input id="numcompte" name="numcompte" type="text" class="form-control">

</div>

</div>

<div class="row mb-3">

<div class="col-2">

<label for="somme">Somme</label>

</div>

<div class="col-6">

<input id="somme" name="somme" type="text" class="form-control">

</div>

</div>

<input type="hidden" id="choix" name="choix" value="3">

<button type="submit">Verser</button>

</form>

</body>

</html>

code de htmx/delete .html

```
<html>
<head>
  <meta charset="UTF-8">
</head>

<body>
<script src="sourcehtmx/htmx.min.js"></script>

<form hx-post="http://localhost/tpnoir/comptepresi.php">
Numcompte:  <input id="numcompte" name="numcompte" type="text"><br>
             <input type="hidden" id="choix" name="choix" value="4"><br>
             <button type="submit">Valider</button>
</form>

</body>

</html>
```

Code de htmx/navigation.html

```
<!DOCTYPE html>
<html>
<head>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">
</head>

<body>
  <script src="sourcehtmx/htmx.min.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
ka7Sk0GlIn4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>

  <button class="btn" hx-get="http://localhost/htmx/creation.html" hx-target="#creation" > Creer
  compte
  </button>

  <button class="btn" hx-get="http://localhost/htmx/lecture.html" hx-target="#creation" > Afficher
  </button>
  <button class="btn" hx-get="http://localhost/htmx/versement.html" hx-target="#creation" > Verser
  </button>

  <button class="btn" hx-get="http://localhost/htmx/delete.html" hx-target="#creation" > Supprimer
```

```

</button>
<div id="creation"></div>

</body>
</html>

```

Résultats

On accède à navigation.html

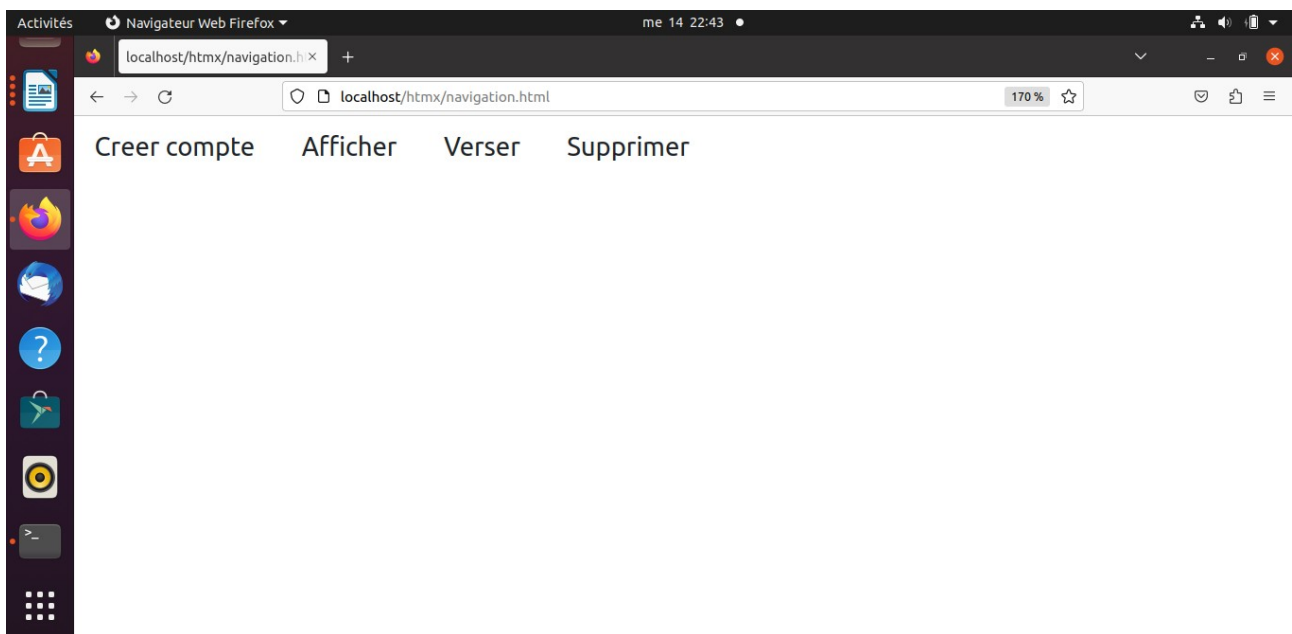


fig4 : nav + [htmx](#)

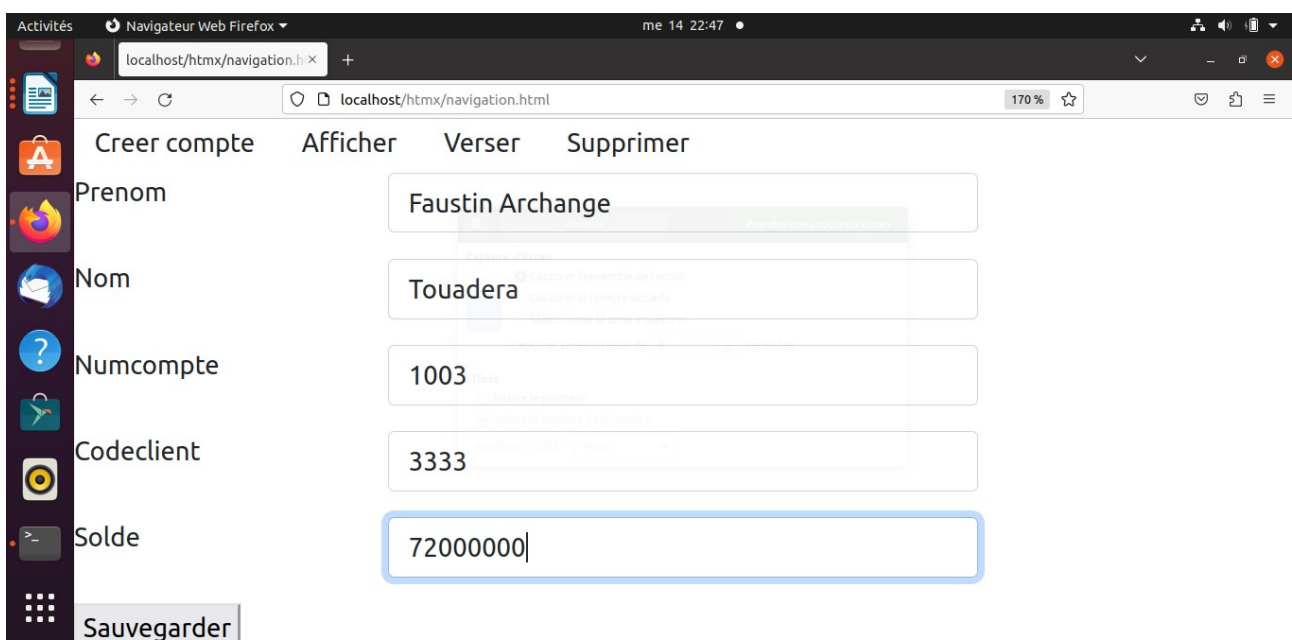


fig5 : création de compte de Touadera

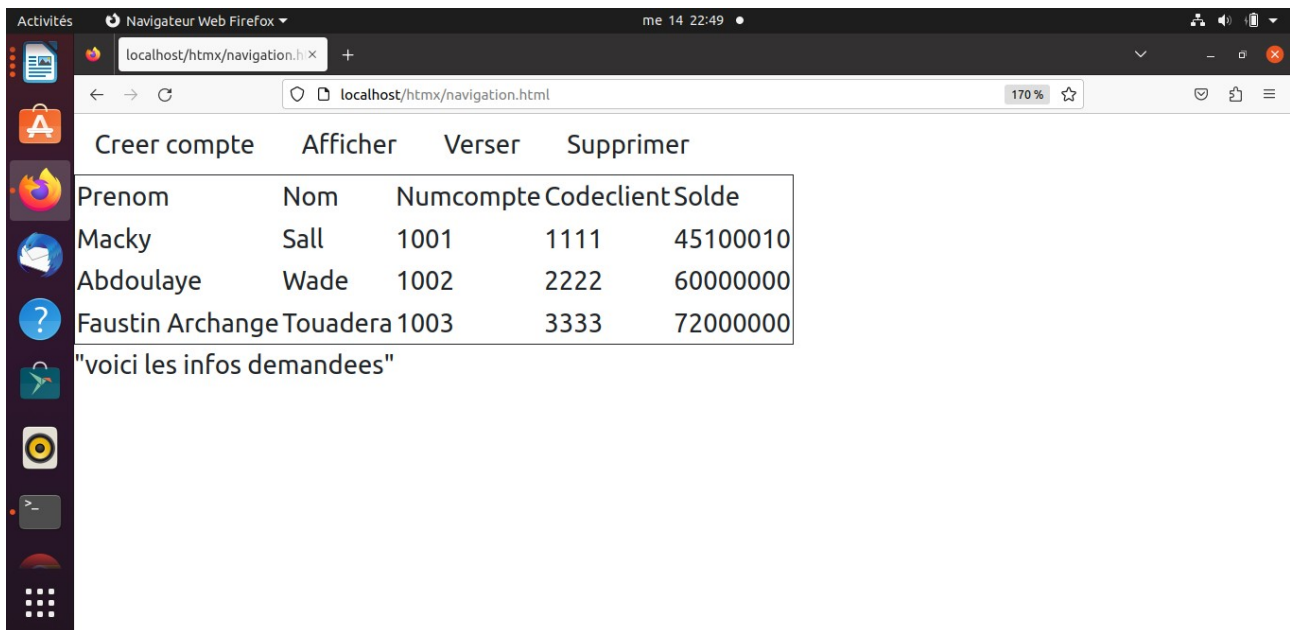


fig. 6 : Affichage des résultats

Versions 8000000 sur le compte de Touadera

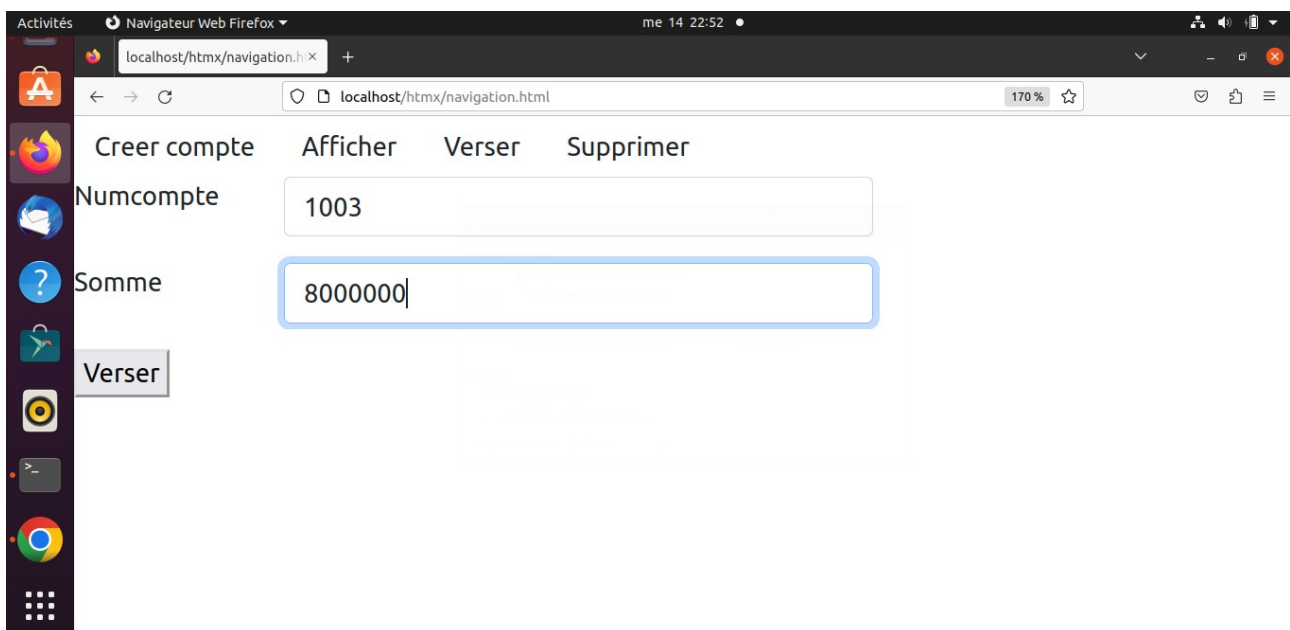


fig7 : Versement sur le compte de Touadera

Après versement, on clique sur afficher

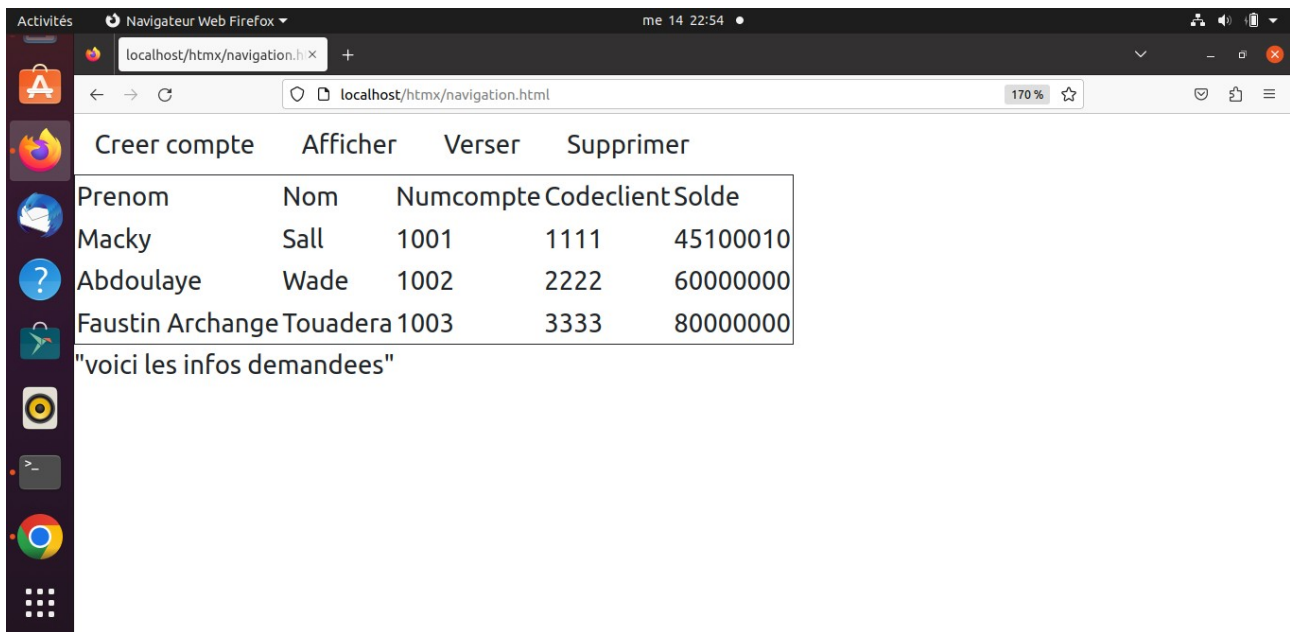


fig8 : résultat de versement

Supprimons le compte de Wade

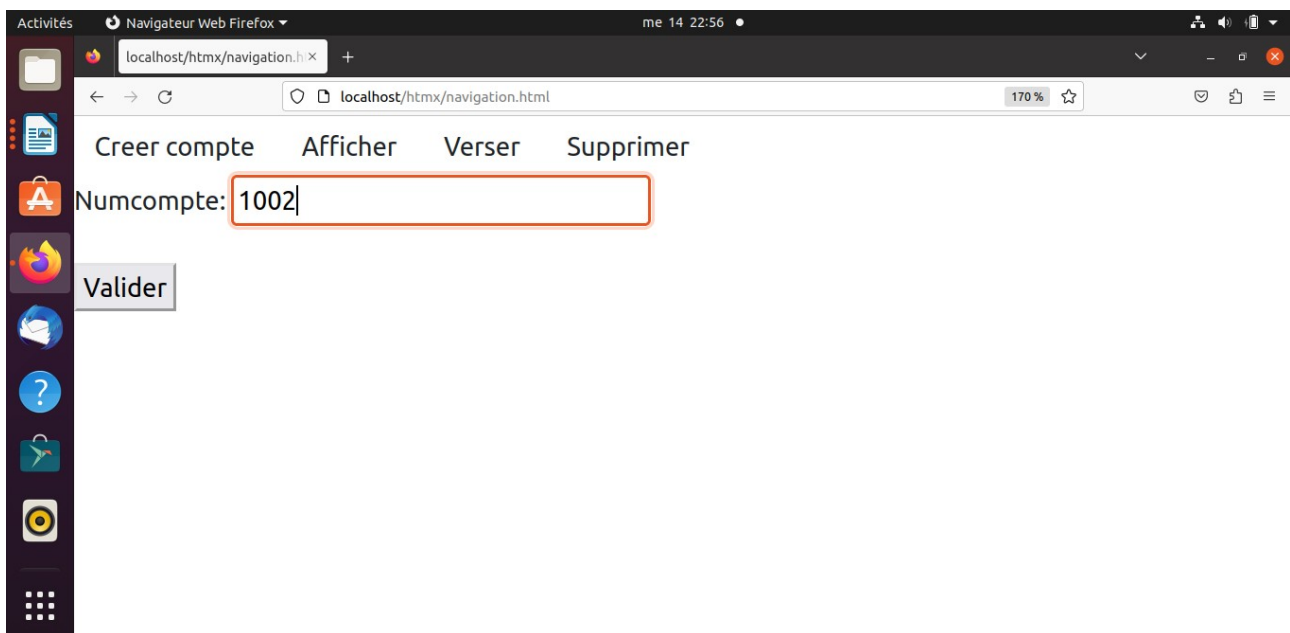


fig. 8 : suppression du compte de Wade

Résultat

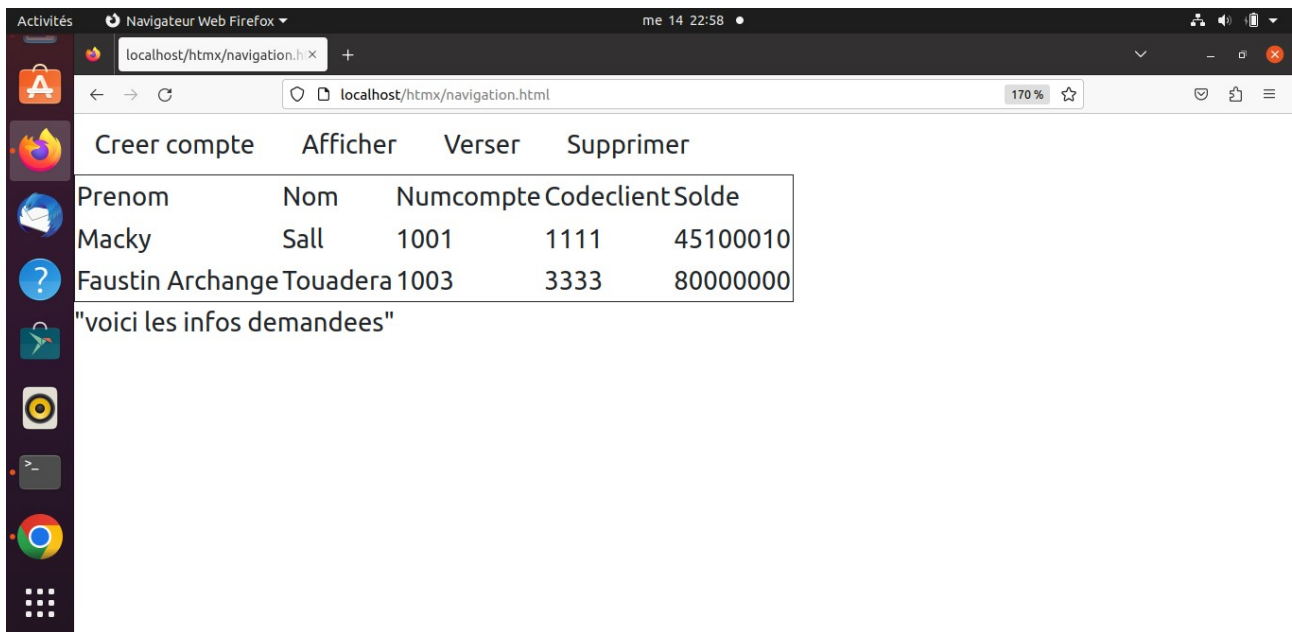


fig9 : Affichage après suppression de Wade

Conclusion :

htmx permet d'avoir une application monopage (SPA) très facilement.
On peut coupler htmx avec bootstrap pour avoir une application avec une interface très agréable.