

## Étape 3 – Affichage d'une fiche PC

Lire le JSON enregistré et l'afficher dans une page Bootstrap 5 lisible.

5TTR

 Exercice ambitieux

# Objectif de l'étape

À la fin, tu pourras accéder à `/inventaire/pc/PC-DE-LUDO` dans ton navigateur et voir une **page complète** avec :

- les infos générales du PC (OS, CPU, RAM, dernière mise à jour),
- une liste des disques avec leur état,
- les 10 processus les plus gourmands,
- les applications installées,
- les téléchargements récents.


On ne fait **pas encore** de graphe ni d'historique. C'est pour l'étape 4.

# 1. Bootstrap 5 – gabarit de base

Crée `public/partials/header.php` (pour réutiliser le même gabarit partout) :

```
<?php
1 // $titre est passée par chaque page qui inclut ce fichier
2 $titre = $titre ?? 'Inventaire';
?>
3 <!DOCTYPE html>
4 <html lang="fr" data-bs-theme="dark">
5 <head>
6     <meta charset="UTF-8">
7     <meta name="viewport" content="width=device-width, initial-scale=1.0">
8     <title><?= htmlspecialchars($titre) ?> – Inventaire</title>
    <link rel="stylesheet"
        href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.m
9     <link rel="stylesheet" href="/inventaire/css/app.css">
10 </head>
11 <body>
```

```

12 <nav class="navbar navbar-expand-lg bg-body-tertiary border-bottom">
13   <div class="container">
14     <a class="navbar-brand fw-bold" href="/inventaire">
15        Inventaire
16     </a>
17     <div class="navbar-nav">
18       <a class="nav-link" href="/inventaire">Dashboard</a>
19     </div>
20   </div>
21 </nav>
22 <main class="container py-4">

```

Et `public/partials/footer.php` :

```

1 </main>
2 <footer class="container py-4 text-center text-secondary small">
3   Projet inventaire – 5e TQ Informatique
4 </footer>
5 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap"
6 </body>
7 </html>

```

💡 Pourquoi `data-bs-theme="dark"` ? Bootstrap 5.3 a un **mode sombre natif**. Une ligne, tout s'adapte. Pratique pour un dashboard d'admin (et plus écoresponsable sur OLED).

⚠️ **TOUJOURS** `htmlspecialchars()` quand tu affiches une donnée qui vient d'ailleurs (JSON, formulaire, URL). Sinon, faille XSS garantie.

## 2. La fiche PC

Crée `public/pc.php` :

```

<?php
1 declare(strict_types=1);
2 require __DIR__ . '/../src/storage.php';
3
4 // L'URL ressemble à : /inventaire/pc/PC-DE-LUDO
5 // On extrait le hostname de l'URL.

```

```

6  $segments = explode('/', trim($_SERVER['REQUEST_URI'], '/'));
7  $hostname = end($segments);
8  $hostname = preg_replace('/[^\a-zA-Z0-9_\-]/', '', $hostname); // sécurité
9
10 if ($hostname === '') {
11     http_response_code(400);
12     echo "Hostname manquant.";
13     exit;
14 }
15
16 $rapport = charger_dernier_rapport($hostname);
17
18 if ($rapport === null) {
19     http_response_code(404);
20     echo "Aucun rapport pour le PC <strong>" . htmlspecialchars($hostname)
21     exit;
22 }
23
24 $titre = "PC " . $hostname;
25 require __DIR__ . '/partials/header.php';
26
27 <h1 class="h2 mb-1"><?>= htmlspecialchars($hostname) ?></h1>
28 <p class="text-secondary">
29     Dernier rapport :
30     <?>= htmlspecialchars($rapport['timestamp'] ?? 'inconnu') ?>
31 </p>
32
33 <div class="row g-3 mb-4">
34     <!-- Carte OS -->
35     <div class="col-md-3">
36         <div class="card h-100">
37             <div class="card-body">
38                 <div class="text-secondary small">Système</div>
39                 <div class="fw-bold">
40                     <?>= htmlspecialchars($rapport['system']['os'] ?? '-') ?>
41                 </div>
42                 <div class="small text-secondary">
43                     <?>= htmlspecialchars($rapport['system']['osVersion'] ?? '-') ?>
44                 </div>
45             </div>
46         </div>
47     </div>
48

```

```

49     <!-- Carte CPU -->
50     <div class="col-md-3">
51         <div class="card h-100">
52             <div class="card-body">
53                 <div class="text-secondary small">Processeur</div>
54                 <div class="fw-bold">
55                     <?>= htmlspecialchars($rapport['system']['cpu'] ?? '-')
56                 </div>
57                 <div class="small text-secondary">
58                     <?>= (int)($rapport['system']['cpuCores'] ?? 0) ?> cœurs
59                 </div>
60             </div>
61         </div>
62     </div>
63
64     <!-- À toi : carte RAM, carte Disque principal -->
65 </div>
66
67 <!-- Liste des disques -->
68 <section class="mb-4">
69     <h2 class="h5">Disques</h2>
70     <table class="table table-sm">
71         <thead>
72             <tr><th>Lettre</th><th>Total (Go)</th><th>Libre (Go)</th><th>Us
73         </thead>
74         <tbody>
75             <?php foreach (($rapport['disks'] ?? []) as $disque): ?>
76                 <?php
77                 $total = (float)($disque['totalGo'] ?? 0);
78                 $libre = (float)($disque['freeGo'] ?? 0);
79                 $usage = $total > 0 ? round((1 - $libre / $total) * 100) :
80                 ?>
81                 <tr>
82                     <td><?>= htmlspecialchars($disque['letter'] ?? '?') ?></td>
83                     <td><?>= number_format($total, 1, ',', ' ') ?></td>
84                     <td><?>= number_format($libre, 1, ',', ' ') ?></td>
85                     <td>
86                         <div class="progress" style="height: 18px;">
87                             <div class="progress-bar
88                                 <?>= $usage > 85 ? 'bg-danger' : ($usage > 70
89                                 style="width: <?>= $usage ?>%">
90                                 <?>= $usage ?> %
91                             </div>
92                         </div>
93                     </td>
94                 </tr>
95             </tbody>
96         </table>
97     </section>
98 </div>

```

```

88         </td>
89     </tr>
90     <?php endforeach; ?>
91 </tbody>
92 </table>
93 </section>
94
95 <!-- À toi : sections processus, applications, téléchargements -->
96
97 <?php require __DIR__ . '/partials/footer.php'; ?>

```

### 3. La fonction charger\_dernier\_rapport

À ajouter dans `src/storage.php` :

```

1  function charger_dernier_rapport(string $hostname): array
2  {
3      $fichier = DATA_ROOT . '/' . $hostname . '/latest.json';
4      if (!is_file($fichier)) {
5          return null;
6      }
7      $contenu = file_get_contents($fichier);
8      $data = json_decode($contenu, true);
9      return is_array($data) ? $data : null;
10 }

```

### 4. Ce que tu dois compléter toi-même

Le code ci-dessus te donne le squelette. À toi de compléter, en suivant exactement le même schéma :

#### a) Carte RAM

Affiche : RAM totale, RAM libre, et une barre de progression indiquant le pourcentage d'utilisation.

#### b) Carte « Disque C: »

Reprend l'info du premier disque pour avoir un aperçu rapide en haut de page.

#### c) Section processus

Liste les `topProcesses` dans un `<table>` ou des `<div class="card">`. Affiche le nom, le CPU (en secondes – explique-le en tooltip Bootstrap), et la mémoire en Mo.

## d) Section applications

Affiche `applications` dans un tableau avec recherche côté client. Le plus simple : un `<input>` qui filtre les lignes en JavaScript.

```
1 document.getElementById('appSearch').addEventListener('input', (e) => {
2     const q = e.target.value.toLowerCase();
3     document.querySelectorAll('#appList tr').forEach(tr => {
4         tr.style.display = tr.textContent.toLowerCase().includes(q) ? '' :
5     });
6 });
```

## e) Section téléchargements récents

Une liste simple. Pense à formater la date en français (`date('d/m/Y H:i', strtotime($download['downloadedAt']))`).

## f) Un indicateur écoresponsable

Au choix :

- âge du dernier rapport (si > 7 jours → badge orange « rapport ancien ») ;
- pourcentage de RAM utilisé en moyenne sur l'historique (à faire à l'étape 4) ;
- **score d'utilisation** : faible RAM + faible CPU pendant une semaine → suggérer « PC sous-utilisé, à mutualiser ? ».

## Travailler avec l'IA – la stratégie « squelette + remplissage »

À cette étape, tu as un **squelette qui tourne** (carte OS, carte CPU, tableau disques). Pour les autres sections, le bon prompt c'est :

Voici ma carte CPU dans une fiche PHP : [colle le bloc HTML de la carte CPU]. Sur le même modèle, fais-moi la carte RAM, qui affiche la RAM totale en Go, la RAM libre, et une barre de progression Bootstrap indiquant le pourcentage d'utilisation. Garde exactement le même style de `card` Bootstrap 5 dark mode. Donne juste le HTML + PHP, sans CSS ajouté.

Pourquoi c'est un **bon** prompt :

- Tu donnes un **exemple existant** (l'IA copie ton style).
- Tu précises **exactement** ce que tu veux.

- Tu **bornes** le périmètre (pas de CSS, pas de framework, pas de JS).

L'IA livre 15 lignes que tu colles, et qui s'intègrent **visuellement** avec le reste de ta page.

## 5. Tester avec plusieurs PCs

Tu n'as qu'un seul PC ? Triche **proprement** : copie ton `latest.json` dans plusieurs sous-dossiers de `data/pcs/` :

```
data/pcs/  
├── PC-LUDO/latest.json  
├── PC-AMINE/latest.json  
└── PC-SOFIA/latest.json
```

Édite à la main les hostnames et quelques valeurs (RAM, espace disque) pour simuler un parc varié. **C'est une technique d'ingénieur classique** : on appelle ça des *fixtures*. On en reparle à l'étape 4 pour le dashboard.

### Critères de réussite de l'étape 3

- [] L'URL `/inventaire/pc/MON-HOSTNAME` affiche bien les infos du PC.
- [] Une URL avec un hostname inconnu renvoie un **404 propre**.
- [] Les disques apparaissent avec une **barre de progression colorée** selon l'usage (vert / jaune / rouge).
- [] Le rendu est lisible en **dark mode** et en **light mode** (essaie de basculer `data-bs-theme`).
- [] **Aucun warning PHP** dans la page (pas de `Undefined index`).
- [] Tu peux expliquer à quoi sert `htmlspecialchars`.

### Pièges fréquents

#### Piège

`Undefined array key "system"`  
Caractères `é` qui deviennent `Ã©`  
Quelqu'un met `<script>` dans son hostname  
404 alors que le fichier existe

#### Solution

Toujours `$rapport['system']['os'] ?? '-'`  
Header `charset=utf-8` + `JSON_UNESCAPED_UNICODE`  
`htmlspecialchars()` partout  
Vérifier `is_file` avec un `var_dump` du chemin

## Pour la suite

À l'étape 4 : dashboard global, graphe d'espace disque dans le temps, et historique. C'est là que ça devient visuel.