

SAÉ 1.05 | Produire un site Web

Mise en situation et consignes :

Les SAE sont évaluées en utilisant une démarche portfolio. Cette démarche vise à rendre les étudiants acteurs de leur évaluation. Chaque SAE est intégrée dans un portfolio dans le but d'y laisser une trace ainsi qu'une analyse commentée et justifiée par l'étudiant des notions acquises.

Les étudiants doivent donc avoir un outil pour montrer les réalisations et les justifications. Cet outil servira aux étudiants pendant les trois années du BUT et évoluera en fonction des nouveaux apprentissages vus par les étudiants.

Cet outil prend la forme d'un site web, hébergé :

- Dans un premier temps, en local sur l'ordinateur que l'étudiant utilise,
- Dans un second temps, sur l'intranet de l'IUT sur votre serveur dédié afin que chaque enseignant puisse y avoir accès.

Chaque étudiant doit réfléchir à la mise en page qui lui semble la plus adéquate.

Pour chaque SAÉ, il faudra avoir au minimum :

- Le numéro de la SAÉ,
- La compétence mise en œuvre,
- Les apprentissages critiques mobilisés avec pour chacun :
 - Une justification et une explication qui indiquent pourquoi et comment l'étudiant a acquis cet apprentissage,
- Les ressources mobilisées,
- Un descriptif de la production réalisée,
- Un visuel de la production réalisée,
- Selon votre convenance, d'autres données.

Les données sont utilisées à plusieurs endroits, par exemple, un apprentissage critique peut intervenir dans plusieurs ressources et plusieurs SAÉ. Une ressource peut être utilisée dans plusieurs SAÉ, etc.

L'idéal est alors d'avoir une structure de données décrivant les ressources, une structure de données décrivant les SAÉ et une structure de données décrivant les SAÉ. Dans cette dernière structure, il n'y aurait alors que le nom des apprentissages critiques, sans toute la description.

S'il est nécessaire d'afficher les données de l'apprentissage critique, il sera alors possible de chercher les informations dans la structure qui décrit les apprentissages critiques.

AC14.01 | Exploiter de manière autonome un environnement de développement efficace et productif

```
js > JS script3.js > ...
1 let str=window.location.href //recherche de l'url de la page
2 var url= new URL(str) //création d'une variable nomme url avec une copie de l'url de la page
3 var name= url.searchParams.get("numéro") //recherche de la valeur du parametre numéro soit ici le numéro de l'AC
4 console.log(name)
5
6 let module=AC[name] //création d'un variable module qui correspond a l'objet d'un AC
7
8 document.querySelector("#numeroAC").innerText=name //Affichage du numéro de l'AC dans la div prévu pour avec l'id n
9 document.querySelector("#titreAC").innerText=module.titre //Affichage du titre de l'AC dans la div prévu avec l'id
10
11 let lien="" //Création d'une variable vide pour y insérer le lien vers un pdf justificatif de l'AC
12 lien+="
```

Mon environnement est efficace et productif grâce aux commentaires que j'ai ajouté dans mon code, surtout au niveau du javascript qui nécessite beaucoup de compréhension. Avec tout ces commentaires, il est possible alors de suivre et comprendre puis faire le code pour une personne qui n'a pas commencé le code lui-même.

J'ai donc appris à mettre des commentaires dans mon code et à faire que mes pages soit propres et claires pour tout le monde.

AC14.02 | Produire des pages Web fluides incluant un balisage sémantique efficace et des interactions simples

```
index.html > html
1  <!DOCTYPE html>
2  <html lang="fr">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Site Web Portfolio - Page Accueil</title>
7      <link rel="stylesheet" href="styles/style.css">
8
9  </head>
10 <body>
11     <div class="flouaccueil">
12         <div class="pageaccueil">
13             <header class="header-accueil">
14                 
15             </header>
16             <div class="flexcolonne">
17                 <div class="boutonaccueil">
18                     <a class="blanc texte bouton" href="liste_S.A.É.html">
19                         <h3 class="marge0">MES PROJETS</h3>
20                     </a>
21                     <a class="blanc texte bouton" href="index.html">
22                         <h3 class="marge0">ACCUEIL</h3>
23                     </a>
24                     <a class="blanc texte bouton" href="Présentation.html">
25                         <h3 class="marge0">PRÉSENTATION</h3>
26                     </a>
27                     <div class="pageaccueil2">
28                         <a class="blanc texte bouton" href="A-Propos.html">
29                             <h3 class="marge0">A PROPOS</h3>
30                         </a>
31                         <a class="blanc texte bouton" href="" onclick="Quitter()">
32                             <h3 class="marge0">QUITTER</h3>
33                         </a>
34                     </div>
35                 </div>
36             </div>
37             <footer class="footer-accueil">
38                 <p class="blanc texte margefooter-accueil">Version 1.0.0</p>
39                 <p class="blanc texte margefooter-accueil">Copyright DA-ROS ALAIN, NE PAS DIFFUSER</p>
40             </footer>
41         </div>
42     </div>
43     <script src="js/dataSAE.js"></script>
44     <script src="js/script.js"></script>
45 </body>
46 </html>
```

Page d'accueil du site

Pour faire une page web incluant un bon balisage efficace et des interactions simples, j'ai utilisé les balises `div` pour faire des blocs de différents éléments, des balises `a` pour faire des liens entre différentes pages. Pour les titres j'ai utilisé les balises de titres. Tout élément a des classes pour faire que tout est un style propre.

AC14.03 | Générer des pages Web à partir de données structurées

```
let liste="" //création variable vide pour y inserer grace a la boucle les valeurs souhaitées.

//Création d'une boucle pour chercher les valeurs dans la base de données et les insérer dans des liens cliquables simples.
for (let key in SAE) //création d'une boucle pour afficher pour chaque SAE, leur numero, titre et compétences mobilisées.
{
    let module= SAE[key]
    let infos=`<a href='infos_S.A.É.html?numéro=${key}' id=${key}>
        <div class='carre flexcolonne'>
            <h1 class='petitemarge blanc texte'>${key}</h1>
            <h3 class='marge0 blanc'>${module.titre}</h3>
            <h3 class='marge0 carrecompetence flexcentre1'>${module.compétences}</h3>
        </div>
    </a>` //utilisation d'un littéral de gabarits
    liste+= infos
}

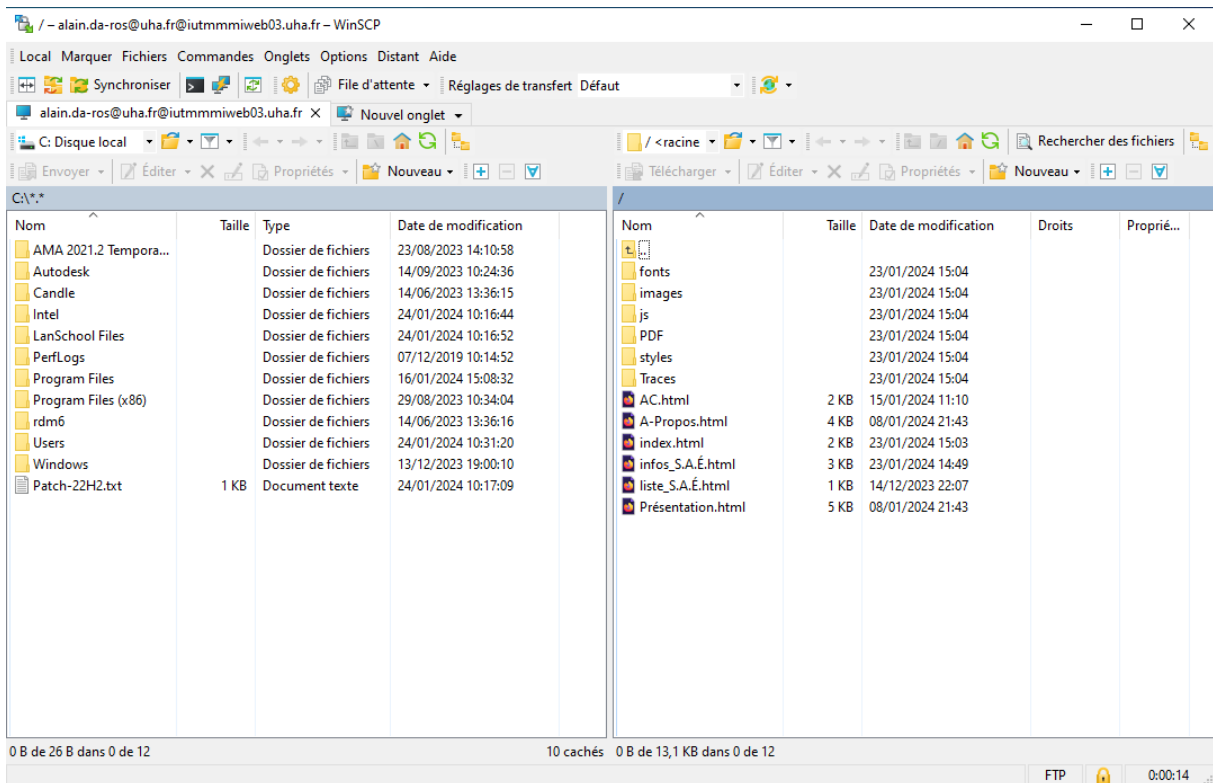
document.querySelector(".sae").innerHTML=liste
```

Capture écran du script java du

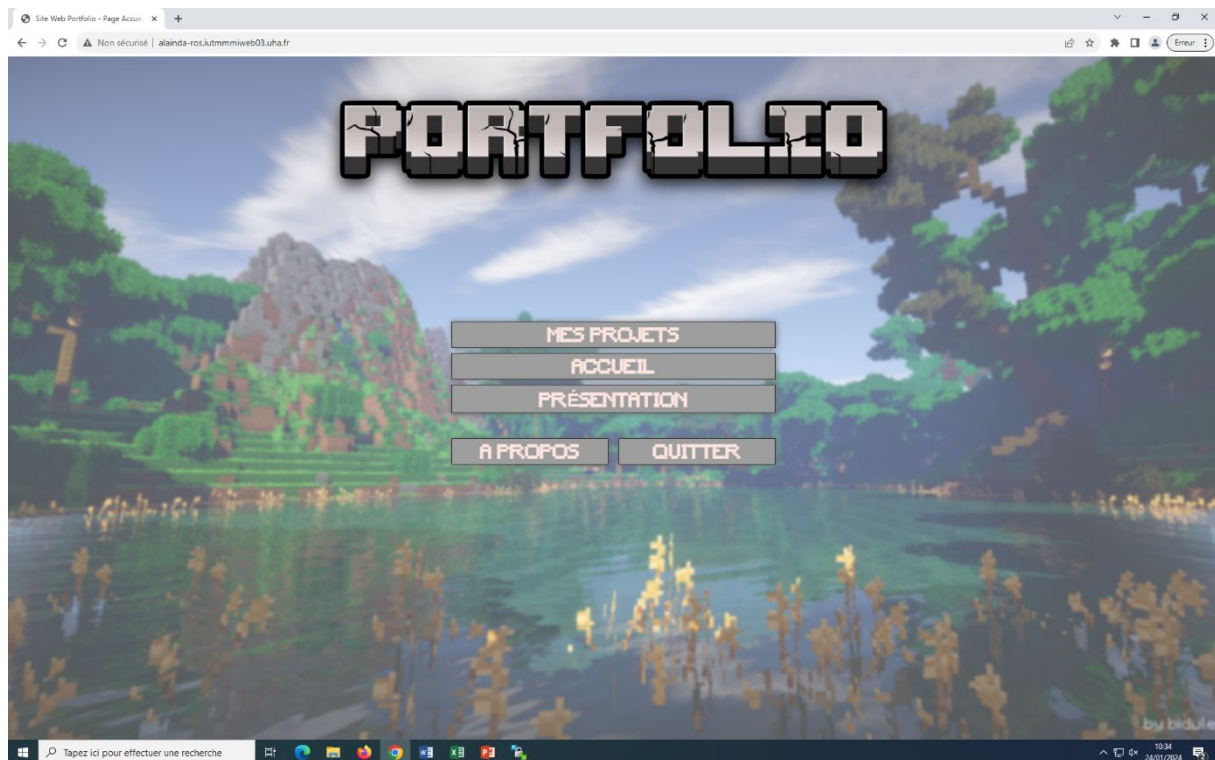
Pour générer les données dans la page internet, il faut prendre les données comprises dans la base de données et les insérer dans la page web. Dans mon cas, j'ai créé sur la page internet une div avec une classe nommée « sae » ou j'importe grâce au code java-script ci-dessus les données voulues qui sont comprises dans la base de données.

J'ai utilisé une boucle afin de faire la même opération pour toutes les sae comprises dans la constante SAE qui forme la base de données. Cela permet alors de chercher pour chaque SAE les valeurs souhaitées.

AC14.04 | Mettre en ligne une application Web en utilisant une solution d'hébergement standard



Appli WinSCP



Site sur le réseau de l'IUT

Pour mettre en ligne notre site sur le réseau de l'IUT, j'ai utilisé l'application WinSCP afin d'avoir accès au fichier du site internet qui porte mon nom. Une fois sur WinSCP, j'ai juste à glisser les fichiers de mon site dans le dossier ouvert. Après avoir placé les fichiers dans le site en ligne sur le réseau, je peux l'ouvrir sur l'ordinateur.

Les difficultés de cette méthode sont d'avoir le bon port et les bons identifiants de connexion dans l'application WinSCP et de ne pas faire d'erreur dans la mise en ligne des fichiers du site.