# Explication code CLOUD CONTROL

Partie Connexion :

**Fichier : connexion.html :**

**Explications :**

* **<!DOCTYPE >** : Déclare le document comme 5.
* **< lang="en">** : Définit la langue du document comme anglais.
* **<head>** : Contient les métadonnées, le lien vers le fichier CSS, et le titre de la page.
  + **<meta charset="UTF-8">** : Définit l'encodage de caractères.
  + **<meta http-equiv="X-UA-Compatible" content="IE=edge">** : Assure la
  + compatibilité avec Internet Explorer.
  + **<meta name="viewport" content="width=device-width, initial-scale=1.0">** : Assure une mise en page responsive.
  + **<link rel="stylesheet" href="connexion.css">** : Lien vers le fichier CSS externe.
  + **<title>connexion</title>** : Titre de la page affiché dans l'onglet du navigateur.
* **<body>** : Contient le contenu visible de la page.
  + **<div class="box">** : Boîte principale contenant le formulaire de connexion.
    - **<span class="borderLine"></span>** : Élément décoratif pour la bordure animée.
    - **<form action="..." method="get">** : Lien de là où sa nous emmènes lorsqu’on a finit de remplir l’identification et qu’on appuie sur le Connexion.

<h2>CLOUD CONTROL</h2>

* <h2> est une balise HTML utilisée pour créer un titre de niveau 2. Ici, il s'agit du titre principal du formulaire, qui est "CLOUD CONTROL".

<div class="inputBox">

* <div> est une balise HTML utilisée pour créer une section ou un conteneur dans lequel nous plaçons d'autres éléments, comme des champs de saisie.
* class="inputBox" est une classe CSS qui peut être utilisée pour appliquer des styles spécifiques à ce conteneur, comme la mise en forme des champs de saisie et des étiquettes.

<input type="text" required="required">

* <input> est une balise HTML qui permet aux utilisateurs de saisir des informations. type="text" spécifie que c'est un champ de texte où l'utilisateur peut entrer du texte.
* required="required" est un attribut qui indique que ce champ doit être rempli avant que le formulaire puisse être soumis.

<span>Identifiant</span>

* <span> est une balise HTML utilisée pour regrouper et styliser du texte en ligne.
* Ici, "Identifiant" est une étiquette qui décrit le champ de texte précédent, indiquant à l'utilisateur ce qu'il doit saisir.

<i></i>

* <i> est une balise HTML utilisée traditionnellement pour styliser le texte en italique. Dans ce contexte, elle pourrait être utilisée pour ajouter un élément décoratif visuel à côté du champ de texte, comme une icône.

<div class="inputBox">

* C'est à nouveau un conteneur pour un autre champ de saisie (mot de passe), avec une classe CSS pour appliquer des styles spécifiques.

<input type="password" required="required">

* <input type="password"> crée un champ de saisie où les caractères saisis sont masqués (pour les mots de passe).
* required="required" assure que ce champ doit être rempli avant de soumettre le formulaire.

<span>Mot De Passe</span>

* Une autre étiquette pour décrire le champ de mot de passe, indiquant à l'utilisateur ce qu'il doit saisir.

<i></i>

* Également utilisé ici comme un élément décoratif visuel optionnel à côté du champ de mot de passe.

<div class="links">

* <div> utilisé pour regrouper les liens supplémentaires ou autres éléments associés au formulaire.

<a href="#">Mot de passe oublié</a>

* <a> est une balise HTML pour créer un lien hypertexte.
* href="#" spécifie que le lien mène à l'URL actuelle (dans ce cas, il ne mène nulle part car il utilise un # comme URL fictive).
* "Mot de passe oublié" est le texte affiché pour le lien, généralement utilisé pour permettre à l'utilisateur de récupérer un mot de passe perdu.

<input type="submit" value="Connexion">

* <input type="submit"> crée un bouton de soumission pour le formulaire.
* value="Connexion" est le texte affiché sur le bouton, indiquant à l'utilisateur que cliquer dessus soumettra les informations saisies

Partie Connexion : CSS

**Fichier : connexion.css :**

* **Importation de la police :**

css

@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700;800;900&display=swap');

* **Styles généraux :**

css

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Poppins', sans-serif;

}

* **Styles pour le corps (body) :**

css

body {

display: flex;

justify-content: center;

align-items: center;

min-height: 100vh;

background: #dbdbdb;

}

* **Styles pour la boîte de connexion (.box) :**

css

.box {

position: relative;

width: 380px;

height: 420px;

background: #1c1c1c;

border-radius: 8px;

overflow: hidden;

}

* **Effets d'animation pour la bordure et le fond :**

css

.box:before, .box:after, .borderLine:before, .borderLine:after {

content: '';

position: absolute;

top: -50%;

left: -50%;

width: 380px;

height: 420px;

transform-origin: bottom right;

animation: animate 6s linear infinite;

}

.box:before {

background: linear-gradient(0deg, transparent, transparent, #2bb0fd, #31b0fa, #3db8ff);

z-index: 1;

}

.box:after {

background: linear-gradient(0deg, transparent, transparent, #00a2ff, #00a2ff, #00a2ff);

z-index: 1;

animation-delay: -3s;

}

.borderLine {

position: absolute;

top: 0;

inset: 0;

}

.borderLine:before {

background: linear-gradient(0deg, transparent, transparent, #2cb2ff, #2cb2ff, #2cb2ff);

z-index: 1;

animation-delay: -1.5s;

}

.borderLine:after {

background: linear-gradient(0deg, transparent, transparent, #00a2ff, #00a2ff, #00a2ff);

z-index: 1;

animation-delay: -4.5s;

}

@keyframes animate {

0% {

transform: rotate(0deg);

}

100% {

transform: rotate(360deg);

}

}

* **Styles pour le formulaire :**

css

.box form {

position: absolute;

inset: 4px;

background: #222;

padding: 50px 40px;

border-radius: 8px;

z-index: 2;

display: flex;

flex-direction: column;

}

.box form h2 {

color: #fff;

font-weight: 500;

text-align: center;

letter-spacing: 0.1em;

}

.box form .inputBox {

position: relative;

width: 300px;

margin-top: 35px;

}

.box form .inputBox input {

position: relative;

width: 100%;

padding: 20px 10px 10px;

background: transparent;

outline: none;

border: none;

box-shadow: none;

color: #23242a;

font-size: 1em;

letter-spacing: 0.05em;

transition: 0.5s;

z-index: 10;

}

.box form .inputBox span {

position: absolute;

left: 0;

padding: 20px 0px 10px;

pointer-events: none;

color: #8f8f8f;

font-size: 1em;

letter-spacing: 0.05em;

transition: 0.5s;

}

.box form .inputBox input:valid ~ span,

.box form .inputBox input:focus ~ span {

color: #fff;

font-size: 0.75em;

transform: translateY(-34px);

}

.box form .inputBox i {

position: absolute;

left: 0;

bottom: 0;

width: 100%;

height: 2px;

background: #fff;

border-radius: 4px;

overflow: hidden;

transition: 0.5s;

pointer-events: none;

}

.box form .inputBox input:valid ~ i,

.box form .inputBox input:focus ~ i {

height: 44px;

}

.links {

display: flex;

justify-content: space-between;

}

.box form .links a {

margin: 10px 0;

font-size: 0.75em;

color: #8f8f8f;

text-decoration: none;

}

.box form .links a:hover,

.box form .links a:nth-child(2) {

color: #fff;

}

.box form input[type="submit"] {

border: none;

outline: none;

padding: 9px 25px;

background: #fff;

cursor: pointer;

font-size: 0.9em;

border-radius: 4px;

font-weight: 800;

width: 200px;

margin-top: 10px;

}

.box form input[type="submit"]:active {

opacity: 0.8;

}

### Instructions pour Modifier le Code

1. **Modifier les Couleurs :**
   * Les couleurs de fond, de bordure et de texte peuvent être ajustées dans les sections correspondantes du CSS.
   * Exemple : Pour changer la couleur de fond de la boîte de connexion :

css

.box {

background: #nouvelleCouleur;

}

1. **Changer le Titre et les Étiquettes :**
   * Le texte du titre et des étiquettes peut être modifié directement dans le .
   * Exemple : Pour changer le texte "V-LEUR" :

<h2>Nouveau Titre</h2>

1. **Modifier les Animations :**
   * Les durées et les effets des animations peuvent être ajustés dans les keyframes (fonctionnalité puissante utilisée pour définir des animations en spécifiant les étapes clés) du CSS.
   * Exemple : Pour changer la durée de l'animation d’une fumée :

css

@keyframes smoke-animation {

0% { ... }

100% {

transform: translate(-50%, -80%) scale(2);

opacity: 0;

}

}

.smoke {

animation: smoke-animation 8s ease-in-out infinite; /\* Changer 6s à 8s \*/

}

1. **Ajouter des Champs de Formulaire :**
   * Pour ajouter de nouveaux champs de saisie, copiez une section inputBox et modifiez les attributs et les étiquettes en conséquence.
   * Exemple :

<div class="inputBox">

<input type="email" required="required">

<span>Email

Accueil.html : Explication :

Fichier : accueil.html :

**Corps de la Page (<body>)** :

* **En-tête (<header>)** : Contient le logo d'April Mutuelle avec un lien externe.
* **Contenu Principal (<main>)** :
  + Divisions (**<div>**) : Organisent les sections principales de la page.
  + **Sections de Détails (<div class="details">)** :
    - **Boîte de Détails (<div class="details-box">)** : Contient toutes les sections de contrôle.
  + **Sections de Contrôle (e.g., NUTANIX, FiwerWall Interne, REVERSE PROXY, etc.)** :
    - Bouton **(<button>** tag) : Déclenche l'affichage des détails associés.
    - Conteneur de Texte **(<div class="text-container hidden">)** : Cache initialement le contenu détaillé.
    - Sous-sections **(<div class="text-display-section">)** : Organisent chaque catégorie de contrôle.
    - **Boutons d'Action (<button class="submit-btn">)** : Exécutent des actions spécifiques comme supprimer ou créer.
    - **Fieldsets (<fieldset>)** : Regroupent des actions apparentées sous un même titre.
  + **Actions Spécifiques (e.g., Créer ou Supprimer des éléments)** :
    - Script **(<script src="accueil.js"></script>)** : Charge un script JavaScript externe pour manipuler les actions.

 **Pied de Page (<footer>)** :

* Lien de Contact : Affiche le CODE code source et un lien pour les demandes spécifiques.

### Étapes pour Ajouter un Nouveau bouton page d’accueil.Une image contenant texte, capture d’écran, Logiciel multimédia, Système d’exploitation Description générée automatiquement

**Exemple d'ajout d'un nouveau bouton pour une machine virtuelle**

<fieldset>

<legend>Nouvelle Machine Virtuelle</legend>

<div class="submit-wrapper">

<button class="submit-btn" onclick="location.href='suprNouvelleVM.html '">Supprimer !</button>

<button class="submit-btn" onclick="location.href='creerNouvelleVM.html '">Créer !</button>

</div>

</fieldset>

* + **<fieldset>** : C'est la balise principale qui définit un groupe d'éléments liés.
  + **<legend>** **</legend>**: C'est la balise qui indique le titre (ce que vous voulez mettre dans le bouton), vous devez mettre le titre entre les deux balises (**<legend>** **</legend>**) .
  + **<div class="submit-wrapper">** : Ceci est un conteneur pour les boutons d'action. Je vous conseil de laisser la même class pour garder un style identique aux autres boutons mais vous pouvez personnaliser les classes CSS selon votre envie.
  + **<button class="submit-btn" onclick="location.href='suprNouvelleVM.html '">Supprimer !</button> :** Pour garder le même style je vous conseil de copier coller, et de changer les mots surligné c’est à dire, le suprNouvelleVM.html correspond à la page qui va renvoyer lorsqu’on va appuyer sur le bouton qui dans cet exemple est Supprimer, et Supprimer correspond à se qu’il y a dans le bouton.
  + **<button class="submit-btn" onclick="location.href='creerNouvelleVM.html '">Créer !</button> :** Idem.
  + **</div> </fieldset> :** N’oubliez pas de fermer les commandes que vous avez déclaré au début.

### Exemple Complet d'Insertion

Voici à quoi pourrait ressembler l'insertion complète dans votre code existant :

**Section NUTANIX**

<div class="text-display-section">

<button class="submit-btn" data-target="textContainerNutanix">NUTANIX</button>

<div id="textContainerNutanix" class="text-container hidden">

<div class="details-box1">

<h2>NUTANIX</h2>

Fieldset existant pour Machine virtuelle Nutanix

<fieldset>

<legend>Machine virtuelle Nutanix</legend>

<div class="submit-wrapper">

<button class="submit-btn" onclick="location.href='suprVMNutanix.html '">Supprimer !</button>

<button class="submit-btn" onclick="location.href='creerVMNutanix.html '">Créer !</button>

</div>

</fieldset>

**Nouveau Fieldset pour une autre machine virtuelle**

<fieldset>

<legend>Nouvelle Machine Virtuelle</legend>

<div class="submit-wrapper">

<button class="submit-btn" onclick="location.href='suprNouvelleVM.html '">Supprimer !</button>

<button class="submit-btn" onclick="location.href='creerNouvelleVM.html '">Créer !</button>

</div>

</fieldset>

Partie Suppression VM Nutanix :

Fichier : suprVMNutanix.html

<title>Création de Machine Virtuelle</title>

* <title>Création de Machine Virtuelle</title> : Définit le titre de la page, affiché dans l'onglet du navigateur. <!-- Feuilles de style -->

<link rel="stylesheet" href="style.css">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

* <link rel="stylesheet" href="style.css"> : Lien vers une feuille de style CSS externe pour la mise en forme de la page.
* <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet"> : Lien vers la feuille de style Bootstrap pour utiliser des styles préconçus.

<body>

<!-- Header de la page -->

<header class="mb-4 d-flex justify-content-between align-items-center">

<img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo mx-auto d-block">

</header>

* <body> : Ouvre la section du document contenant le contenu visible de la page.
* <header class="mb-4 d-flex justify-content-between align-items-center"> : Débute la section en-tête avec des classes CSS pour la mise en forme.
* <img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo mx-auto d-block"> : Affiche une image avec le logo et applique des classes pour le style.

<!-- Contenu principal -->

<main>

<div class="container">

<div class="details-box p-4">

<!-- Titre principal -->

<h1 class="text-center mb-4">CRÉATION DE MACHINE VIRTUELLE</h1>

* <main> : Débute la section principale du contenu de la page.
* <div class="container"> : Débute une boîte de contenu centrée.
* <div class="details-box p-4"> : Débute une boîte avec des détails, avec des classes pour le style.
* <h1 class="text-center mb-4">CRÉATION DE MACHINE VIRTUELLE</h1> : Titre principal de la section, centré.

<!-- Bouton de retour à l'accueil -->

<a href="accueil.html" class="submit-btn">Accueil</a>

* <a href="accueil.html" class="submit-btn">Accueil</a> : Lien vers la page d'accueil avec une classe pour le style.

<!-- Bouton toggle pour les détails -->

<div class="btn main-btn bounce-btn mx-auto d-block mb-4" id="btn1">

<ion-icon name="chevron-down"></ion-icon>

</div>

* <div class="btn main-btn bounce-btn mx-auto d-block mb-4" id="btn1"> : Bouton pour afficher ou cacher des détails, avec des classes pour le style et un ID unique.
* <ion-icon name="chevron-down"></ion-icon> : Icône affichée dans le bouton.

<!-- Section des détails -->

<div class="details" id="details1">

<!-- Formulaire de création -->

<form action="process.php" method="post" id="creationForm">

<div class="details-box1 p-4 mb-4">

<!-- Titre du formulaire -->

<h2 class="text-center mb-4">NUTANIX</h2>

* <div class="details" id="details1"> : Débute une section cachée de détails.
* <form action="process.php" method="post" id="creationForm"> : Débute un formulaire qui envoie les données à "process.php" en utilisant la méthode POST.
* <div class="details-box1 p-4 mb-4"> : Débute une autre boîte de détails avec des classes pour le style.
* <h2 class="text-center mb-4">NUTANIX</h2> : Titre de cette section du formulaire, centré.

<!-- Légende pour le groupe de sélection -->

<fieldset class="mb-4">

<legend>Sélectionnez la machine virtuelle que vous voulez supprimer:</legend>

<!-- Grille pour les champs de saisie -->

<div class="row">

* <fieldset class="mb-4"> : Débute une section de groupe de champs de formulaire.
* <legend>Sélectionnez la machine virtuelle que vous voulez supprimer:</legend> : Légende descriptive du groupe de champs.
* <div class="row"> : Débute une grille pour organiser les champs de saisie en lignes et colonnes.

<div class="col-md-6 mb-3">

<!-- Sélection de l'environnement -->

<label>

<select name="environnement" class="form-select" id="environnement" required>

<option value="">Sélectionnez un environnement</option>

<option value="Developpement">Développement</option>

<option value="Recette">Recette</option>

<option value="Preproduction">Préproduction</option>

<option value="Infrastructure">Infrastructure</option>

<option value="Production">Production</option>

</select>

</label>

</div>

* <div class="col-md-6 mb-3"> : Débute une colonne de la grille occupant 6/12 de la largeur de la ligne.
* <label> : Débute une étiquette pour le champ suivant.
* <select name="environnement" class="form-select" id="environnement" required> : Liste déroulante pour sélectionner un environnement.
* <option value="">Sélectionnez un environnement</option> : Option par défaut de la liste.
* Les autres <option> ajoutent des choix à la liste déroulante.

<div class="col-md-6 mb-3">

<!-- Sélection du type de serveur -->

<label>

<select name="type\_serveur" class="form-select" id="type\_serveur" required>

<option value="">Sélectionnez un type de serveur</option>

<option value="Windows\_Server">Windows Server</option>

<option value="RedHat\_Entreprise">RedHat Entreprise</option>

</select>

</label>

</div>

* Colonne suivante de la grille pour sélectionner le type de serveur.

<div class="col-md-6 mb-3">

<!-- Sélection du composant -->

<label>

<select name="composant" class="form-select" id="composant" required>

<option value="">Sélectionnez un composant</option>

<option value="APP">Application (APP)</option>

<option value="SQL">Base de données SQL</option>

<option value="WEB">Serveur Web (WEB)</option>

</select>

</label>

</div>

* Colonne suivante de la grille pour sélectionner le composant.

<div class="col-md-6 mb-3">

<!-- Sélection du nom de la machine virtuelle -->

<label>

<select name="Nom\_VM" class="form-select" id="NomVM" required>

<option value="">Nom de la Machine Virtuelle</option>

<!-- Options filtrées par environnement -->

<!-- Options pour Developpement -->

<option value="DEVXXXSQL-VLIXX" data-environment="Developpement" data-os="RedHat\_Entreprise" data-component="SQL">DEVXXXSQL-VLIXX</option>

<option value="DEVXXXSQL-22XX" data-environment="Developpement" data-os="Windows\_Server" data-component="SQL">DEVXXXSQL-22XX</option>

<!-- Plus d'options pour d'autres environnements et types -->

</select>

</label>

</div>

</div>

</fieldset>

</div>

* Colonne suivante pour sélectionner le nom de la machine virtuelle avec options filtrées par environnement, système d'exploitation, et composant.

<div class="text-item text-center">

<button type="submit" class="submit-btn">Soumettre</button>

</div>

</form>

</div>

</div>

</div>

</main>

* Bouton pour soumettre le formulaire, centré.

<!-- Script pour le comportement de la page -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

<script src="script.js"></script>

<script>

document.addEventListener('DOMContentLoaded', function () {

const environnementSelect = document.getElementById('environnement');

const typeServeurSelect = document.getElementById('type\_serveur');

const composantSelect = document.getElementById('composant');

const nomVMSelect = document.getElementById('NomVM');

// Fonction pour filtrer et sélectionner automatiquement l'option de Nom\_VM

function filterNomVMOptions() {

const selectedEnvironment = environnementSelect.value;

const selectedOS = typeServeurSelect.value;

const selectedComponent = composantSelect.value;

let firstMatch = null;

// Parcourir toutes les options de Nom\_VM

for (const option of nomVMSelect.options) {

const optionEnvironment = option.getAttribute('data-environment');

const optionOS = option.getAttribute('data-os');

const optionComponent = option.getAttribute('data-component');

if (

(!selectedEnvironment || optionEnvironment === selectedEnvironment) &&

(!selectedOS || optionOS === selectedOS) &&

(!selectedComponent || optionComponent === selectedComponent)

) {

option.style.display = 'block';

if (!firstMatch) {

firstMatch = option;

}

} else {

option.style.display = 'none';

}

}

// Sélectionner automatiquement la première option correspondante

if (firstMatch) {

nomVMSelect.value = firstMatch.value;

} else {

nomVMSelect.value = '';

}

}

// Ajouter des écouteurs d'événements aux sélections

environnementSelect.addEventListener('change', filterNomVMOptions);

typeServeurSelect.addEventListener('change', filterNomVMOptions);

composantSelect.addEventListener('change', filterNomVMOptions);

// Appeler la fonction de filtrage lors du chargement de la page

filterNomVMOptions();

});

</script>

</body>

</html>

* Scripts JavaScript pour gérer le comportement dynamique de la page.
* document.addEventListener('DOMContentLoaded', function () {...}) : Exécute le code lorsque la page est complètement chargée.
* filterNomVMOptions() : Fonction pour filtrer les options de la liste des machines virtuelles en fonction des sélections.
* Ajoute des écouteurs d'événements pour mettre à jour les options lorsque les sélections changent.

### Ajouter des options

Pour ajouter de nouvelles options dans les sélections :

1. **Environnement :**
   * Ajoutez une nouvelle option dans le select avec id="environnement".

<option value="NouveauEnvironnement">Nouveau Environnement</option>

1. **Type de serveur :**
   * Ajoutez une nouvelle option dans le select avec id="type\_serveur".

<option value="NouveauServeur">Nouveau Serveur</option>

1. **Composant :**
   * Ajoutez une nouvelle option dans le select avec id="composant".

<option value="NouveauComposant">Nouveau Composant</option>

1. **Nom de la machine virtuelle :**
   * Ajoutez une nouvelle option dans le select avec id="NomVM", en spécifiant les attributs data-environment, data-os, et data-component.

<option value="NouveauNomVM" data-environment="NouveauEnvironnement" data-os="NouveauServeur" data-component="NouveauComposant">NouveauNomVM</option>

Partie Création VM Nutanix :

### Feuilles de style

<!-- Feuilles de style -->

<link rel="stylesheet" href="style.css">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

* <link rel="stylesheet" href="style.css">: Lie un fichier CSS local appelé style.css pour les styles personnalisés.
* <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">: Lie la feuille de style Bootstrap version 5.3.0 depuis un CDN (Content Delivery Network).

### En-tête de la page

<header class="mb-4 d-flex justify-content-between align-items-center">

<img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo mx-auto d-block">

</header>

* <header class="mb-4 d-flex justify-content-between align-items-center">: Crée un en-tête avec des classes Bootstrap pour les marges (mb-4), et des classes flexbox pour la disposition (d-flex justify-content-between align-items-center).
* <img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo mx-auto d-block">: Ajoute une image de logo centrée horizontalement avec les classes Bootstrap mx-auto d-block.

### Contenu principal

<main>

<div class="container">

<div class="details-box p-4">

<!-- Titre principal -->

<h1 class="text-center mb-4">CRÉATION DE MACHINE VIRTUELLE</h1>

<!-- Bouton de retour à l'accueil -->

<a href="accueil.html" class="submit-btn">Accueil</a>

<!-- Bouton toggle pour les détails -->

<div class="btn main-btn bounce-btn mx-auto d-block mb-4" id="btn1">

<ion-icon name="chevron-down"></ion-icon>

</div>

* <main>: Balise principale pour le contenu de la page.
* <div class="container">: Conteneur Bootstrap pour un alignement et un espacement centralisés.
* <div class="details-box p-4">: Un conteneur avec des classes personnalisées details-box et p-4 pour le padding.
* <h1 class="text-center mb-4">CRÉATION DE MACHINE VIRTUELLE</h1>: Un titre principal centré avec une marge inférieure.
* <a href="accueil.html" class="submit-btn">Accueil</a>: Un lien hypertexte stylisé comme un bouton pour retourner à l'accueil.
* <div class="btn main-btn bounce-btn mx-auto d-block mb-4" id="btn1">: Un bouton stylisé pour afficher/masquer les détails avec un effet de rebond et centré.

### Section des détails et formulaire de création

<!-- Section des détails -->

<div class="details" id="details1">

<!-- Formulaire de création -->

<form action="process.php" method="post" id="creationForm">

<div class="details-box1 p-4 mb-4">

<!-- Titre du formulaire -->

<h2 class="text-center mb-4">NUTANIX</h2>

<!-- Légende pour le groupe de sélection -->

<fieldset class="mb-4">

<legend>Sélectionnez les options de la machine virtuelle :</legend>

<!-- Grille pour les champs de saisie -->

<div class="row">

* <div class="details" id="details1">: Section des détails avec un ID pour le ciblage JavaScript.
* <form action="process.php" method="post" id="creationForm">: Formulaire HTML pour soumettre les données à process.php avec la méthode POST.
* <div class="details-box1 p-4 mb-4">: Conteneur avec des styles personnalisés et padding.
* <h2 class="text-center mb-4">NUTANIX</h2>: Titre du formulaire centré avec une marge inférieure.
* <fieldset class="mb-4">: Groupe de champs avec une légende.
* <legend>Sélectionnez les options de la machine virtuelle :</legend>: Légende décrivant le groupe de champs.
* <div class="row">: Ligne Bootstrap pour créer une grille de champs.

### Conteneur de Détails

<div class="details" id="details1">

* **<div class="details" id="details1">** : Conteneur principal pour les détails du formulaire.

### Formulaire de Création

<form action="process.php" method="post" id="creationForm">

* **<form action="process.php" method="post" id="creationForm">** : Début du formulaire qui envoie les données à process.php en utilisant la méthode POST.

### Section des Détails

<div class="details-box1 p-4 mb-4">

<h2 class="text-center mb-4">NUTANIX</h2>

* **<div class="details-box1 p-4 mb-4">** : Conteneur pour les détails avec un espacement et des marges.
* **<h2 class="text-center mb-4">NUTANIX</h2>** : Titre du formulaire, centré.

### Légende et Groupe de Sélection

<fieldset class="mb-4">

<legend>Sélectionnez les options de la machine virtuelle :</legend>

* **<fieldset class="mb-4">** : Groupe de champs avec une légende.
* **<legend>Sélectionnez les options de la machine virtuelle :</legend>** : Légende pour le groupe de champs.

### Grille de Champs de Saisie

Chaque section contient un champ de saisie ou de sélection. Voici quelques exemples :

#### Sélection de l'Environnement

<div class="col-md-4 mb-3">

<label>

<select name="environnement" class="form-select" id="environnement" required>

<option value="">Sélectionnez un environnement</option>

<option value="Developpement">Développement</option>

<!-- Autres options -->

</select>

</label>

</div>

* **<div class="col-md-4 mb-3">** : Conteneur pour le champ avec une largeur de 4 colonnes et une marge inférieure.
* **<label>** : Étiquette pour le champ.
* **<select name="environnement" class="form-select" id="environnement" required>** : Liste déroulante pour sélectionner un environnement.

#### Champ pour le Code Projet

<div class="col-md-4 mb-3">

<label>

<input type="text" name="code\_projet" class="form-control" placeholder="Code projet :" required>

</label>

</div>

* **<input type="text" name="code\_projet" class="form-control" placeholder="Code projet :" required>** : Champ de texte pour entrer le code projet.

### Sélection du Nom de la Machine Virtuelle

<div class="col-md-4 mb-3">

<label>

<select name="Nom\_VM" class="form-select" id="NomVM" required>

<option value="">Nom de la Machine Virtuelle</option>

<!-- Options filtrées par environnement -->

</select>

</label>

</div>

* **<select name="Nom\_VM" class="form-select" id="NomVM" required>** : Liste déroulante pour sélectionner le nom de la machine virtuelle avec des options filtrées par environnement.

### Bouton de Soumission

<button type="submit" class="submit-btn mx-auto d-block mb-4">Soumettre</button>

* **<button type="submit" class="submit-btn mx-auto d-block mb-4">Soumettre</button>** : Bouton pour soumettre le formulaire, centré avec des marges.

### Exemple Complet

<div class="details" id="details1">

<form action="process.php" method="post" id="creationForm">

<div class="details-box1 p-4 mb-4">

<h2 class="text-center mb-4">NUTANIX</h2>

<fieldset class="mb-4">

<legend>Sélectionnez les options de la machine virtuelle :</legend>

<div class="row">

<!-- Plusieurs champs de saisie et de sélection ici -->

</div>

</fieldset>

</div>

<button type="submit" class="submit-btn mx-auto d-block mb-4">Soumettre</button>

</form>

</div>

* **Structure Globale** : Conteneur principal avec un formulaire comprenant des champs pour saisir les détails de la machine virtuelle et un bouton de soumission.

### Champs de saisie dans le formulaire

<div class="col-md-4 mb-3">

<label>

<select name="environnement" class="form-select" id="environnement" required>

<option value="">Sélectionnez un environnement</option>

<option value="Developpement">Développement</option>

<option value="Recette">Recette</option>

<option value="Preproduction">Préproduction</option>

<option value="Infrastructure">Infrastructure</option>

<option value="Production">Production</option>

</select>

</label>

</div>

* <div class="col-md-4 mb-3">: Colonne Bootstrap de taille moyenne (4 sur 12) avec une marge inférieure.
* <label>: Conteneur pour les champs de formulaire.
* <select name="environnement" class="form-select" id="environnement" required>: Menu déroulant pour sélectionner l'environnement, avec un ID pour le ciblage JavaScript et une classe Bootstrap form-select.
* <option value="">Sélectionnez un environnement</option>: Option par défaut sans valeur.
* Plusieurs <option value="...">: Options disponibles dans le menu déroulant.

### Exemple de champ de texte

<div class="col-md-4 mb-3">

<label>

<input type="text" name="code\_projet" class="form-control" placeholder="Code projet :" required>

</label>

</div>

* <input type="text" name="code\_projet" class="form-control" placeholder="Code projet :" required>: Champ de texte avec un placeholder et une classe Bootstrap form-control.

### Sélection du nom de la machine virtuelle

<div class="col-md-4 mb-3">

<label>

<select name="Nom\_VM" class="form-select" id="NomVM" required>

<option value="">Nom de la Machine Virtuelle</option>

<!-- Options filtrées par environnement -->

<!-- Options pour Developpement -->

<option value="DEVXXXSQL-VLIXX" data-environment="Developpement" data-os="RedHat\_Entreprise" data-component="SQL">DEVXXXSQL-VLIXX</option>

<!-- ... Autres options ... -->

</select>

</label>

</div>

* <select name="Nom\_VM" class="form-select" id="NomVM" required>: Menu déroulant pour sélectionner le nom de la machine virtuelle.
* <option value="">Nom de la Machine Virtuelle</option>: Option par défaut sans valeur.
* Plusieurs options avec des attributs data-\* pour le filtrage dynamique.

### Soumission du formulaire

<!-- Bouton de soumission du formulaire -->

<button type="submit" class="submit-btn mx-auto d-block mb-4">Soumettre</button>

* <button type="submit" class="submit-btn mx-auto d-block mb-4">Soumettre</button>: Bouton de soumission centré avec une marge inférieure.

### Scripts

<!-- Script Bootstrap -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

<!-- Script Ionicons -->

<script src="https://cdn.jsdelivr.net/npm/ionicons@5.5.4/dist/ionicons/ionicons.esm.js" type="module"></script>

<script src="https://cdn.jsdelivr.net/npm/ionicons@5.5.4/dist/ionicons/ionicons.js" nomodule></script>

<!-- Script de filtrage -->

<script>

// Fonction pour mettre à jour les options de periode\_maintenance en fonction de l'environnement sélectionné

function updateSelectOptions(selectedValue, selectName) {

const periodeMaintenanceSelect = document.getElementById('periodeMaintenance');

const options = periodeMaintenanceSelect.querySelectorAll('option');

options.forEach(option => {

const parentValue = option.getAttribute('data-parent');

if (parentValue === selectedValue || selectedValue === '') {

option.style.display = 'block';

} else {

option.style.display = 'none';

}

});

// Réinitialisation du select après filtrage

periodeMaintenanceSelect.value = '';

}

// Écouteur d'événement pour le changement de l'environnement

const environnementSelect = document.getElementById('environnement');

environnementSelect.addEventListener('change', function() {

const selectedValue = this.value;

updateSelectOptions(selectedValue, 'environnement');

});

// Initialisation pour cacher les options non liées au premier select

updateSelectOptions('', 'environnement');

function filterOptions() {

const env = document.getElementById('environnement').value;

const os = document.getElementById('type\_serveur').value;

const component = document.getElementById('composant').value;

const options = document.querySelectorAll('#NomVM option');

options.forEach(option => {

const optionEnv = option.getAttribute('data-environment');

const optionOs = option.getAttribute('data-os');

const optionComponent = option.getAttribute('data-component');

if (

(env && optionEnv !== env) ||

(os && optionOs !== os) ||

(component && optionComponent !== component)

) {

option.style.display = 'none';

} else {

option.style.display = 'block';

}

});

}

// Écouteurs d'événements pour le filtrage des options

document.getElementById('environnement').addEventListener('change', filterOptions);

document.getElementById('type\_serveur').addEventListener('change', filterOptions);

document.getElementById('composant').addEventListener('change', filterOptions);

// Initialisation du filtrage

filterOptions();

</script>

* Plusieurs scripts pour intégrer Bootstrap et Ionicons.
* Script personnalisé pour filtrer les options dans les menus déroulants en fonction des sélections d'autres menus.

Ce bloc de code HTML représente une série d'options pour un menu déroulant dans un formulaire. Chaque <option> définit une configuration de machine virtuelle potentielle avec plusieurs attributs personnalisés (data-environment, data-os, data-component) qui spécifient différents aspects de la configuration de la machine virtuelle :

<option value="DEVXXXSQL-VLIXX" data-environment="Developpement" data-os="RedHat\_Entreprise" data-component="SQL">DEVXXXSQL-VLIXX</option>

<option value="DEVXXXSQL-22XX" data-environment="Developpement" data-os="Windows\_Server" data-component="SQL">DEVXXXSQL-22XX</option>

<option value="DEVXXXAPP-VLIXX" data-environment="Developpement" data-os="RedHat\_Entreprise" data-component="APP">DEVXXXAPP-VLIXX</option>

<option value="DEVXXXAPP-22XX" data-environment="Developpement" data-os="Windows\_Server" data-component="APP">DEVXXXAPP-22XX</option>

<option value="DEVXXXWEB-VLIXX" data-environment="Developpement" data-os="RedHat\_Entreprise" data-component="WEB">DEVXXXWEB-VLIXX</option>

<option value="DEVXXXWEB-22XX" data-environment="Developpement" data-os="Windows\_Server" data-component="WEB">DEVXXXWEB-22XX</option>

<!-- Option pour Recette -->

<option value="RECXXXSQL-VLIXX" data-environment="Recette" data-os="RedHat\_Entreprise" data-component="SQL">RECXXXSQL-VLIXX</option>

<option value="RECXXXSQL-22XX" data-environment="Recette" data-os="Windows\_Server" data-component="SQL">RECXXXSQL-22XX</option>

<option value="RECXXXAPP-VLIXX" data-environment="Recette" data-os="RedHat\_Entreprise" data-component="APP">RECXXXAPP-VLIXX</option>

<option value="RECXXXAPP-22XX" data-environment="Recette" data-os="Windows\_Server" data-component="APP">RECXXXAPP-22XX</option>

<option value="RECXXXWEB-VLIXX" data-environment="Recette" data-os="RedHat\_Entreprise" data-component="WEB">RECXXXWEB-VLIXX</option>

<option value="RECXXXWEB-22XX" data-environment="Recette" data-os="Windows\_Server" data-component="WEB">RECXXXWEB-22XX</option>

<!-- Options pour Infrastructure -->

<option value="INFXXXSQL-VLIXX" data-environment="Infrastructure" data-os="RedHat\_Entreprise" data-component="SQL">INFXXXSQL-VLIXX</option>

<option value="INFXXXSQL-22XX" data-environment="Infrastructure" data-os="Windows\_Server" data-component="SQL">INFXXXSQL-22XX</option>

<option value="INFXXXAPP-VLIXX" data-environment="Infrastructure" data-os="RedHat\_Entreprise" data-component="APP">INFXXXAPP-VLIXX</option>

<option value="INFXXXAPP-22XX" data-environment="Infrastructure" data-os="Windows\_Server" data-component="APP">INFXXXAPP-22XX</option>

<option value="INFXXXWEB-VLIXX" data-environment="Infrastructure" data-os="RedHat\_Entreprise" data-component="WEB">INFXXXWEB-VLIXX</option>

<option value="INFXXXWEB-22XX" data-environment="Infrastructure" data-os="Windows\_Server" data-component="WEB">INFXXXWEB-22XX</option>

<!-- Options pour Preproduction -->

<option value="PPRXXXSQL-VLIXX" data-environment="Preproduction" data-os="RedHat\_Entreprise" data-component="SQL">PPRXXXSQL-VLIXX</option>

<option value="PPRXXXSQL-22XX" data-environment="Preproduction" data-os="Windows\_Server" data-component="SQL">PPRXXXSQL-22XX</option>

<option value="PPRXXXAPP-VLIXX" data-environment="Preproduction" data-os="RedHat\_Entreprise" data-component="APP">PPRXXXAPP-VLIXX</option>

<option value="PPRXXXAPP-22XX" data-environment="Preproduction" data-os="Windows\_Server" data-component="APP">PPRXXXAPP-22XX</option>

<option value="PPRXXXWEB-VLIXX" data-environment="Preproduction" data-os="RedHat\_Entreprise" data-component="WEB">PPRXXXWEB-VLIXX</option>

<option value="PPRXXXWEB-22XX" data-environment="Preproduction" data-os="Windows\_Server" data-component="WEB">PPRXXXWEB-22XX</option>

<!-- Options pour Production -->

<option value="PROXXXSQL-VLIXX" data-environment="Production" data-os="RedHat\_Entreprise" data-component="SQL">PROXXXSQL-VLIXX</option>

<option value="PROXXXSQL-22XX" data-environment="Production" data-os="Windows\_Server" data-component="SQL">PROXXXSQL-22XX</option>

<option value="PROXXXAPP-VLIXX" data-environment="Production" data-os="RedHat\_Entreprise" data-component="APP">PROXXXAPP-VLIXX</option>

<option value="PROXXXAPP-22XX" data-environment="Production" data-os="Windows\_Server" data-component="APP">PROXXXAPP-22XX</option>

<option value="PROXXXWEB-VLIXX" data-environment="Production" data-os="RedHat\_Entreprise" data-component="WEB">PROXXXWEB-VLIXX</option>

<option value="PROXXXWEB-22XX" data-environment="Production" data-os="Windows\_Server" data-component="WEB">PROXXXWEB-22XX</option>

Chaque <option> contient les attributs suivants :

* value: La valeur qui sera soumise lorsque cette option est sélectionnée.
* data-environment: Indique l'environnement dans lequel cette configuration de machine virtuelle est utilisée (par exemple, Développement, Recette, Production, etc.).
* data-os: Spécifie le système d'exploitation de la machine virtuelle (par exemple, RedHat Entreprise, Windows Server, etc.).
* data-component: Décrit le composant principal de la machine virtuelle (par exemple, SQL, APP, WEB, etc.).

Pour ajouter une nouvelle option à ce menu déroulant dans un formulaire HTML, vous devez simplement ajouter un nouveau bloc <option> avec les attributs nécessaires. Voici comment procéder :

1. **Copiez un modèle existant :** Copiez l'une des lignes existantes <option> pour vous assurer que tous les attributs requis sont inclus.
2. **Modifiez les valeurs :** Modifiez les valeurs des attributs value, data-environment, data-os, et data-component selon vos besoins. Assurez-vous que la valeur dans la balise <option> (le texte entre les balises d'ouverture et de fermeture <option>...</option>) est significative pour les utilisateurs.
3. **Insérez la nouvelle option :** Ajoutez la nouvelle ligne <option> à l'emplacement approprié dans votre code HTML, généralement à l'intérieur de la balise <select> qui définit le menu déroulant.

Voici un exemple de comment ajouter une nouvelle option pour une application fictive dans l'environnement de Préproduction avec RedHat Entreprise comme système d'exploitation :

<option value="PPRXXXAPP-33XX" data-environment="Preproduction" data-os="RedHat\_Entreprise" data-component="APP">PPRXXXAPP-33XX</option>

Dans ce cas :

* value: PPRXXXAPP-33XX sera la valeur soumise lorsque cette option est sélectionnée.
* data-environment: Preproduction indique que cette option est pour l'environnement de Préproduction.
* data-os: RedHat\_Entreprise spécifie le système d'exploitation RedHat Entreprise.
* data-component: APP représente le composant de l'application.

### Points à Garder à l'Esprit

* **CSS et Design** : Assurez-vous que le nouveau fieldset respecte le style et la mise en page globale de votre page en utilisant les classes CSS existantes.
* **JavaScript** : Si nécessaire, adaptez le comportement des boutons avec des fonctions JavaScript pour des actions plus complexes ou dynamiques.
* **Maintenabilité** : Documentez toujours les modifications apportées à votre code pour faciliter la maintenance future et la compréhension par d'autres développeurs.

CreerVMNutanix. Explication :Une image contenant texte, logiciel, Système d’exploitation, Icône d’ordinateur

Description générée automatiquement

<head>

* Cette balise ouvre la section <head> du document , qui contient des métadonnées et des références aux ressources externes utilisées par la page

<title>Création de Machine Virtuelle</title>

* Cette balise <title> définit le titre de la page qui s'affiche dans l'onglet du navigateur ou dans la barre de titre de la fenêtre du navigateur.

<link rel="stylesheet" href="style.css">

* Ce tag <link> lie une feuille de style externe (style.css) à la page , permettant de définir le style et la mise en forme de la page.

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

* Ce <link> tag lie la feuille de style Bootstrap à partir d'un CDN (Content Delivery Network), fournissant des styles prédéfinis pour les composants et la mise en page basée sur Bootstrap.

</head>

* Ferme la section <head> du document .

<body>

* Ouvre la section <body> du document , qui contient tout le contenu visible de la page.

<header class="mb-4 d-flex justify-content-between align-items-center">

* Cette balise <header> ouvre la section d'en-tête de la page, utilisée généralement pour inclure des éléments tels que le logo, les liens de navigation, etc. Les classes Bootstrap (mb-4, d-flex, justify-content-between, align-items-center) définissent le style de mise en page (marges, flexibilité et alignement).

<img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo mx-auto d-block">

* Cet <img> tag affiche une image avec le chemin spécifié (src="assets/april-mutuelle-logo.png"), l'attribut alt fournit un texte alternatif pour l'image à des fins d'accessibilité, et les classes Bootstrap (logo, mx-auto, d-block) définissent le style de l'image (marges, centrage horizontal).

</header>

* Ferme la section d'en-tête <header> de la page.

<main>

* Ouvre la section <main> qui contient le contenu principal de la page.

<div class="container">

* Ouvre un conteneur Bootstrap qui centre et aligne le contenu selon la grille Bootstrap.

<div class="details-box p-4">

* Cette <div> est utilisée pour envelopper un contenu avec une classe spécifique (details-box) qui lui donne un style particulier (remplissage de 4 côtés avec p-4).

<h1 class="text-center mb-4">CRÉATION DE MACHINE VIRTUELLE</h1>

* <h1> est un titre de niveau 1 qui affiche le texte "CRÉATION DE MACHINE VIRTUELLE" centré (text-center) avec une marge inférieure de 4 (mb-4).

<a href="accueil. " class="submit-btn">Accueil</a>

* <a> est utilisé pour créer un lien hypertexte vers accueil. avec le texte "Accueil". La classe submit-btn applique le style d'un bouton de soumission de formulaire.

<div class="btn main-btn bounce-btn mx-auto d-block" id="btn1">

<ion-icon name="chevron-down"></ion-icon>

</div>

* Cette <div> crée un bouton stylisé (btn, main-btn, bounce-btn) avec une icône <ion-icon> (qui nécessite la bibliothèque Ionicons) représentant une flèche vers le bas.

<div class="details" id="details1">

* Ouvre une <div> avec la classe details et l'identifiant details1, utilisée pour contenir des détails spécifiques qui peuvent être affichés ou masqués.

<form action="process.php" method="post" id="creationForm">

* <form> démarre un formulaire qui envoie les données saisies à process.php via la méthode POST lorsque le formulaire est soumis. L'attribut id="creationForm" permet de référencer ce formulaire avec JavaScript ou CSS.

<div class="details-box1 p-4 mb-4">

* <div> enveloppe un contenu spécifique avec la classe details-box1, lui donnant un style particulier (remplissage de 4 côtés avec p-4 et une marge inférieure de 4 avec mb-4).

<h2 class="text-center mb-4">NUTANIX</h2>

* <h2> est un titre de niveau 2 qui affiche le texte "NUTANIX" centré (text-center) avec une marge inférieure de 4 (mb-4).

<fieldset class="mb-4">

<legend>Sélectionnez les options de la machine virtuelle :</legend>

* <fieldset> groupe des contrôles de formulaire avec une légende (Sélectionnez les options de la machine virtuelle :). La classe mb-4 ajoute une marge inférieure de 4.

<div class="row">

* <div> commence une ligne Bootstrap (row), qui organise les éléments en colonnes.

<div class="col-md-4 mb-3">

<label>

<select name="environnement" class="form-select" id="environnement" required>

<option value="">Sélectionnez un environnement</option>

<option value="Developpement">Développement</option>

<option value="Recette">Recette</option>

<option value="Preproduction">Préproduction</option>

<option value="Infrastructure">Infrastructure</option>

<option value="Production">Production</option>

</select>

</label>

</div>

* <div> crée une colonne Bootstrap de largeur moyenne (col-md-4) avec une marge inférieure de 3 (mb-3). À l'intérieur, un <select> affiche une liste déroulante (form-select) pour sélectionner un environnement avec des options spécifiques.

<label>

<input type="text" name="code\_projet" class="form-control" placeholder="Code projet :" required>

</label>

* <input> est utilisé pour créer un champ de saisie (type="text") pour le code de projet avec une classe Bootstrap form-control pour le style et un texte de placeholder "Code projet :" pour guider l'utilisateur.

<div class="col-md-4 mb-3">

<!-- ... Autres champs de formulaire similaires ... -->

</div>

* D'autres <div> suivent avec des colonnes Bootstrap (col-md-4) pour d'autres champs de formulaire (par exemple, type de serveur, type de machine, composant, etc.), chacun avec des balises <select> ou <input> appropriées pour la saisie ou la sélection de données.

<button type="submit" class="submit-btn mx-auto d-block mb-4">Soumettre</button>

* <button> crée un bouton de soumission de formulaire (type="submit") avec une classe Bootstrap submit-btn pour le style (centré avec mx-auto, avec une marge inférieure de 4 mb-4).

</fieldset>

* Ferme le <fieldset> qui groupe les contrôles de formulaire.

</form>

* Ferme le formulaire .

</div>

* Ferme la <div> qui enveloppe le contenu principal du formulaire.

</main>

* Ferme la section <main> qui contient le contenu principal de la page.

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

* Ce <script> tag inclut le fichier JavaScript de Bootstrap à partir d'un CDN (Content Delivery Network), qui est nécessaire pour activer certains composants JavaScript de Bootstrap.

<script src="script.js"></script>

* Ce <script> tag inclut un fichier JavaScript local (script.js) qui peut contenir des scripts personnalisés pour manipuler le comportement de la page.

</body>

* Ferme la section <body> du document .

</ >

* Ferme le document .

suprVMNutanix. Explication : <head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="style.css">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

<title>Document</title>

</head>

* La section <head> contient des métadonnées et des références aux ressources utilisées par la page. Voici une explication détaillée :
  + <meta charset="UTF-8"> : Définit l'encodage des caractères utilisé par le document comme UTF-8, assurant que tous les caractères peuvent être correctement affichés.
  + <meta http-equiv="X-UA-Compatible" content="IE=edge"> : Indique aux anciennes versions d'Internet Explorer d'utiliser le mode de rendu le plus récent disponible.
  + <meta name="viewport" content="width=device-width, initial-scale=1.0"> : Définit la largeur de l'affichage sur les appareils (responsive) et l'échelle initiale de la page.
  + <link rel="stylesheet" href="style.css"> : Lie une feuille de style locale (style.css) pour définir le style de la page.
  + <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet"> : Lie la feuille de style Bootstrap à partir d'un CDN pour utiliser les styles prédéfinis de Bootstrap.
  + <title>Document</title> : Définit le titre de la page qui apparaît dans l'onglet du navigateur.

<style>

.submit-btn {

padding: 10px 20px;

background-color: #003B62;

color: white;

border: none;

border-radius: 5px;

font-size: 16px;

cursor: pointer;

animation: bounce 1s infinite;

}

@keyframes bounce {

0%, 100% {

transform: translateY(0);

}

50% {

transform: translateY(-10px);

}

}

</style>

* La balise <style> définit des styles CSS internes qui s'appliquent localement à la page :
  + .submit-btn : Classe CSS appliquée aux boutons avec un style de fond, couleur de texte, etc.
  + @keyframes bounce : Définit une animation CSS (bounce) qui fait rebondir le bouton de soumission (submit-btn).

<body>

<header>

<img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo">

</header>

<main>

<div class="wrapper">

<div class="details-box">

<!-- Contenu principal de la page -->

</div>

</div>

</main>

<footer class="lienContact">

<p>&copy; 2024 - <a href="https://mydesk.april.com/index.php?token=N9L3jFYVmmc4b6Z0jbI%2BPw%3D%3D&timestamp=1718876673839&name=com.portail.65b36ec12f9a6#!?page=5e43cc1124518">Pour toute demande</a></p>

</footer>

<script src="accueil.js"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

</body>

* La section <body> contient tout le contenu visible de la page :
  + <header> : En-tête de la page contenant le logo de l'entreprise.
  + <main> : Contient le contenu principal de la page.
    - <div class="wrapper"> : Enveloppe le contenu principal avec une classe spécifique.
      * <div class="details-box"> : Conteneur pour une boîte de détails spécifiques.
        + Le reste du contenu comme les titres, les boutons, les formulaires, etc., est inclus ici.
  + <footer class="lienContact"> : Pied de page contenant des informations de copyright et un lien vers une page externe pour les demandes.
  + <script src="accueil.js"></script> : Lie un fichier JavaScript local (accueil.js) pour des scripts personnalisés.
  + <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script> : Lie le fichier JavaScript de Bootstrap à partir d'un CDN pour activer les composants JavaScript de Bootstrap.

Les autres pages Suppression et Création Explication :

Fichier : SuprVNETNutanix.html, suprVLANPHPiPAM.html, suprUnbindREVERSE\_PROXY.html, suprServREVERSE\_PROXY.html, suprServiceREVERSE\_PROXY.html, suprResPALO\_ALTO.html, suprReseauPHPiPAM.html, suprReglFWPALO\_ALTO.html, suprPolicyREVERSE\_PROXY.html, suprobjtPALO\_ALTO.html, suprGRPobjtPALO\_ALTO.html, suprEntréMachinePHPiPAM.html, suprDisqueNutanix.html, suprCSVSREVERSE\_PROXY.html, suprCertifREVERSE\_PROXY.html, RegistDNSpublic.html, creerVNETNutanix.html, creerVLANPHPiPAM.html, creerUnbindREVERSE\_PROXY.html, creerServREVERSE\_PROXY.html, creerServiceREVERSE\_PROXY.html, creerResPALO\_ALTO.html, creerReseauPHPiPAM.html, creerReglFWPALO\_ALTO.html, creerPolicyREVERSE\_PROXY.html, creerobjtPALO\_ALTO.html, creerGRPobjtPALO\_ALTO.html, creerEntréMachinePHPiPAM.html, creerDisqueNutanix.html, creerCSVSREVERSE\_PROXY.html, creerCertifREVERSE\_PROXY.html

1. **<!DOCTYPE html>**
   * C'est la déclaration du type de document. Elle informe le navigateur que le document est en HTML5.
2. **<html lang="fr">**
   * Ouvre la balise HTML et définit la langue du document comme le français (lang="fr").
3. **<head>**
   * Ouvre la balise head. Cette section contient des informations méta et des liens vers des fichiers externes comme les styles CSS.
4. **<meta charset="UTF-8">**
   * Spécifie l'encodage des caractères du document comme UTF-8, ce qui permet de représenter presque tous les caractères des langues du monde.
5. **<meta http-equiv="X-UA-Compatible" content="IE=edge">**
   * Indique à Internet Explorer d'utiliser la dernière version de son moteur de rendu.
6. **<meta name="viewport" content="width=device-width, initial-scale=1.0">**
   * Configure le viewport pour assurer que le site est bien affiché sur tous les appareils (responsiveness).
7. **<link rel="stylesheet" href="style.css">**
   * Lien vers un fichier CSS externe nommé style.css, qui contient des règles de style personnalisées.
8. **<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">**
   * Lien vers une feuille de style CSS externe hébergée par Bootstrap, une bibliothèque de design.
9. **<title>Document</title>**
   * Définit le titre de la page qui apparaît dans l'onglet du navigateur.
10. **</head>**
    * Ferme la balise head.
11. **<style>**
    * Ouvre une balise style pour ajouter des styles CSS directement dans le document HTML.
12. **.submit-btn { ... }**
    * Définition de la classe submit-btn qui stylise les boutons de soumission :
      + padding: 10px 20px; : Ajoute un espacement interne au bouton.
      + background-color: #003B62; : Définit la couleur de fond du bouton.
      + color: white; : Définit la couleur du texte en blanc.
      + border: none; : Supprime la bordure par défaut.
      + border-radius: 5px; : Ajoute des coins arrondis.
      + font-size: 16px; : Définit la taille de la police.
      + cursor: pointer; : Change le curseur en une main pointée.
      + animation: bounce 1s infinite; : Applique une animation de rebond.
13. **@keyframes bounce { ... }**
    * Définition de l'animation bounce :
      + 0%, 100% { transform: translateY(0); } : La position de départ et de fin ne bouge pas verticalement.
      + 50% { transform: translateY(-10px); } : À mi-chemin, l'élément se déplace de 10 pixels vers le haut.
14. **</style>**
    * Ferme la balise style.
15. **<body>**
    * Ouvre la balise body, où commence le contenu visible de la page.
16. **<header>**
    * Ouvre la balise header pour contenir les éléments d'en-tête.
17. **<img src="assets/april-mutuelle-logo.png" alt="Logo" class="logo">**
    * Affiche une image (logo) située dans le dossier assets. Le texte alternatif alt="Logo" est utilisé si l'image ne peut pas être affichée.
18. **</header>**
    * Ferme la balise header.
19. **<main>**
    * Ouvre la balise main, contenant le contenu principal de la page.
20. **<div class="wrapper">**
    * Ouvre une div (section) avec la classe wrapper, utilisée pour envelopper le contenu et éventuellement ajouter des styles spécifiques.
21. **<div class="details-box">**
    * Ouvre une div avec la classe details-box pour contenir les détails spécifiques.
22. **<h1>SUPPRESSION</h1>**
    * Affiche un titre de niveau 1 avec le texte "SUPPRESSION".
23. **<div class="btn main-btn bounce-btn" id="btn1">**
    * Ouvre une div avec les classes btn, main-btn, et bounce-btn, et l'ID btn1, pour un bouton animé.
24. **<ion-icon name="chevron-down"></ion-icon>**
    * Utilise un icône provenant de la bibliothèque Ionicons.
25. **</div>**
    * Ferme la div du bouton.
26. **<div class="details" id="details1">**
    * Ouvre une div avec la classe details et l'ID details1, contenant les détails à afficher ou cacher.
27. **<div class="wrapper">**
    * Ouvre une nouvelle div wrapper pour structurer les éléments internes.
28. **<form action="process.php" method="post">**
    * Ouvre un formulaire qui enverra les données à process.php en utilisant la méthode POST.
29. **<div class="details-box1">**
    * Ouvre une div avec la classe details-box1 pour contenir plus de détails.
30. **<h2>NUTANIX</h2>**
    * Affiche un titre de niveau 2 avec le texte "NUTANIX".
31. **<div class="details" id="detailsNutanix">**
    * Ouvre une div avec la classe details et l'ID detailsNutanix, contenant des informations spécifiques.
32. **<fieldset>**
    * Ouvre une balise fieldset pour grouper les éléments de formulaire liés.
33. **<legend>Selectionnez le Disque que vous voulez supprimer :<abbr title="Ce champ est obligatoire"></abbr></legend>**
    * Affiche une légende pour le fieldset avec une abréviation (abbr) indiquant un champ obligatoire.
34. **</fieldset>**
    * Ferme la balise fieldset.
35. **<nav class="navbar navbar-expand-lg bg-body-tertiary mb-4">**
    * Ouvre une balise de navigation avec des classes Bootstrap pour une barre de navigation responsive.
36. **<div class="container-fluid">**
    * Ouvre une div avec la classe container-fluid pour un conteneur flexible.
37. **<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarTogglerDemo02" aria-controls="navbarTogglerDemo02" aria-expanded="false" aria-label="Toggle navigation">**
    * Ouvre un bouton pour toggler (afficher/masquer) le menu de navigation sur les petits écrans.
38. **<span class="navbar-toggler-icon"></span>**
    * Affiche une icône de toggler.
39. **</button>**
    * Ferme le bouton toggler.
40. **<div class="collapse navbar-collapse" id="navbarTogglerDemo02">**
    * Ouvre une div pour contenir les éléments de la navigation qui seront affichés/masqués.
41. **<ul class="navbar-nav me-auto mb-2 mb-lg-0">**
    * Ouvre une liste non ordonnée (ul) pour les éléments de navigation.
42. **<li class="nav-item">**
    * Ouvre un élément de liste (li) pour un item de navigation.
43. **<a class="submit-btn" aria-current="page" href="accueil.html">Accueil</a>**
    * Ajoute un lien (a) avec la classe submit-btn, redirigeant vers "accueil.html".
44. **</li>**
    * Ferme l'élément de liste.
45. **</ul>**
    * Ferme la liste non ordonnée.
46. **<form class="d-flex position-relative" role="search">**
    * Ouvre un formulaire de recherche avec des classes pour la mise en forme.
47. **<input id="search-input" class="form-control me-2" type="search" placeholder="Search" aria-label="Search">**
    * Ajoute un champ de saisie pour la recherche avec un identifiant search-input.
48. **<div id="suggestions" class="suggestions"></div>**
    * Ajoute une div pour afficher les suggestions de recherche, avec l'ID suggestions.
49. **</form>**
    * Ferme le formulaire de recherche.
50. **</div>**
    * Ferme la div navbar-collapse.
51. **</div>**
    * Ferme la div container-fluid.
52. **</nav>**
    * Ferme la balise de navigation.
53. **<div class="text-item text-center">**
    * Ouvre une div centrée pour contenir un élément de texte.
54. **<button type="submit" class="submit-btn">Soumettre</button>**
    * Ajoute un bouton de soumission avec la classe submit-btn.
55. **</div>**
    * Ferme la div text-item.
56. **</div>**
    * Ferme la div detailsNutanix.
57. **</div>**
    * Ferme la div details-box1.
58. **</form>**
    * Ferme le formulaire.
59. **</div>**
    * Ferme la div details1.
60. **</main>**
    * Ferme la balise main.
61. **<footer class="lienContact">**
    * Ouvre une balise footer avec la classe lienContact.
62. **<p>&copy; 2024 - <a href="https://mydesk.april.com/index.php?token=N9L3jFYVmmc4b6Z0jbI%2BPw%3D%3D&timestamp=1718876673839&name=com.portail.65b36ec12f9a6#!?page=5e43cc1124518">Pour toute demande</a></p>**
    * Ajoute un paragraphe contenant du texte et un lien hypertexte.
63. **</footer>**
    * Ferme la balise footer.
64. **<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>**
    * Lien vers un fichier JavaScript externe de Bootstrap pour des fonctionnalités interactives.
65. **<script>**
    * Ouvre une balise script pour inclure du JavaScript directement dans le document.
66. **document.addEventListener('DOMContentLoaded', () => { ... });**
    * Ajoute un écouteur d'événement pour exécuter le code une fois que le document est complètement chargé.
67. **const searchInput = document.getElementById('search-input');**
    * Sélectionne l'élément avec l'ID search-input.
68. **const suggestionsContainer = document.getElementById('suggestions');**
    * Sélectionne l'élément avec l'ID suggestions.
69. **const words = [...];**
    * **Crée une liste de mots pour les suggestions de recherche.**
70. **searchInput.addEventListener('input', () => { ... });**
    * Ajoute un écouteur d'événement pour détecter les changements dans le champ de recherche.
71. **const query = searchInput.value.toLowerCase();**
    * Récupère la valeur du champ de recherche et la convertit en minuscules.
72. **suggestionsContainer.innerHTML = '';**
    * Vide le contenu des suggestions précédentes.
73. **if (query) { ... }**
    * Si le champ de recherche contient quelque chose, exécute le code à l'intérieur des accolades.
74. **const filteredWords = words.filter(word => word.toLowerCase().includes(query));**
    * Filtre les mots qui contiennent la chaîne de recherche.
75. **if (filteredWords.length) { ... } else { suggestionsContainer.style.display = 'none'; }**
    * Si des mots correspondants sont trouvés, affiche les suggestions, sinon cache le conteneur de suggestions.
76. **filteredWords.forEach(word => { ... });**
    * Pour chaque mot filtré, crée un élément div pour l'afficher comme suggestion.
77. **const div = document.createElement('div');**
    * Crée une nouvelle div pour chaque suggestion.
78. **div.classList.add('suggestion-item');**
    * Ajoute une classe à la div de suggestion.
79. **div.textContent = word;**
    * Définit le texte de la div comme le mot de la suggestion.
80. **div.addEventListener('click', () => { ... });**
    * Ajoute un écouteur d'événement pour remplir le champ de recherche lorsque la suggestion est cliquée.
81. **searchInput.value = word;**
    * Remplit le champ de recherche avec le mot de la suggestion.
82. **suggestionsContainer.style.display = 'none';**
    * Cache le conteneur de suggestions.
83. **suggestionsContainer.appendChild(div);**
    * Ajoute la div de suggestion au conteneur de suggestions.
84. **document.addEventListener('click', (e) => { ... });**
    * Ajoute un écouteur d'événement pour cacher les suggestions lorsque l'utilisateur clique en dehors du champ de recherche.
85. **if (!suggestionsContainer.contains(e.target) && e.target !== searchInput) { suggestionsContainer.style.display = 'none'; }**
    * Si le clic n'est pas sur le conteneur de suggestions ou le champ de recherche, cache le conteneur de suggestions.
86. **</script>**
    * Ferme la balise script.
87. **</body>**
    * Ferme la balise body.
88. **</html>**
    * Ferme la balise HTML.

### Ajouter des Options

Pour ajouter des options de suppression dans le formulaire, vous pouvez utiliser des balises <option> à l'intérieur d'une balise <select> dans votre formulaire. Voici un exemple de comment le faire :

<form action="process.php" method="post">

<div class="details-box1">

<h2>NUTANIX</h2>

<div class="details" id="detailsNutanix">

<fieldset>

<legend>Selectionnez le Disque que vous voulez supprimer :<abbr title="Ce champ est obligatoire"></abbr></legend>

<select name="disque">

<option value="disque1">Disque 1</option>

<option value="disque2">Disque 2</option>

<option value="disque3">Disque 3</option>

</select>

</fieldset>

<div class="text-item text-center">

<button type="submit" class="submit-btn">Soumettre</button>

</div>

</div>

</div>

</form>

Explication des nouvelles lignes :

* **<select name="disque">**
  + Crée une liste déroulante où l'utilisateur peut sélectionner un disque. Le name="disque" définit le nom de cette sélection, qui sera envoyé au serveur.
* **<option value="disque1">Disque 1</option>**
  + Crée une option dans la liste déroulante. L'utilisateur verra "Disque 1", et la valeur envoyée au serveur sera disque1.
* **<option value="disque2">Disque 2</option>**
  + Crée une deuxième option.
* **<option value="disque3">Disque 3</option>**
  + Crée une troisième optio