

Projet 3 Stage 2

Utilisation d'Ajax avec Javascript



Dans le cadre de mon stage, lors de ma mission concernant le développement d'une application de gestion des congés, j'ai découvert le protocole Ajax et je l'ai utilisé pour optimiser les performances de cette application.

Histoire :

Le langage de programmation JavaScript est apparu pour la première fois en 1996, dans le navigateur Web Netscape Navigator.

Le langage XML et l'interface de programmation Document Object Model (DOM) ont été normalisés par le World Wide Web Consortium en 1998.

Ajax est en fait introduit par XMLHttpRequest, initialement un composant ActiveX créé en 1998 par Microsoft pour leur application web Outlook Web Access, puis il a été ajouté à la norme ECMAScript relative au langage JavaScript et mis en œuvre sur la plupart des navigateurs du marché entre 2002 et 2005.

Le terme Ajax a été introduit par Jesse James Garrett (informaticien américain), le 18 février 2005, dans un article sur le site Web Adaptive Path3. Depuis, il a rapidement gagné en popularité.

Principe :

Asynchronous Javascript And Xml (AJAX) : cet acronyme désigne un nouveau type de conception de pages Web permettant l'actualisation de certaines données d'une page sans procéder au rechargement total de cette page.

Cette méthode de conception repose sur la combinaison de technologies déjà existantes : HTML/CSS, Javascript/DOM, XML et les requêtes HTTP.

Par exemple, on peut faire des LiveSearch (recherche en direct). C'est-à-dire que, en même temps que l'on tape des mots clés pour une recherche, on voit apparaître la liste des résultats (sans avoir à recharger une page). Ca n'est qu'un exemple, les applications de ce type de conception d'un site web sont innombrables et pour la plupart reste à inventer...

Parmi les applications utilisant AJAX on peut citer Gmail, Windows Live...

Fonctionnement :

Concrètement, un objet Javascript (l'objet XMLHttpRequest) permet d'envoyer une requête HTTP (POST, GET...) au serveur et de récupérer le résultat. La conséquence directe est que l'on peut s'affranchir du rechargement d'une page pour récupérer une information sur le serveur.

Lorsque ces technologies sont combinées dans le modèle AJAX, les applications web sont capables de réaliser des mises à jour rapides et incrémentales de l'interface utilisateur sans devoir recharger la page entière dans le navigateur.

Les applications fonctionnent plus rapidement et sont plus réactives aux actions de l'utilisatrice ou de l'utilisateur.

Ajax fonctionne sur tous les navigateurs Web courants : Google Chrome, Safari, Mozilla Firefox, Internet Explorer, Microsoft Edge, Opera, etc...

Utilisation lors du projet dans le cadre du stage:

Dans l'extrait de code suivant, j'ai utilisé l'AJAX pour appeler dynamiquement un fichier PHP à l'intérieur d'une modale BOOTSTRAP.

```
<!-- Gérer le nombre de jours -->
<div class="modal fade p-0" id="gestionModifJours" tabindex="-1" role="dialog" aria-labelledby="gestionModifJours" aria-hidden="true">
  <div class="modal-dialog">
    <form action="index.php?page=gestion_modif_jours" method="post" class="login_form modal_padding">
      <div class="modal-content modal_gestionModifJours">test</div>
    </form>
  </div>
</div>

<script type="text/javascript">
  // affichage de la modale de gestion du nombre de jours
  $(''.gestionModifJours').click(function(){
    // récupère la valeur de la variable
    var id_gestionModifJours = $(this).attr('data-id');
    $.ajax({
      // effectue la requête dans la page citée
      url:"app/conges_civils/inc/modal_gestionModifJours.php?idc="+id_gestionModifJours,
      // insère le résultat de la requête dans le conteneur (class) ".gestionModifJours"
      success:function(result){
        $(''.modal_gestionModifJours').html(result);
      }
    });
    // vidage de la modale à la fermeture
    $(''.modal_gestionModifJours').html('');
  });
</script>
```

Pourquoi ?

Cela a permis de charger le contenu de la modale uniquement lors de l'ouverture de celle-ci et non au chargement de la page, et donc de rendre ce chargement plus rapide.

C'est particulièrement intéressant lorsqu'une page contient beaucoup de modales dont le contenu change dynamiquement en fonction d'un paramètre (dans une boucle par exemple) car on peut créer une seule structure de modale et ensuite changer le contenu de la modale lors du clic en fonction de l'id.

Le gain de poids et donc la vitesse de chargement sont alors considérablement impactés.

La structure de la modale et le lien permettant de l'ouvrir sont inscrits dans la page d'appel, puis du code Javascript permet de déclencher le code AJAX lors du clic sur un div ayant une certaine Class.

Lors de ce clic, l'AJAX procède à l'inclusion d'un fichier ciblé via son url dans un div ayant une Class donnée.

Comme le code du fichier appelé est exécuté lors du clic, il est alors possible de faire des requêtes vers le serveur et la base de données et d'en afficher les résultats sans avoir à recharger la page (d'où le terme de chargement asynchrone).