

Différence entre :

- Cookies
- Session Storage
- Local Storage

Zarrouk.elyes@gmail.com

Introduction

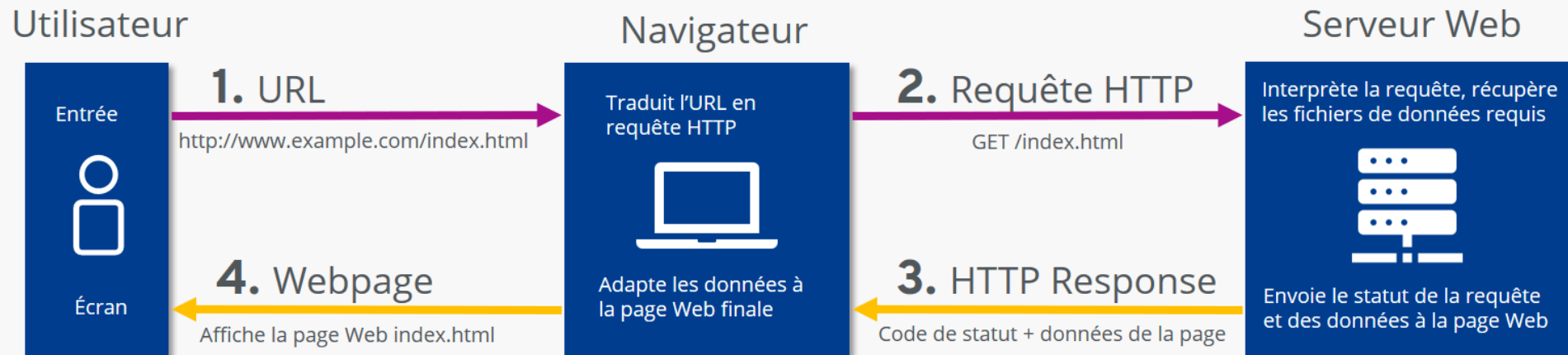
- HTTP signifie « **Hypertext Transfer Protocol** ».
- Ce protocole a été développé par Tim Berners-Lee avec d'autres concepts qui ont servi de base à la création du World Wide Web
- HTTP détermine comment la page est transmise du serveur au client.
- le Web utilisait des protocoles HTTP pour envoyer des messages (d'ailleurs, SSL est plus sécurisé, vous devriez utiliser HTTPS au lieu de HTTP).

Introduction

- Ces protocoles sont des protocoles sans état.
- Dans un protocole sans état, chaque requête ne stocke aucun état ou « information persistante » ; chaque demande est sa propre île et il n'a aucune idée des autres demandes.
- Avoir un protocole sans état optimise les performances, mais il y a aussi un problème : et si vous avez besoin de vous souvenir d'une session utilisateur ?
- Comment un serveur peut-il s'en souvenir si vous utilisez un protocole sans état ?

HTTP

Processus de communication HTTP



Cookie

- Date d'expiration peut être modifiée par le serveur ou le client
- N'est pas accessible si HttpOnly=True
- Supporte une sécurité de la couche transport (Secure Socket Layer SSL)
- Les données sont transférées dans chaque requête HTTP
- Taille maximale 4Kb

```
document.cookie = "cname=its_delicious";  
document.cookie = "description=its_awesome";  
console.log(document.cookie);
```

Web Storage

- Le Web Storage permet au navigateur de stocker des informations de façon locale, sans avoir à communiquer avec un quelconque serveur.
- L'utilisation des cookies, par exemple, n'était pas idéale : avec une taille limitée, ils n'étaient pas adaptés au stockage local. La solution la plus pratique était de stocker ces données en ligne, grâce à des appels AJAX : pratique lourde en ressources !
- Désormais, le Web Storage met en place deux espaces de stockage.
- Le premier, Session Storage, est destiné à la mémorisation de données ayant une faible durée de vie : celles-ci seront supprimées à la fermeture du navigateur.
- Local Storage, quant à lui, bénéficie d'une durée de vie plus longue (les données ne sont pas supprimées à la fermeture du navigateur et restent disponibles (théoriquement) pour toujours) et possède une portée étendue à toutes les pages d'un même domaine.
- Au sein d'un même site, on peut donc mettre en place un vrai espace de stockage. En plus, contrairement aux cookies, la limite de stockage est assez grande (au moins 5 Mb) et les informations ne sont jamais transférées vers le serveur.

LocalStorage

- Pas de Date d'expiration
- Lu seulement coté Client
- Ne Supporte une sécurité pas de la couche transport (Secure Socket Layer SSL)
- Les données ne sont pas transférées dans chaque requête HTTP
- Taille maximale 5Mb selon le navigateur

LocalStorage

```
localStorage.setItem('strawberry', 'pancake');  
localStorage.getItem('strawberry'); // pancake`  
localStorage.chocolate = 'waffle';  
localStorage.chocolate; // waffle  
localStorage['blueberry'] = 'donut';  
localStorage['blueberry']; // donut;  
delete localStorage.strawberry;
```


SessionStorage

- Données expirées si le navigateur est fermé
- Lu seulement coté Client
- Ne Supporte une sécurité pas de la couche transport (Secure Socket Layer SSL)
- Les données ne sont pas transférées dans chaque requête HTTP
- Taille maximale 5Mb selon le navigateur

SessionStorage

```
sessionStorage.setItem('strawberry', 'pancake');  
sessionStorage.getItem('strawberry'); // pancake`  
sessionStorage.chocolate = 'waffle';  
sessionStorage.chocolate; // waffle  
sessionStorage['blueberry'] = 'donut';  
sessionStorage['blueberry']; // donut;  
delete sessionStorage.strawberry;
```