

Tech Talk

présenté par Lyne

29-04-2024

# LE DÉFILEMENT SANS FIN

du clic au swipe, une fonctionnalité révolutionnaire.

# TODAY'S AGENDA

**1**

définition, historique et importance

**3**

référencement : les solutions et bonnes pratiques

**2**

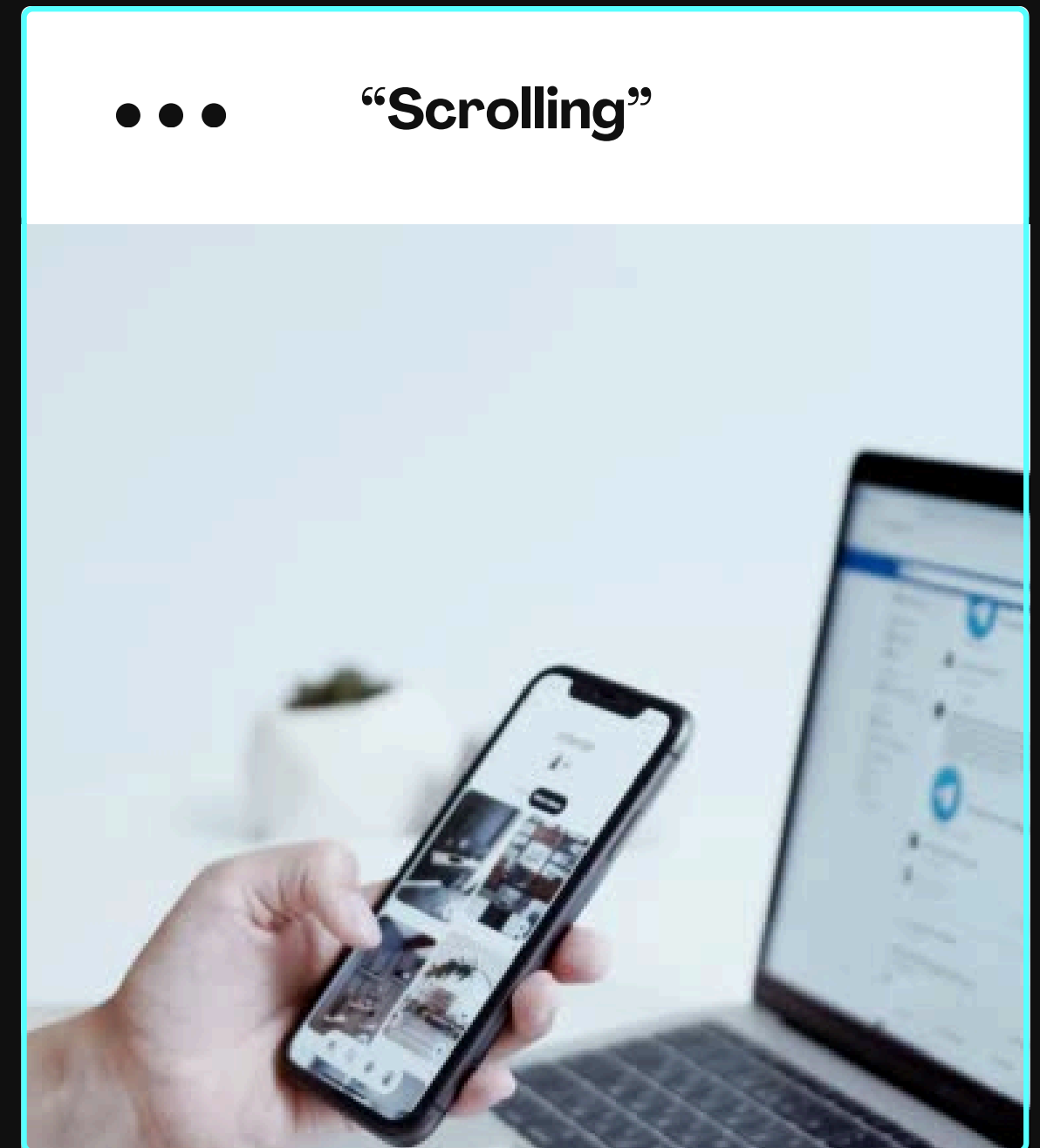
Mécanismes et fonctionnement

**4**

mot de la fin et éventuelles questions

# DISCLAIMER

C'est un sujet qui ne m'était que vaguement connu avant de faire cette présentation. Toute précision, correction ou complément d'information concernant le sujet est la bienvenue.



# QUÉSACO ?

...

## définition, historique et importance

le défilement infini est un concept apparu en 2006. Il permet de nourrir l'utilisateur de nouveau contenu, sans clics ni pages de chargement. Celui-ci apparaît au fil et à mesure que le visiteur "scroll" vers les derniers éléments initialement affichés. Aza Raskin, son inventeur, est connu pour avoir été le designer et responsable user experience de Firefox. il défend une utilisation éthique de la technologie tout critiquant ses effets sur la vie quotidienne et la société. C'est aussi le fils de Jef Raskin, spécialiste des interactions homme-machine et initiateur du projet Macintosh d'Apple.



## • • • mécanismes et fonctionnement

```
1  const container = document.querySelector('.container');
2
3  const URL = 'https://dog.ceo/api/breeds/image/random'
4
5  // get the images
6
7  function loadImages(numImages = 10){
8    let i=0;
9    while(i < numImages){
10     fetch('https://dog.ceo/api/breeds/image/random')
11     .then(response=>response.json())
12     .then(data=>{
13       // console.log(data.message)
14       const img = document.createElement('img');
15       img.src = `${data.message}`
16       container.appendChild(img)
17     })
18     i++;
19   }
20 }
21
22 loadImages();
23
24
25
26 // listen for scroll event and load more images if we reach the bottom of window
27 window.addEventListener('scroll', ()=>{
28   console.log("scrolled", window.scrollY) //scrolled from top
29   console.log(window.innerHeight) //visible part of screen
30   if(window.scrollY + window.innerHeight >= document.documentElement.scrollHeight){
31     loadImages();
32   }
33 })
```

Grâce à l'API, le code va générer 10 photos de chiens, aléatoirement.

si la partie défilée et la partie visible sont supérieures à la hauteur totale du document, alors on va charger d'autres images.

## HOW ?

avec Javascript, en utilisant une fonction, une boucle *while* et un event listener.

- La fonction loadImages va aller chercher l'image et créer un élément img dans le fichier HTML
- la boucle while va générer des images tant que i est inférieur au nombre d'images voulues.
- **window.scrollHeight**: indique la hauteur du document entier
- **window.scrollY**: indique jusqu'où le document a été défilé (depuis le haut)
- **window.innerHeight**: Indique la partie visible du document

# WHAT ELSE?

• • •

## référencement, solutions et bonnes pratiques

le défilement infini, si mal pratiqué, présente des inconvénients :

- temps de chargement
- contenu invisible par le googleBot
- duplication de contenu
- soucis de crawling
- compromet l'analyse des statistiques du site
- ...

Les bonnes pratiques consistent à utiliser un URL unique pour chaque page, et que celles-ci soient accessibles indépendamment des autres. Une autre bonne pratique est d'implémenter en JS un système permettant de manipuler l'historique de navigation afin d'analyser le comportement des utilisateurs. Le but étant que le navigateur comprenne que le scroll fait passer de page en page.

# LE MOT DE LA FIN



# THAT'S IT !

**Merci de votre attention, j'espère que ça vous a plu.**

**ressources :**

<https://www.educative.io/answers/how-to-implement-infinite-scrolling-in-javascript>

<https://www.anthedesign.fr/developpement-web/infinite-scroll/>

<https://www.digitalwellness.fr/scroll-infini>

[https://fr.wikipedia.org/wiki/Aza\\_Raskin](https://fr.wikipedia.org/wiki/Aza_Raskin)