

## PARTIE TP 0 :

### USER INTERFACE :

Terme utilisé principalement dans le domaine des technologies de l'information, l'User Interface (UI) ou Interface Utilisateur désigne l'environnement graphique dans lequel navigue l'utilisateur d'une application, d'un logiciel ou d'un site Web.

### BROWSER ENGINE :

Un moteur de rendu HTML est le composant logiciel de tous les navigateurs permettant d'afficher une page web : il transforme un document HTML, ainsi que toutes les autres ressources associées à la page, en une représentation visuelle interactive pour l'utilisateur. Il est de ce fait le cœur d'un navigateur web.

### RENDERING ENGINE :

Un moteur de rendu est un logiciel qui dessine du texte et des images sur l'écran. Le moteur tire le texte structuré d'un document (souvent HTML) et le met en forme correctement en fonction des déclarations de style données (souvent données en CSS). Exemples de moteurs de mise en page : Blink, Gecko, EdgeHTML, WebKit.

### NETWORKING :

Le réseautage d'affaires (network) désigne une méthode transactionnelle basée sur la théorie des réseaux créant des possibilités d'échanges ou de dons entre des professionnels. Beaucoup d'entrepreneurs ont un même état d'esprit et considèrent que le réseau d'affaires est une méthode plus efficace que les efforts dépensés en publicité ou en relations publiques. En effet, cette activité est peu onéreuse et repose davantage sur un engagement personnel que sur l'argent détenu par une entreprise.

### JAVASCRIPT INTERPRETER :

En informatique, on parle de code interprété ou compilé. JavaScript est un langage interprété : le code est exécuté de haut en bas et le résultat du code exécuté est envoyé immédiatement. Vous n'avez pas à transformer le code en une autre forme avant que le navigateur ne l'exécute. Car cela n'est pas plus nécessaire.

### UI BACKEND :

Le backend, c'est toute la partie que l'utilisateur ne voit pas, mais qui lui permet de réaliser des actions sur un site ou une application. Et une fois que le backend a renvoyé l'information, je peux de nouveau la consulter et interagir avec, je suis de nouveau sur le frontend.

### DATA PERSISTANCE :

La persistance désigne l'ensemble des techniques qui permettent de stocker des données sur votre iPhone. Il existe plusieurs techniques de persistance : Techniquement, la persistance permet d'enregistrer des informations sur un support physique, comme un disque ou un SSD.

## PARTIE TP 1 :

Nous avons commencé par ouvrir git bash, nous avons ensuite rentrer les différentes commandes pour lier notre dossier sur notre ordinateur à notre compte github en ligne

```
$ git clone https://HugosSeulin@github.com/HugosSeulin/Hugo-Seulin.git
```

```
$ cd Hugo-Seulin/
```

```
$ git status
```

```
$ls ( pour trouver le nom de mon dossier )
```

```
MINGW64:/c/Users/hugos/coursprog/Hugo-Seulin

hugos@PC-Hugo MINGW64 ~
$ git status
fatal: not a git repository (or any of the parent directories): .git

hugos@PC-Hugo MINGW64 ~
$ ls
-1.14-windows.xml
'3D Objects'/
AppData/
'Application Data'@
Contacts/
Cookies@
'Creative Cloud Files'/
Documents/
Downloads/
Favorites/
IntelGraphicsProfiles/
Links/
'Local Settings'@
'Menu Démarrer'@
'Mes documents'@
Modèles@
Music/
NTUSER.DAT
NTUSER.DAT{a2332f18-cdbf-11ec-8680-002248483d79}.TM.blf
NTUSER.DAT{a2332f18-cdbf-11ec-8680-002248483d79}.TMContainer00000000000000000000
1.regtrans-ms
NTUSER.DAT{a2332f18-cdbf-11ec-8680-002248483d79}.TMContainer00000000000000000000
2.regtrans-ms
OneDrive/
Pictures/
Recent@
'Saved Games'/
Searches/
SendTo@
Videos/
'Voisinage d'impression'@
'Voisinage réseau'@
anse1/
coursprog/
curseforge/
ntuser.dat.LOG1
ntuser.dat.LOG2
ntuser.ini

hugos@PC-Hugo MINGW64 ~
$ git clone
fatal: You must specify a repository to clone.

usage: git clone [<options>] [--] <repo> [<dir>]

    -v, --verbose          be more verbose
    -q, --quiet            be more quiet
    --progress            force progress reporting
    --reject-shallow      don't clone shallow repository
    -n, --no-checkout      don't create a checkout
```

Le résultat du clonage de notre dossier au dessous :

```
hugos@PC-Hugo MINGW64 ~
$ git clone
fatal: You must specify a repository to clone.

usage: git clone [<options>] [--] <repo> [<dir>]

    -v, --verbose            be more verbose
    -q, --quiet              be more quiet
    --progress               force progress reporting
    --reject-shallow         don't clone shallow repository
    -n, --no-checkout        don't create a checkout
    --bare                   create a bare repository
    --mirror                 create a mirror repository (implies bare)
    -l, --local              to clone from a local repository
    --no-hardlinks           don't use local hardlinks, always copy
    -s, --shared              setup as shared repository
    --recurse-submodules[=<pathspec>]
                             initialize submodules in the clone
    --recursive ...         alias of --recurse-submodules
    -j, --jobs <n>          number of submodules cloned in parallel
    --template <template-directory>
                             directory from which templates will be used
    --reference <repo>      reference repository
    --reference-if-able <repo>
                             reference repository
    --dissociate             use --reference only while cloning
    -o, --origin <name>     use <name> instead of 'origin' to track upstream
    -b, --branch <branch>   checkout <branch> instead of the remote's HEAD
    -u, --upload-pack <path>
                             path to git-upload-pack on the remote
    --depth <depth>         create a shallow clone of that depth
    --shallow-since <time>
                             create a shallow clone since a specific time
    --shallow-exclude <revision>
                             deepen history of shallow clone, excluding rev
    --single-branch          clone only one branch, HEAD or --branch
    --no-tags                don't clone any tags, and make later fetches not to follow them
    --shallow-submodules     any cloned submodules will be shallow
    --separate-git-dir <gitdir>
                             separate git dir from working tree
    -c, --config <key=value>
                             set config inside the new repository
    --server-option <server-specific>
                             option to transmit
    -4, --ipv4               use IPv4 addresses only
    -6, --ipv6               use IPv6 addresses only
    --filter <args>          object filtering
    --also-filter-submodules
                             apply partial clone filters to submodules
    --remote-submodules      any cloned submodules will use their remote-tracking branch
    --sparse                 initialize sparse-checkout file to include only files at root
    --bundle-uri <uri>       a URI for downloading bundles before fetching from origin remote

hugos@PC-Hugo MINGW64 ~
```

```

hugos@PC-Hugo MINGW64 ~
$ git clone https://HugosSeulin@github.com/HugosSeulin/Hugo-Seulin.git
Cloning into 'Hugo-Seulin'...
warning: You appear to have cloned an empty repository.

hugos@PC-Hugo MINGW64 ~
$ cd Hugo-Seulin/

hugos@PC-Hugo MINGW64 ~/Hugo-Seulin (main)
$ git status
On branch main

No commits yet

nothing to commit (create/copy files and use "git add" to track)

hugos@PC-Hugo MINGW64 ~/Hugo-Seulin (main)
$ pwd
/c/Users/hugos/Hugo-Seulin

hugos@PC-Hugo MINGW64 ~/Hugo-Seulin (main)
$ cd ..

hugos@PC-Hugo MINGW64 ~
$ cd c
Contacts/          Cookies/          Creative Cloud Files/  coursprog/        curseforge/

hugos@PC-Hugo MINGW64 ~
$ cd c
Contacts/          Cookies/          Creative Cloud Files/  coursprog/        curseforge/

hugos@PC-Hugo MINGW64 ~
$ cd coursprog/
.git/              Hugo-Seulin/  README.md

hugos@PC-Hugo MINGW64 ~
$ cd coursprog/
.git/              Hugo-Seulin/  README.md

hugos@PC-Hugo MINGW64 ~
$ cd coursprog/Hugo-Seulin/

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ pwd
/c/Users/hugos/coursprog/Hugo-Seulin
*
hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ *

```

Sur la capture au-dessus, nous avons dû résoudre un problème avec un de mes dossiers car il y avait eu un double clonage ce qui posait problème pour lier les deux choses.

Nous savons donc utilisés les commandes :

```

$ pwd
$ cd ..
$ cd c
$ cd c
$ cd coursprog/
$ cd coursprog/Hugo-Seulin/
$ pwd

```

```

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ cd Desktop/
bash: cd: Desktop/: No such file or directory

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ git add .AC
$ git add .

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ git commit -m
error: switch 'm' requires a value

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ git commit -m "ajoute fichier rendu"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'hugos@PC-Hugo.(none)')

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ git config --global user.email "hugo.seulin1@gmail.com"

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ git commit -m "ajoute fichier rendu"
[main (root-commit) 2e72959] ajoute fichier rendu
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Koro Sensei.jpg

hugos@PC-Hugo MINGW64 ~/coursprog/Hugo-Seulin (main)
$ |

```

```

$ cd Desktop/
$ git add .
$ git commit -m "ajoute fichier rendu"
$ git push

```

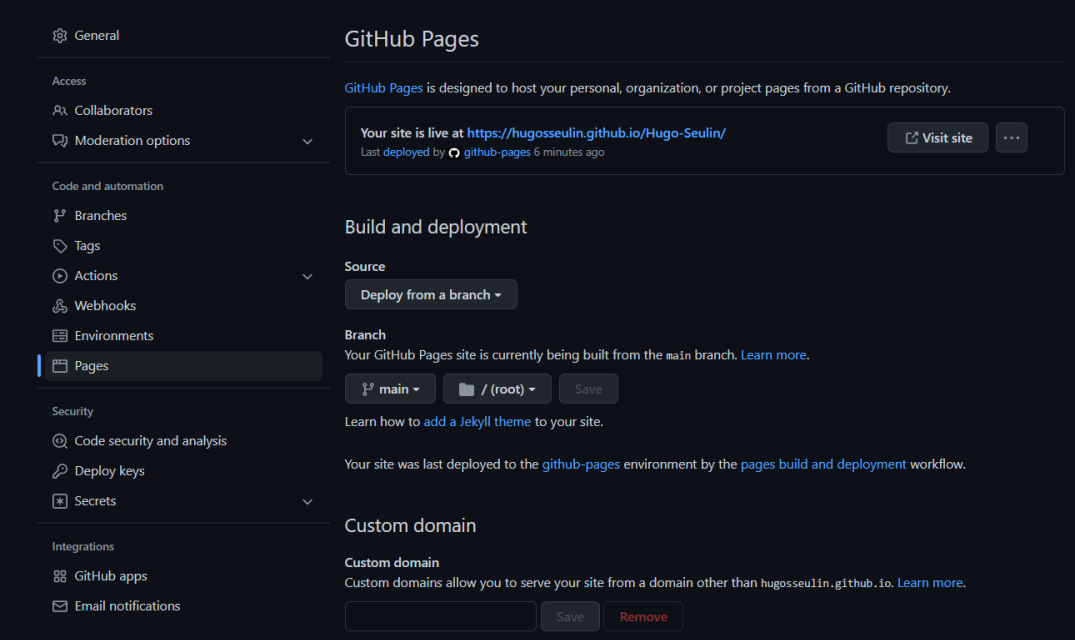
Une fois cela terminé, nos deux dossiers sont liés et les documents déposés sur le dossier pc sont transférés sur github.

## PARTIE TP 2 :

Dans ce tp, nous devons ajouter le squelette d'un site sur notre github afin de le créer et de pouvoir y accéder, afin de rechercher les différents problèmes.

	HugosSeulin ajoute fichier rendu	✓ 2104146 40 minutes ago	🕒 4 commits
📁	img	ajoute fichier rendu	40 minutes ago
📄	Compte rendu M.Linardi.pdf	ajoute fichier rendu	1 hour ago
📄	index.html	ajoute fichier rendu	40 minutes ago
📄	media.css	ajoute fichier rendu	40 minutes ago
📄	standard.css	ajoute fichier rendu	40 minutes ago
📄	w3.css	ajoute fichier rendu	40 minutes ago

Il faut commencer par télécharger le squelette du site sur teams et penser à faire les commandes pour relier les informations des dossiers ( `$git add .` ; `$ git commit -m "ajoute fichier rendu"` ; `$ git push` ), cela fait, il faut se rendre sur notre repository puis dans “settings” et enfin dans l'onglet “Pages”.



**General**

Access

- Collaborators
- Moderation options

Code and automation

- Branches
- Tags
- Actions
- Webhooks
- Environments
- Pages**

Security


- Code security and analysis
- Deploy keys
- Secrets

Integrations

- GitHub apps
- Email notifications

### GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is live at <https://hugosseulin.github.io/Hugo-Seulin/>  
Last deployed by  github-pages 6 minutes ago

[Visit site](#) [...](#)

### Build and deployment

**Source**

Deploy from a branch

**Branch**

Your GitHub Pages site is currently being built from the `main` branch. [Learn more.](#)

`main` `/(root)` [Save](#)

[Learn how to add a Jekyll theme to your site.](#)

Your site was last deployed to the `github-pages` environment by the `pages build and deployment` workflow.

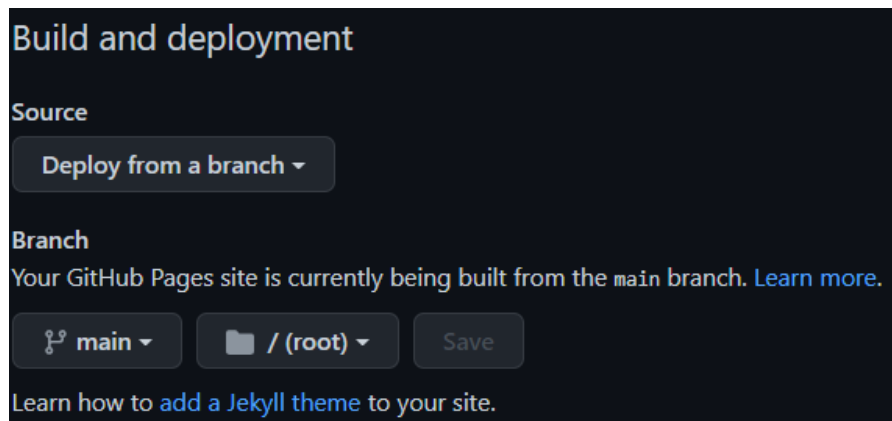
### Custom domain

**Custom domain**

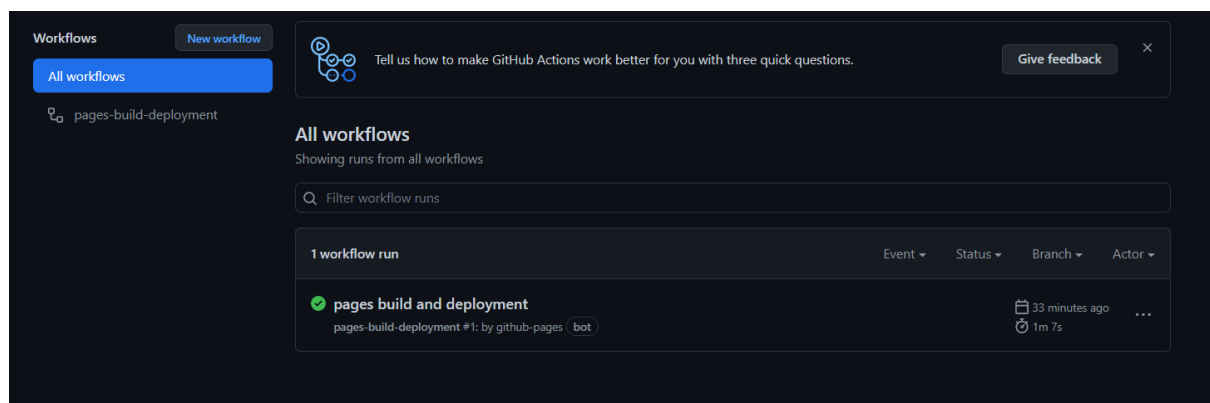
Custom domains allow you to serve your site from a domain other than `hugosseulin.github.io`. [Learn more.](#)

[Save](#) [Remove](#)

Une fois dedans, il faut sélectionner la partie “branch”, “None” et choisir le terme “main” puis faire “save”.

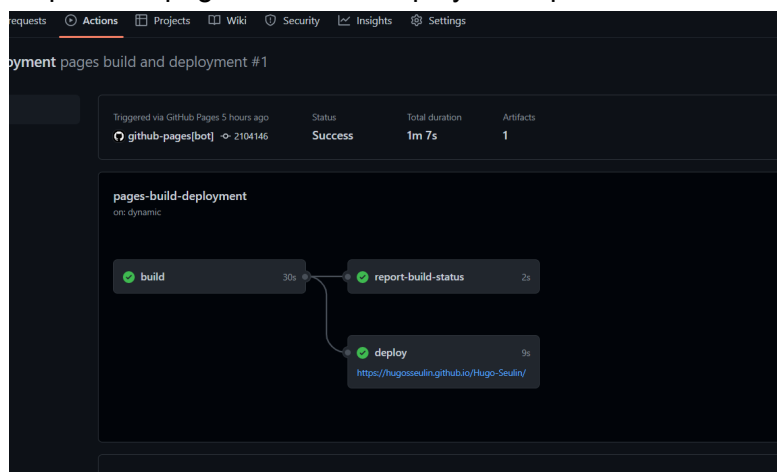


Une fois cela fait, nous devons nous rendre dans l'onglet "action" où le téléchargement de notre site sera affiché "in progress" ou "in queue" puis le temps qu'il aura mis ( 1m7s pour moi ).



Une fois cela fait, 2 solutions s'offrent à nous :

- Cliquer sur "pages build and deployment" puis sur le lien en dessous de "deploy".




- Retourner dans "settings", "pages", une fois sur cette page, le lien y sera affiché avec en plus une icône si nous préférons ouvrir la page dans un nouvel onglet.

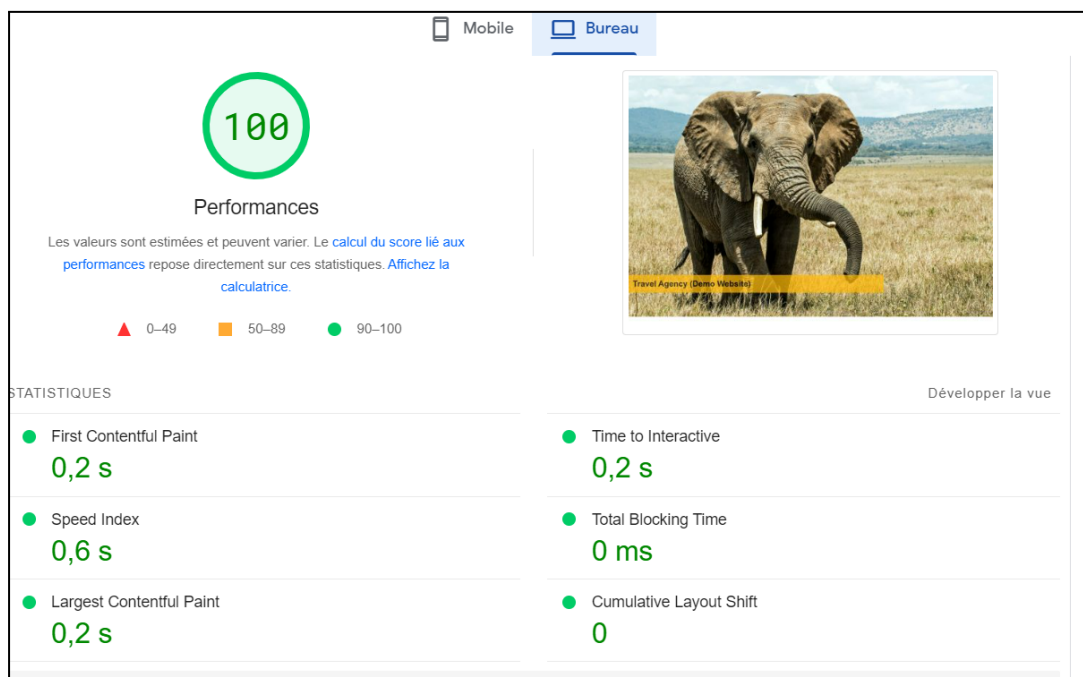
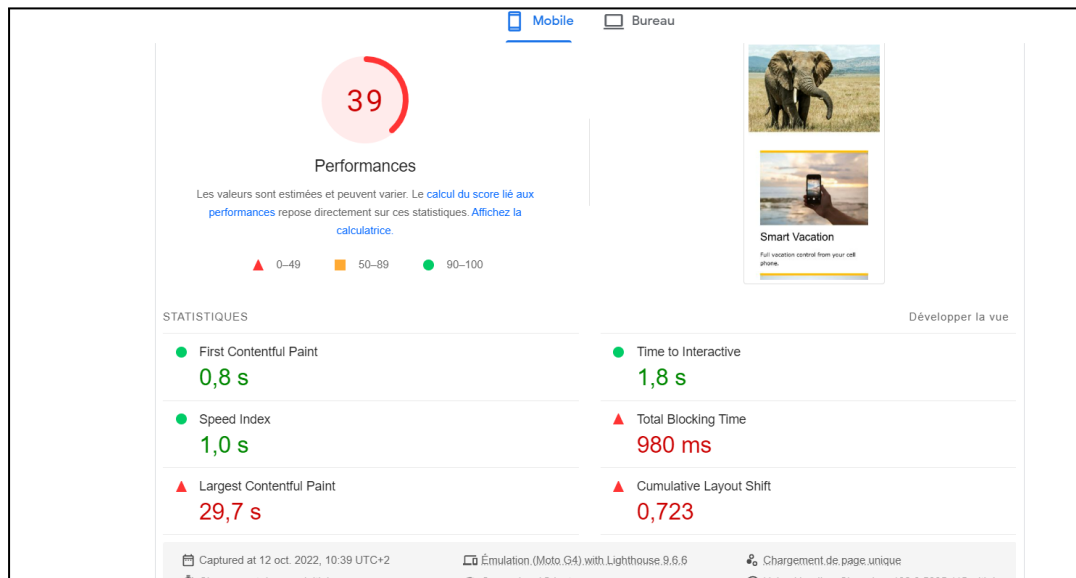
# GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is live at <https://hugosseulin.github.io/Hugo-Seulin/>

Last deployed by  github-pages 6 hours ago

 Visit site



Nous pouvons apercevoir que sur mobile, il est assez long de charger cette page car les images ne sont pas au bon format / trop grandes, la note est donc assez basse ( 39/100 ). Contrairement aux mobiles, les ordinateurs ont assez peu de difficultés à charger ce qui se voit grâce à la note ( 100/100 ).

La raison de ces mauvaises notes est principalement dûe à une seule image qui a un format bien trop grand ([https://hugosseulin.github.io/Hugo-Seulin/img/main\\_HOME.jpg](https://hugosseulin.github.io/Hugo-Seulin/img/main_HOME.jpg)). Il faudrait donc essayer de réduire la taille de l'image grâce à un compresseur en ligne par exemple.



Nous ouvrons la home page du site, vous notez quelque chose d'insolite ?

Ce que j'ai pu constater est que le temps de chargement de la page est relativement long et l'image de l'éléphant met encore plus de temps à charger. Cela est sûrement bien plus long pour mobile selon "PageSpeed Insights" comparé aux ordinateurs qui est un peu plus rapide.

**Comment on peut réduire les requêtes HTTP ?** Afin de réduire le nombre de requêtes HTTP, il y a différentes étapes possible qui sont les suivantes :

- Combiner les fichiers CSS
- Déterminer quelles autres parties de votre page allongent le temps de chargement (l'image de l'éléphant par exemple dans ce cas )
- Réduire la taille des fichiers des images restantes ( image de l'éléphant, image de lion de mer, etc
- Supprimer les images inutiles
- Vérifier le nombre de requêtes HTTP généré par votre site web grâce au panneau de Google Chrome

**Est-ce que ton code est «clean»? What ?**

Il y a peut-être un peu trop de classes CSS qui sont écrites de manière trop compliquées ainsi que la rédaction du code qui est trop aérée. Il faudrait donc enlever certains espaces, certains saut de lignes, etc