

AUDIT DE PERFORMANCE

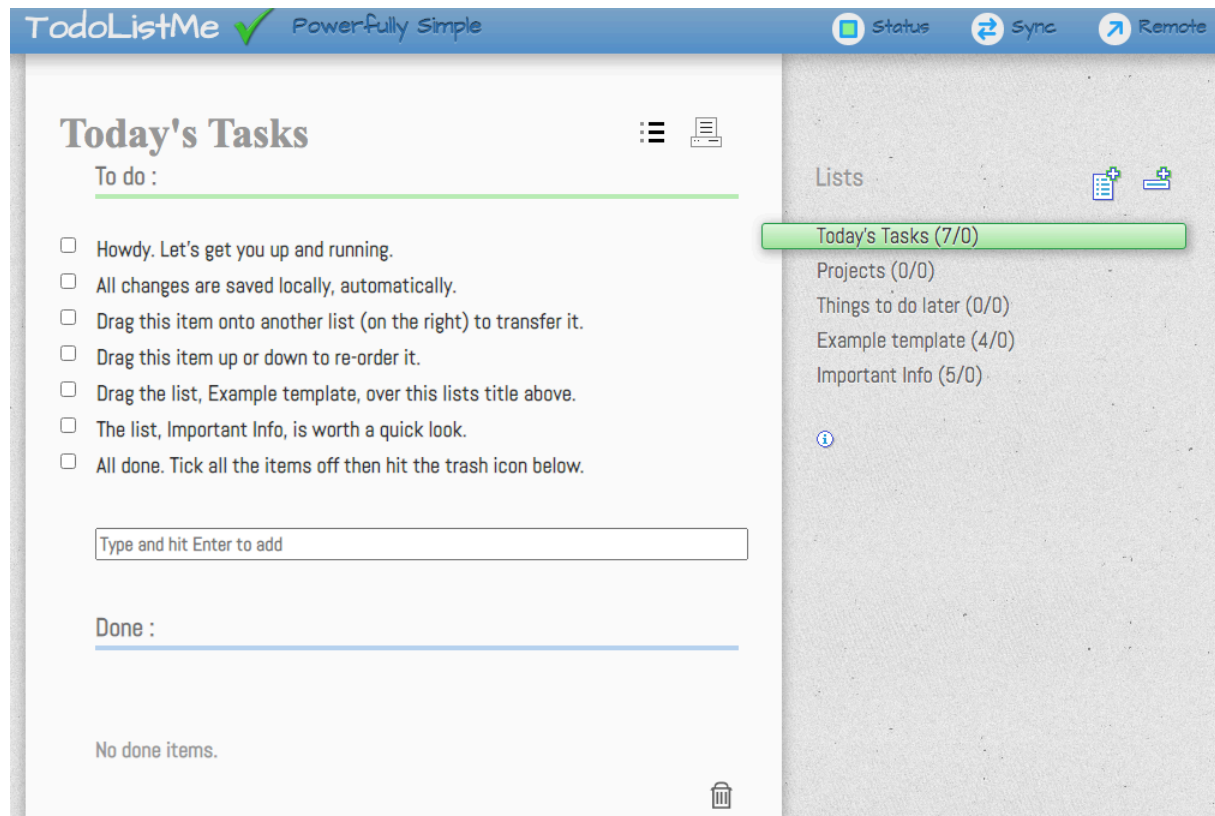
<http://todolistme.net/>

1. INTRODUCTION

Afin d'identifier des pistes d'amélioration de notre application "*TO-DO List*" nous allons analyser une application concurrentielle : <http://todolistme.net/>

2. PRÉSENTATION DE L'APPLICATION TODOLISTME.NET

2.1 Interface utilisateur



L'interface est beaucoup plus chargée que notre application "*TO-DO List*". L'utilisateur peut vite être perdu dans les différentes fonctionnalités de l'application, celle-ci semble être moins intuitive que notre application.

2.1 Fonctionnalités

2.1.1 Ajout d'une tâche

Identique à notre application, "*todolistme*" propose l'ajout de tâches et affiche ces dernières sous forme de liste.

2.1.2 Tri des tâches

Il est possible de trier les tâches via un système de drag & drop.

Le drag & drop permet de :

- Déplacer les tâches dans la liste
- Déplacer les tâches dans une catégorie de tâches
- Déplacer les tâches à réaliser à une date ultérieure de celle du jour de création

Il est également possible de trier les tâches via un clic sur le pictogramme de tri. Cette fonction permet de trier les tâches par ordre alphabétique, par le Top 3 des tâches, ou encore au hasard.

2.1.3 Ajout d'une catégorie de tâches

L'application "*todolistme*" propose la fonctionnalité de créer des catégories de tâches via un simple clic sur le pictogramme d'ajout de catégorie.

2.1.4 Ajout d'une liste de tâches

L'application "*todolistme*" propose la fonctionnalité de créer des listes de tâches via un simple clic sur le pictogramme d'ajout de liste.

2.1.5 Imprimer une liste de tâches

En cliquant sur le pictogramme de l'imprimante une nouvelle page s'ouvre et permet l'impression de la liste.

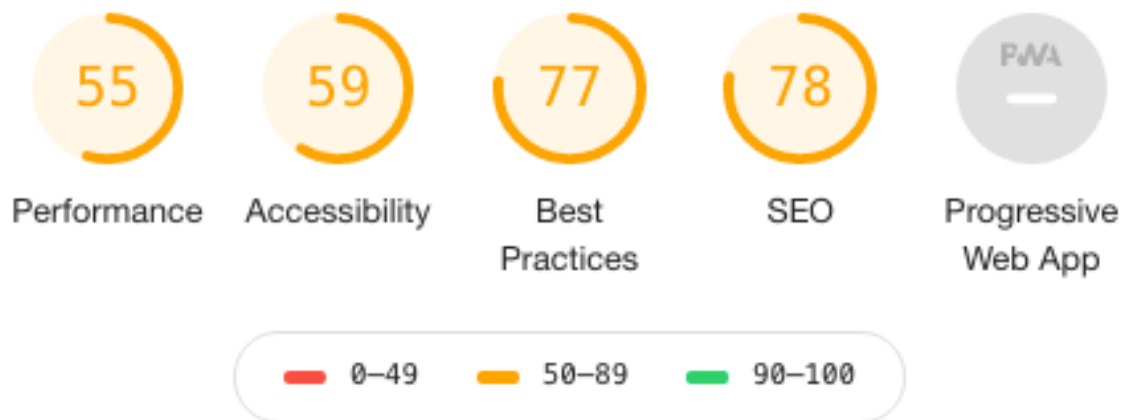
2.1.6 Création de compte + synchronisation

"*todolistme*" propose la création d'un compte utilisateur afin de pouvoir accéder à sa liste de tâches sur d'autres supports en se connectant à l'application.

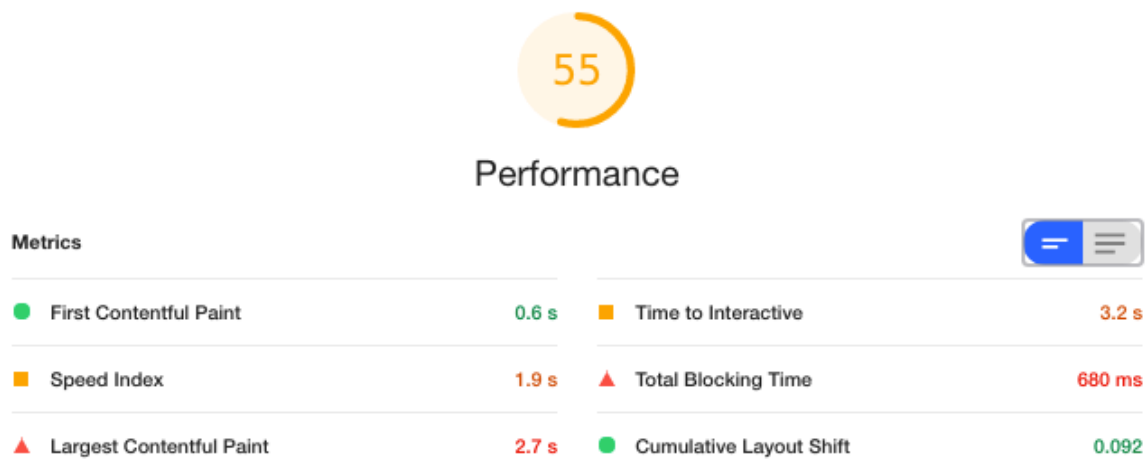
3. PERFORMANCES

Afin d'analyser les performances de l'application "*todolistme*" nous utilisons l'outil *Lighthouse* proposée par le navigateur Google Chrome.

3.1 Analyse du rapport de performance avec Lighthouse



3.1.1 Performance



- *First Contentful Paint* mesure le temps qu'il faut au navigateur pour restituer le premier élément de contenu du DOM au moment où un utilisateur accède à l'application.

Pour notre application concurrente le temps est de 0,6s, soit un score tout à fait raisonnable.

- *Time to Interactive* est une donnée importante car certains sites optimisent la visibilité du contenu au détriment de l'interactivité. Cela peut créer une

expérience utilisateur frustrante car le site semble prêt, mais lorsque l'utilisateur essaie d'interagir avec lui, rien ne se passe.

Pour notre application concurrente le temps est de 3,2s, soit un score assez faible.

- *Total Blocking Time* mesure la durée totale pendant laquelle une page est empêchée de répondre aux entrées de l'utilisateur, telles que les clics de souris, les pressions sur l'écran ou les pressions sur le clavier. La somme est calculée en ajoutant la partie bloquante de toutes les tâches longues entre *First Contentful Paint* et *Time to Interactive*. Toute tâche qui s'exécute pendant plus de 50 ms est une tâche longue. La durée après 50 ms est la partie bloquante. Par exemple, si Lighthouse détecte une tâche de 70 ms, la partie bloquante sera de 20 ms.

Pour notre application concurrente le temps est de 680ms, soit un score très faible.

- *Speed index* mesure la vitesse à laquelle le contenu est affiché visuellement pendant le chargement de la page. Lighthouse capture d'abord une vidéo du chargement de la page dans le navigateur et calcule la progression visuelle entre les images. Lighthouse utilise ensuite le module *Speedline Node.js* pour générer le score *Speed Index*.

Pour notre application concurrente le temps est de 1,9s, soit un score faible.

- *Largest Contentful Paint* (LCP) mesure le moment où le plus grand élément de contenu de la fenêtre est rendu à l'écran. Cela correspond approximativement au moment où le contenu principal de la page est visible pour les utilisateurs.

Pour notre application concurrente le temps est de 2,7s, soit un score très faible.

- *Cumulative Layout Shift* mesure la somme totale de tous les scores de changement de mise en page individuels pour chaque changement de mise en page inattendu qui se produit pendant toute la durée de vie de la page.

Pour notre application concurrente la note est de 0,092, soit un très bon score.

3.1.2 Accessibilité



Accessibility

L'analyse de l'accessibilité de l'application "*todolistme*" révèle les points suivants :

- **Contraste** : Les couleurs d'arrière-plan et de premier plan n'ont pas un rapport de contraste suffisant.
- Les éléments `<frame>` ou `<iframe>` n'ont pas de "title"
- Les images n'ont pas d'attribut alt
- Les éléments du formulaire n'ont pas de label
- Les balises de titre h1, h2, h3...etc... ne sont pas dans ordre logique
- La balise `<html>` ne comprends pas d'attribut de langue

3.1.3 Meilleures pratiques



Best Practices

- Le site n'utilise pas le protocole https
- La librairie jQuery n'est pas à jour de la dernière version

3.1.4 SEO



- L'application ne contient pas de balise `<meta name = "viewport">` avec une "width" ou une "initial-scale"
- Les images n'ont pas d'attribut alt