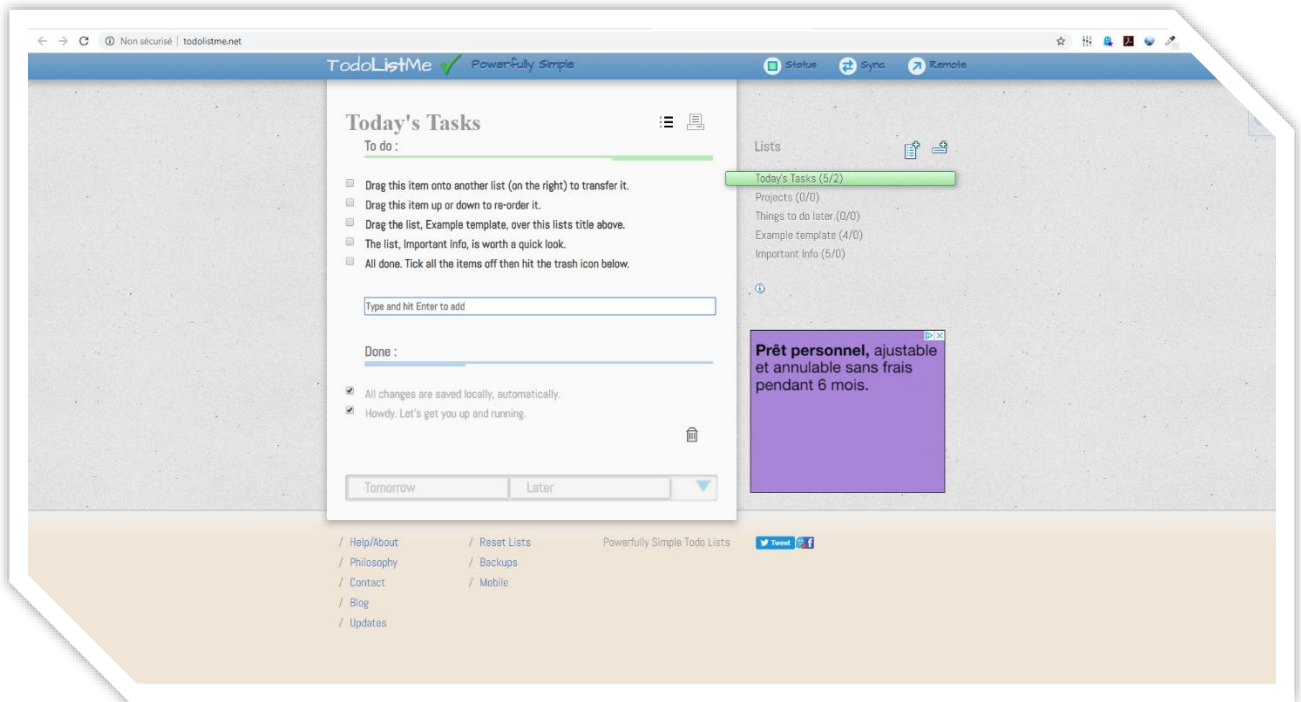


# AUDIT PROJET 8

Grégory Huyghe

## Application concurrente

L'application concurrente qui va être analysée dans ce document est TodoListMe (<http://todolistme.net/>). C'est aussi une application qui permet de lister des tâches à effectuer, de les ranger par catégories ou date, par complétion, de les supprimer, ainsi que de créer de nouvelles catégories de tri.



























Ses possibilités sont donc plus nombreuses mais l'audit sur ses performances va nous révéler qu'il y a de nombreux écueils que l'on pourra éviter si nous souhaitons développer notre outil.

L'objectif de cet audit est d'avoir une vision d'ensemble des performances de l'application concurrente par rapport à la nôtre, afin d'avoir toutes les informations nécessaires en vue d'un éventuel scaling de notre application.

## Performance

La performance de l'application concurrente va être analysée aux outils de la console du navigateur Chrome.

Le chargement complet de l'application prend 4,95s, pour un total de 943kb et 72 fichiers transférés.






























Name	Size	Time	▲	Waterfall
 data:application/ja...		184 B	4 ms	
 ga.js		17.0 KB	63 ms	
 osd.js?cb=%2Fr20100101		27.9 KB	90 ms	
 ca-pub-9971420945046594.js		403 B	133 ms	
 integrator.js?domain=todolistme.net		456 B	133 ms	
 plusone.js		17.1 KB	137 ms	
 integrator.js?domain=todolistme.net		225 B	148 ms	
 cb=gapi.loaded_0		18.7 KB	163 ms	
 adsbygoogle.js		34.0 KB	170 ms	
 jquery-2.2.4.min.js		29.6 KB	256 ms	
 1441128190-postmessagerelay.js		4.2 KB	276 ms	
 ca-pub-9971420945046594.js		146 B	290 ms	
 show_ads_impl.js		78.6 KB	291 ms	
 cb=gapi.loaded_0		48.8 KB	292 ms	
 rpcshindig_random.js?onload=init		5.0 KB	295 ms	
 show_ads_impl.js		78.6 KB	362 ms	
 show_ads.js		25.3 KB	391 ms	
 lib.js		1.7 KB	522 ms	
 lists.js		9.9 KB	524 ms	
 button.509719336ca39171c37a321231ccaf83.js		7.2 KB	533 ms	
 widgets.js		93.4 KB	562 ms	
 cb=gapi.loaded_1		33.5 KB	568 ms	
 javascript_e.js		10.1 KB	652 ms	
 jquery-ui.js		122 KB	661 ms	
24 / 72 requests   664 KB / 943 KB transferred   1.8 MB / 2.1 MB resources   Finish: 5.49 s   DOMContentLoaded: 1.99 s   Load: 4.95 s				

Au niveau des scripts, les fichiers nécessaires au bon fonctionnement de l'application sont lib.js, lists.js et javascript\_e.js. Leur poids est léger et pour les deux premiers le chargement rapide.

Au niveau des publicités, il y a deux fichiers show\_ads de 78.6 et 25.3kb chacun.









Ensuite, il y a widgets.js dont l'URL est : <http://platform.twitter.com/widgets.js>, donc le bouton Twitter pour un total de 93,4kb.

Enfin, jquery-ui.js qui charge en 661ms pour un total de 122kb.

Name	Size	Time	Waterfall
	(memory cache)	0 ms	
	(memory cache)	0 ms	
	177 B	48 ms	
	582 B	157 ms	
	529 B	170 ms	
	1.7 KB	174 ms	
	945 B	192 ms	
	433 B	195 ms	
	973 B	204 ms	
	812 B	208 ms	
	542 B	209 ms	
	512 B	216 ms	
	417 B	218 ms	
	7.1 KB	229 ms	
	1.1 KB	246 ms	
	1.1 KB	249 ms	
	1.3 KB	327 ms	
	725 B	337 ms	
	922 B	337 ms	
	510 B	395 ms	
	334 B	400 ms	
	792 B	400 ms	
	913 B	402 ms	
	1.1 KB	408 ms	
	3.3 KB	496 ms	
	6.0 KB	529 ms	
	814 B	590 ms	
	759 B	603 ms	
	129 KB	1.75 s	
29 / 72 requests   163 KB / 943 KB transferred   173 KB / 2.1 MB resources   Finish: 5.49 s   DOMContentLoaded: 1.99 s   Load: 4.95 s			

Les images impactent grandement le temps de chargement global de l'application.

Texture.png est l'image utilisée en background et pèse 129kb pour 1.75s de chargement.

Name	Size	Time	Waterfall
	7.2 KB	104 ms	
	710 B	157 ms	
	1.0 KB	228 ms	
	177 B	238 ms	
	21 B	242 ms	
	32.7 KB	299 ms	
	4.6 KB	710 ms	
	15.3 KB	820 ms	

En triant par Doc dans l'onglet Network de la console Chrome, on aperçoit des temps de chargement dédiés aux publicités (Google Ads) et à Twitter (tweet\_button et widget\_iframe).

Name	Size	Time	W
rpc:shindig_random.js?onload=init	5.0 KB	295 ms	
tweet_button.d753e00c3e838c1b2558149bd3f6e...	32.7 KB	299 ms	
style_g.css	7.2 KB	301 ms	
top_sync_waiting.png	1.3 KB	327 ms	
category_up.png	725 B	337 ms	
arrow_up.png	922 B	337 ms	
show_ads_impl.js	78.6 KB	362 ms	
show_ads.js	25.3 KB	391 ms	
copy.png	510 B	395 ms	
delete.gif	334 B	400 ms	
top_saved.png	792 B	400 ms	
top_new_window.png	913 B	402 ms	
top_sync.png	1.1 KB	408 ms	
googlelogo_color_150x54dp.png	3.3 KB	496 ms	
favicon.ico	17.4 KB	512 ms	
lib.js	1.7 KB	522 ms	
lists.js	9.9 KB	524 ms	
tick.png	6.0 KB	529 ms	
button.509719336ca39171c37a321231ccaf83.js	7.2 KB	533 ms	
widgets.js	93.4 KB	562 ms	
cb=gapi.loaded_1	33.5 KB	568 ms	
top_not_saved.png	814 B	590 ms	
category_down.png	759 B	603 ms	
javascript_e.js	10.1 KB	652 ms	
jquery-ui.js	122 KB	661 ms	
todolistme.net	4.6 KB	710 ms	
settings	382 B	712 ms	
widget_iframe.d753e00c3e838c1b2558149bd3f6...	15.3 KB	820 ms	
texture.png	129 KB	1.75 s	
72 requests   943 KB transferred   2.1 MB resources   Finish: 5.49 s   DOMContentLoaded: 1.99 s   Load: 4.95 s			

En analysant le chargement au niveau global sans trier par type de fichiers, on aperçoit que parmi les éléments mettant plus de 600ms à charger, il y a deux images (category\_down et texture), deux scripts (javascript\_e et jquery-ui), la partie html du site (todolistme), une requête http pour Twitter (settings), et le widget de twitter (widget\_iframe).

- Bilan

Ces tests ont été effectués avec le cache désactivé. L'activation de celui-ci permet de diviser le temps de chargement par 2. Les points de vigilance semblent donc être l'implémentation des réseaux sociaux (Twitter), l'utilisation d'une image en background, l'utilisation de la bibliothèque jQuery (abandonnée dans Bootstrap 5), ainsi que l'ajout de publicités.

## Propositions en vue d'un scaling

Si nous souhaitons améliorer notre propre application, certains écueils peuvent être évités après analyse de l'application concurrente.

- Un background simple, une couleur. L'utilisation d'une image alourdit considérablement le chargement du site. Même si la mise en cache supprime ce problème quand l'utilisateur revient, la première fois est importante en termes d'image pour l'entreprise.
- Ne pas monétiser le site web ou de manière plus légère. Offrir des fonctionnalités premium contre abonnement peut permettre de générer des revenus sans alourdir le fonctionnement de l'application, surtout qu'il n'y a pas de mise en cache pour cet élément.
- Implémenter les réseaux sociaux mais laisser l'utilisateur libre de les activer ou non. Cela permet de respecter la RGPD et allégera le chargement de l'application. La mise en cache rendra la navigation plus facile pour les utilisateurs réguliers.
- Ne pas utiliser jQuery mais plutôt un framework tel que Vue, ou si les fonctionnalités sont simples, directement vanilla JS. Ceci évitera le chargement d'une librairie. Les navigateurs ont maintenant une compatibilité plus performante et couvrir IE n'est pas nécessairement utile.