

Choisissez une librairie adaptée aux besoins de votre projet

8–10 minutes

Votre page web est désormais totalement opérationnelle :

- vous pouvez interagir avec les fiches produits à l'aide du menu de filtres ;
- les données affichées sont récupérées dynamiquement grâce à une API en ligne.

Vous décidez maintenant de concentrer vos efforts sur l'analyse et l'affichage des avis utilisateurs. Actuellement, ces derniers déposent des avis avec un système de notation par étoiles, de 1 à 5. Vous souhaitez faciliter la lecture de ces avis en affichant une synthèse de cette évaluation, inspirée du site web d'Amazon :



Pour développer cette fonctionnalité, nous allons utiliser des **librairies**. Alors, c'est parti ! Voyons ensemble ce qu'est une librairie, dans quels cas vous en avez besoin, et comment bien les

choisir.

Découvrez les librairies

La plus connue d'entre elles est sans doute **jQuery**. Cette librairie offre une solution clé en main pour écrire le code de manipulation du DOM de manière plus concise.

Ainsi, pour récupérer un élément du DOM et changer son contenu textuel en version standard, on écrirait :

```
document.querySelector("button").innerText =  
"Rejoignez l'entreprise Les Bonnes Pièces !";
```

Alors qu'avec jQuery cela donnerait :

```
$("button").text("Rejoignez l'entreprise Les Bonnes  
Pièces !");
```

La librairie ajoute la variable **\$** comme alias de la fonction *document.querySelector*. On peut changer le texte de l'élément sélectionné avec la fonction **text** plutôt qu'avec la propriété *innerText*.

Le code est beaucoup plus concis et se lit plus vite, mais il faut connaître le fonctionnement de jQuery pour en profiter. De vous à moi, cela peut être un investissement rentable si vous travaillez beaucoup avec JavaScript et le DOM. 😊

Arbitrez entre développement interne et utilisation d'une librairie

Quand vous développez une fonctionnalité sur une application, vous avez donc deux possibilités : la développer vous-même, ou utiliser une librairie. Explorons ensemble les critères de décision qui vous permettront de choisir entre les deux.

Dans quels cas utiliser une librairie ?

Quand vous disposez d'un temps limité pour développer votre application

Dans cette situation, vous devez aller vite. Une librairie vous offrira une solution clé en main, et vous permettra de gagner du temps. Petit bémol cependant : vous devrez personnaliser la librairie pour obtenir un résultat optimal.

Quand vous ne pouvez pas faire mieux vous-même

Inutile de réinventer la roue ! Certaines librairies contiennent des tests automatisés qui garantissent la justesse, la robustesse et la stabilité du code. Il serait dommage de ne pas utiliser ces librairies de haute qualité dans un projet.

Quand le code à développer nécessite une expertise poussée

C'est le cas, par exemple, pour les librairies de modélisation et de rendu en 3D pour créer des images de synthèse. Ce genre de code fait appel à des connaissances très pointues en mathématiques. Autre domaine concerné : la cryptographie et le chiffrement. Dans ce cas, développer son propre code de chiffrement est même une faute professionnelle. Il faut toujours utiliser une librairie.

Dans quels cas développer votre propre code ?

Quand la licence logicielle de la librairie est trop restrictive

Il arrive aussi que l'auteur d'une librairie écrive son propre contrat : en interdisant l'utilisation de sa librairie pour des utilisations militaires, par exemple. Si votre projet concerne cette industrie, vous devrez trouver une alternative ou développer le code vous-même.

Quand vous avez absolument besoin d'un support technique

Certaines librairies peuvent répondre à votre besoin, mais ne proposent pas de support technique. C'est souvent le cas avec les projets open source maintenus par des bénévoles sur leur temps

libre. Ils ne pourront pas vous répondre en priorité, ni apporter des changements ou des corrections selon vos besoins. Si la librairie vous intéresse vraiment, vous pouvez toujours créer **un fork** et apporter vos propres modifications (à condition que la licence logicielle vous le permette).

Euh... c'est quoi un fork ?

Pour faire simple, on peut dire que c'est une copie d'un dépôt GitHub/GitLab. Je vous invite à jeter un œil à [ce chapitre](#) du cours *Devenez un expert de Git et GitHub* pour en savoir plus. 😊

Quand votre besoin est trop spécifique ou innovant

Il se peut que personne n'ait eu le même besoin que vous. Dans ce cas, il vous faudra développer vous-même la solution. Vous pourrez alors publier votre code en ligne et créer votre propre librairie. Cependant, cela reste un cas de figure assez rare.

Faites le bon choix pour le site des Bonnes Pièces

L'heure est venue de prendre une décision par rapport à notre besoin ! Dans notre cas :

- nous ne sommes pas spécialement pressés par le temps ;
- il existe de très bonnes librairies de graphiques ;
- l'affichage de graphique requiert rigueur et formalisme ;
- nous supposons que les licences ne seront pas restrictives : à confirmer lors de notre recherche ;
- notre besoin n'est pas spécifique : l'affichage de graphique est une fonctionnalité courante ;
- nous n'avons pas besoin de support technique.

Pour notre site web des Bonnes Pièces, nous choisissons donc d'utiliser une librairie. Voyons ensemble comment trouver une librairie adaptée. 😊

Trouvez des librairies sur internet

Utilisez des sites de référence

Dans le domaine du développement web en JavaScript, le [site web NPM](#) est LA référence. Il dispose d'une barre de recherche et d'informations clés sur chaque librairie, pour vous permettre de faire le bon choix.

Le site <https://github.com> vous permet également de trouver des librairies. Cependant, restez vigilant dans votre recherche : le site propose des librairies écrites dans d'autres langages informatiques que le JavaScript, qui ne pourront pas s'exécuter dans votre navigateur.

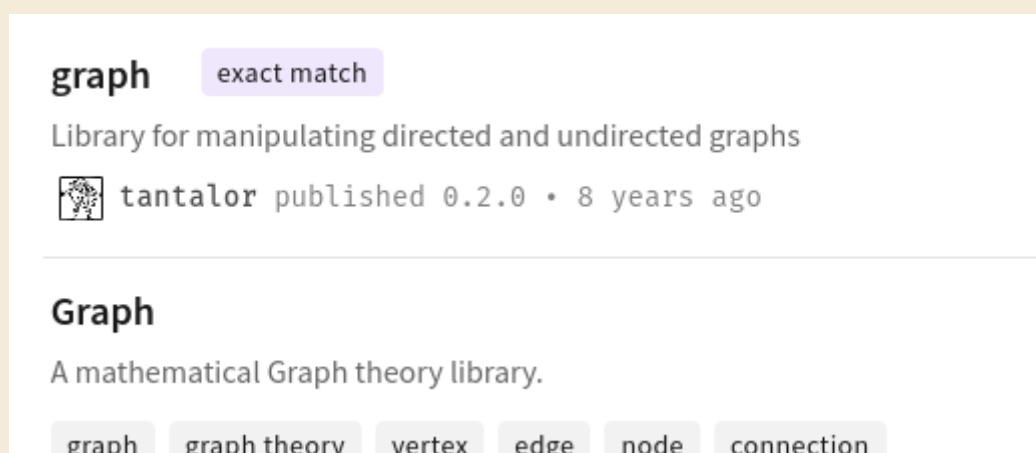
Enfin, sachez que vous pouvez aussi faire une recherche sur votre moteur de recherche favori. La petite astuce que je vous conseille est de faire précéder votre recherche par le mot clé "awesome" (génial, en anglais). Par exemple "awesome javascript graph". Vous tomberez sur des collections ressources géniales rassemblées par d'autres développeurs passionnés. 😎

Choisissez votre librairie de graphiques

Rendez-vous sur le site de NPM et faites la recherche "graph" (graphique, en anglais).

Nous pouvons déjà écarter les deux premiers résultats :

- Le premier n'a pas été mis à jour depuis 8 ans.
- Le deuxième concerne la théorie des graphes en mathématiques.





yoni published 0.0.1 • Invalid date

Les deux premiers résultats de votre recherche sur npmjs.com ne correspondent pas à notre besoin

On trouve ensuite des résultats concernant GraphQL, un langage de requêtage de bases de données. Cela ne nous concerne pas...

Continuons tout de même notre recherche. Plus loin dans la liste, on trouve un résultat avec une description intéressante : "Simple HTML5 charts using the canvas element". En français : Graphiques simples en HTML 5 avec l'élément canvas.

chart.js

Simple HTML5 charts using the canvas element.

canvas

charts

data

graphs

html5

responsive



chartjs-ci published 3.7.1 • 3 months ago

Un résultat de recherche semble correspondre à notre besoin

La dernière publication date d'il y a trois mois, et on y parle de l'élément canvas et d'HTML. C'est plutôt positif ! En cliquant sur ce résultat de recherche pour afficher la page de description de cette librairie, on peut voir des informations intéressantes sur la droite :

Weekly Downloads

2,077,141

Version

3.7.1

License

MIT

Statistiques de la librairie Chart.js

Plus de deux millions de téléchargements hebdomadaires, cette librairie est donc populaire. Elle est à sa version 3 avec 7 mises à jour, elle semble donc mature. La licence MIT est très permissive : elle offre la possibilité de réutiliser très facilement le code en cas de modification, ou lorsque l'on forke le projet. Bingo ! Nous avons trouvé notre librairie. 🥳

En résumé

- En tant que développeur, vous pouvez utiliser des librairies pour gagner du temps dans le développement de votre projet.
- Dans certains cas cependant, il peut être préférable de recourir à un développement en interne :
 - quand le besoin est trop spécifique ;
 - quand la licence logicielle est trop restrictive.
- Vous pouvez trouver des librairies par le biais de sites de référencement tels que :
 - npmjs.com ;
 - GitHub ;
 - Un moteur de recherche avec la recherche “awesome + votre technologie”.

Nous avons fait un choix important pour notre site web en optant pour Chart.js. Retrouvez-moi dans le prochain chapitre pour installer la librairie sur votre ordinateur.

- <#>