



# Présentation de l'outil de veille

---

CRÉATION DU SITE WEB « MENU MAKER » BY QWENTA

# Pourquoi faire une veille technologique ?

Dans la gestion d'un projet de création de site web, la veille est un outil dont un développeur ne peut se passer.

Cette veille permet de se tenir informé des tendances actuelles et des avancées technologiques, de voir ce que fait la concurrence. Le but de cette veille est de trouver ce qui peut nous permettre d'améliorer la qualité et la quantité du travail fourni. Parfois elle permet d'anticiper les évolutions futures.

La veille est un processus consistant à surveiller en permanence tout ce qui est étroitement lié avec le projet en cours, sur Internet. Bien sûr, on ne peut pas passer tout son temps à chercher des informations. Pour cela, il existe des outils qui permettent d'automatiser la recherche et de trier les résultats en fonction des critères que l'on choisit.

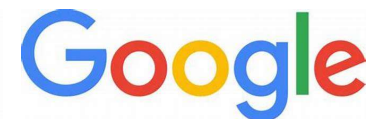


Les outils pour mettre en place une veille informatique sont nombreux. Parmi les plus connus, on peut citer Feedly, daily.dev, Pearltrees, Wakelet, X (anciennement Twitter), LinkedIn, Netvibes, Google Alerts, etc...

Tous ces outils permettent de s'abonner à des newsletters, des flux RSS et des blogs, de suivre des comptes sur les réseaux sociaux, de recevoir des alertes quand un nouveau contenu est créé.

Pour ce projet, j'ai utilisé plusieurs plateformes, pour me tenir informé des dernières évolutions dans le domaine du développement web: Netvibes, Daily.dev, Feedly, et LinkedIn.

Il ne faut pas non plus oublier les moteurs de recherche, qui sont maintenant aidés par l'intelligence artificielle, comme Google et Bing.



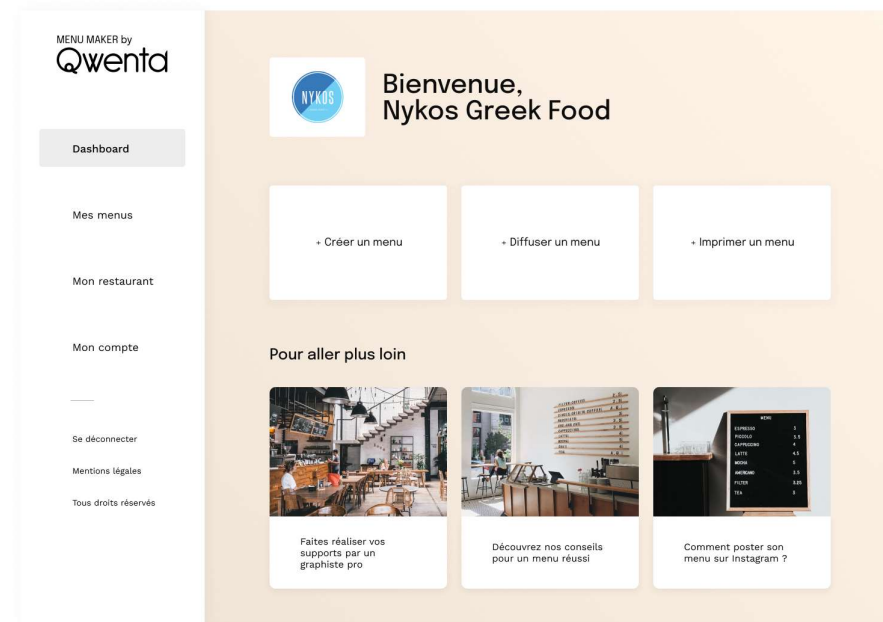


# Comment mettre la veille en place ?

J'ai commencé par bien étudier le projet MENU MAKER. Menu Maker by Qwenta est une interface sur laquelle les restaurateurs peuvent se connecter pour créer, ou personnaliser leurs menus, les diffuser en ligne ou les imprimer. Le menu pourra être exporté en PDF, diffusé sur Deliveroo et partagé sur Instagram.

Le site devra être en version desktop, dynamique et compatible avec les dernières versions de Chrome, Safari et Firefox. Il faudra déterminer la programmation côté serveur. J'ai détaillé la maquette Figma, ainsi que les spécifications fonctionnelles qui sont fournies, pour établir une liste de tout ce qui semblait nécessaire : Choix d'un langage, Framework, Librairies, Base de données, API, Hébergeur, Outils de sécurisation.

Pour commencer la veille, j'ai rassemblé un maximum d'informations: des sites Internet, des blogs tenus par des développeurs web, des articles de médias spécialisés. J'ai aussi lancé des alertes sur des mots-clés en rapport avec le projet : Javascript, front-end, back-end, base de données, API, Deliveroo, Instagram, PDF, etc...





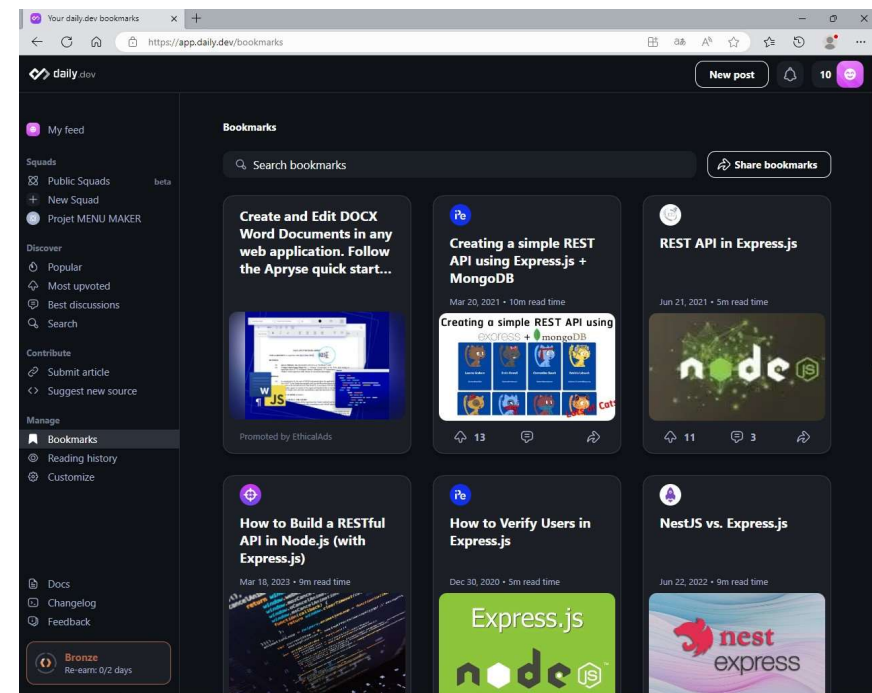
## Préciser la méthode de la veille.

Parmi tous les articles consultés, j'ai dû appliquer des critères de sélection :

- Lien avec le projet, pour ne pas s'écarter trop du sujet.
- Date de parution récente, pour rester dans les tendances actuelles.
- Lisibilité et temps nécessaire pour la lecture (pour éviter de perdre tout son temps).
- Consultation des avis, et du nombre d'abonnés, pour vérifier le sérieux de l'article.

Le fonctionnement des outils de veille est simple. On peut consulter les articles en rapport avec les mots-clés choisis, puis il faut cliquer sur « Ajouter » ou « Suivre » lorsqu'un article semble intéressant. Cet article est alors conservé dans les favoris, ou bookmarks. On peut supprimer les autres. L'algorithme de recherche affine ses résultats en fonction des choix que l'on fait. A la fin, on peut classer les articles conservés dans des rubriques que l'on crée.

Capture d'écran de Daily.Dev





## Quel est le but de cette veille ?

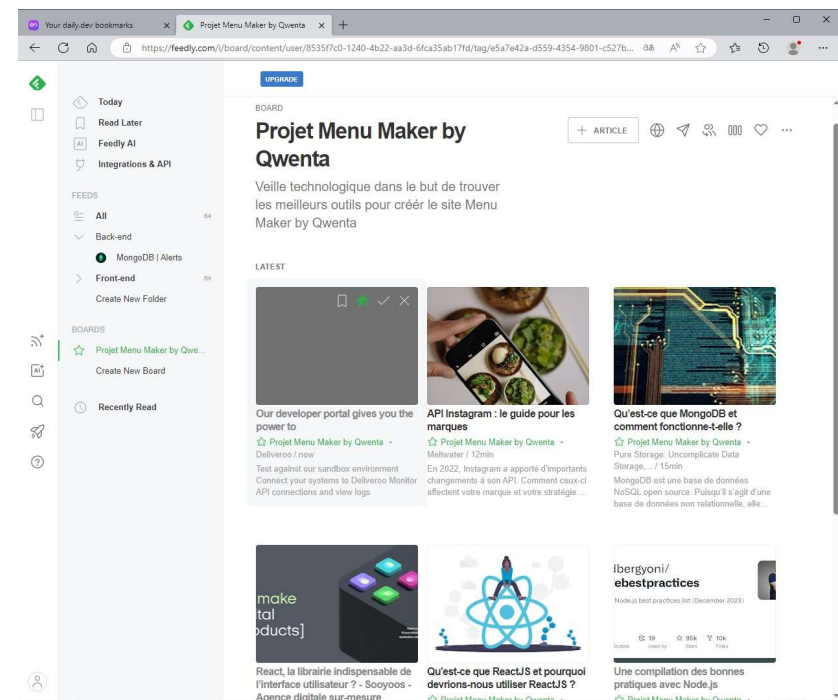
Le but de cette veille est de trouver les outils qui nous permettront de réaliser le projet. J'ai donc commencé à trier et classer les sources en fonction de leur catégorie :

HTML, CSS et JAVASCRIPT se sont bien sûr imposés, dès le départ, car ils correspondent à ce que nous recherchons pour créer notre site web. Pour le reste, c'est moins évident. Quel Framework choisir ? Quelle est la meilleure base de données ? Il y avait des choix à faire, parmi les plus utilisés et les plus en vogue.

- Front-end : REACT, VUE.JS, ANGULAR
- Back-end : DJANGO, EXPRESS.JS, LARAVEL, RUBY
- Base de données : MONGO.DB, MySQL, PostgreSQL

Dans plusieurs articles récents, j'ai rencontré une expression que je ne connaissais pas: Stack MERN. De quoi s'agissait-il ? Cela a retenu mon attention et excité ma curiosité.

### Capture d'écran de Feedly





## La Stack MERN : Idéale pour notre projet

La Stack MERN, est un acronyme pour MongoDB, Express.js, React et Node.js. Elle permet aux développeurs full stack de développer un site de A à Z sans avoir à faire intervenir une autre compétence. Ils peuvent ainsi créer un site web de bout en bout, ils gèrent à la fois le front-end et le back-end.

La stack MERN donne la possibilité de maîtriser à la fois la partie algorithmique et logique du backend, et la partie design, expérience utilisateur et animation de la partie front-end.

Le fait qu'un seul développeur puisse effectuer le travail de deux permet des économies de recrutement pour les entreprises. Cela semble correspondre parfaitement aux attentes de notre projet.

La veille technologique mise en place m'a permis de trouver les outils les plus pertinents pour le projet.

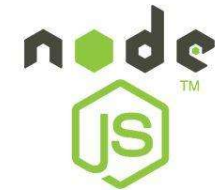


## Détails de la Stack MERN

- **React** est le Framework de développement front-end développé par Facebook. Il permet de mettre en place la partie client de l'application web, en utilisant une API qui permet de récupérer des données stockées en base de données. Ces données sont ensuite mises en place au sein des différentes pages de l'application web.

- Le serveur géré avec **Node** et son Framework **Express**. Node est un environnement d'exécution du langage Javascript hors d'un navigateur web. Un serveur Node est donc simplement une application en Javascript côté serveur. Quant à lui, le Framework Express est le Framework de référence sur Node permettant de gérer des requêtes HTTP et ainsi de faire fonctionner correctement le back-end.

- La base de données **MongoDB** est une base de données NoSQL basée sur des documents au sein de collections. À première vue, son format est plus simple à prendre en main qu'une base de données relationnelle. Il n'y a donc pas de requête SQL pour récupérer des données. Les fonctions définies par MongoDB permettent de remplacer et simplifier ce processus. MongoDB s'interface particulièrement bien avec les applications Javascript, ce qui fait sa popularité auprès de nombreux développeurs web.







## Outil de curation : Wakelet

---

Après avoir fait toutes ces recherches, je souhaitais les sauvegarder et les partager avec les membres de l'équipe. Pour cela, j'ai choisi un outil de curation nommé **WAKELET**. La curation de contenus est une pratique qui consiste à sélectionner, éditer et partager les contenus les plus pertinents du Web pour une requête ou un sujet donné.

Wakelet est une plateforme gratuite de stockage de contenu qui permet aux utilisateurs d'enregistrer, d'organiser et de partager ce même contenu sur Internet. Vous pouvez sauvegarder tout le contenu que vous trouvez sur Internet : articles, vidéos, messages issus des médias sociaux, des podcasts, des images, des notes, tout contenu numérique, etc... Wakelet vous permet également de collaborer avec d'autres personnes sur des collections, afin de partager des idées, de l'inspiration et des connaissances.

Dans notre projet, Wakelet permet de transmettre à chaque membre de l'équipe un lien d'accès (et un QRCode). Ils pourront alors consulter régulièrement cette page, et se tenir informés des derniers articles ajoutés. Dans sa version payante, Wakelet permet à ces personnes de participer, en ajoutant eux-mêmes des articles. Il est aussi possible d'envoyer régulièrement une newsletter aux membres de l'équipe.

---



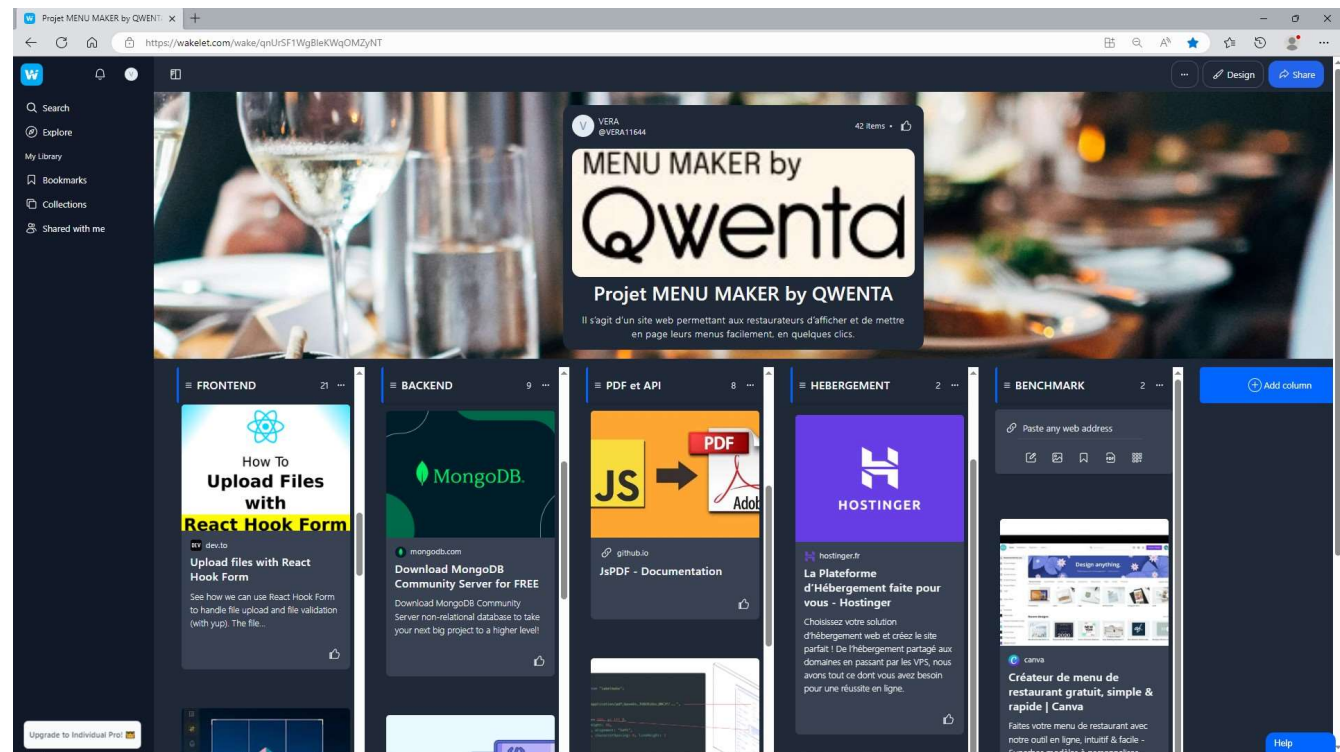
# Wakelet, pour organiser les informations

Dans la capture d'écran ci-jointe, on peut voir que j'ai pu personnaliser la présentation. J'ai choisi un titre ( Menu Maker by Qwenta ), un sous titre, une description et une image de fond qui correspond au projet.

J'ai créé 5 rubriques en rapport avec le sujet :

- Frontend
- Backend
- PDF et API
- Hébergement
- Benchmark

Chaque rubrique contient entre 3 et 20 articles. Il faut bien sûr continuer à alimenter les rubriques, lorsque de nouveaux articles apparaissent.



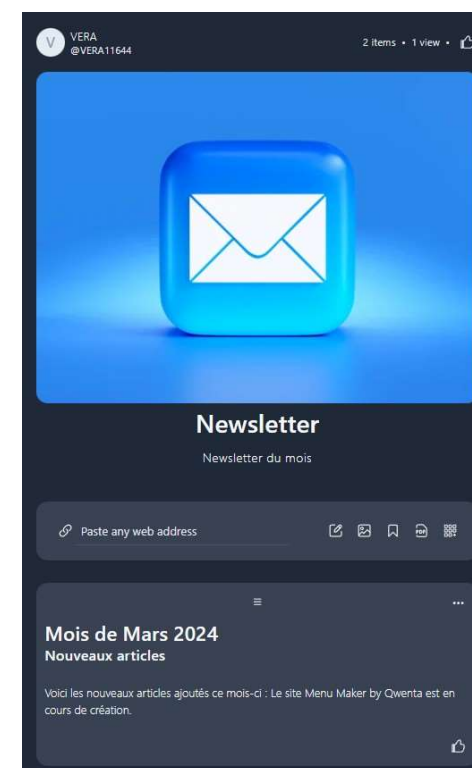
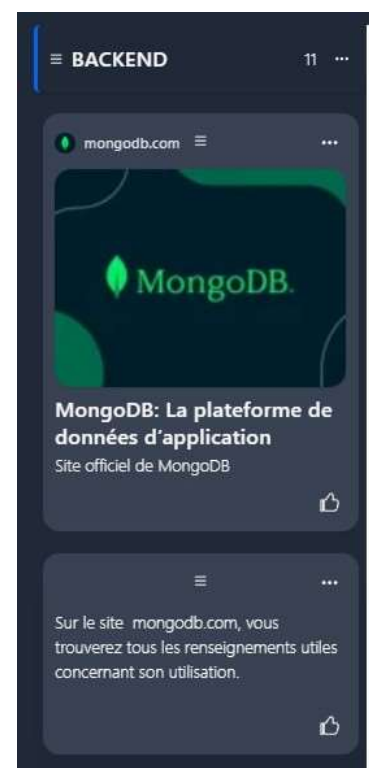
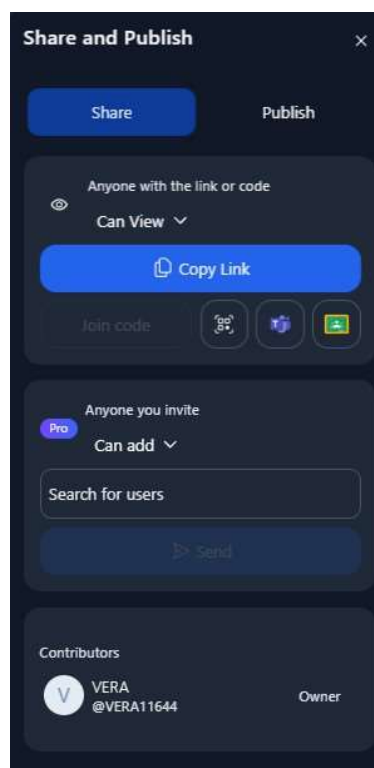


## Commenter, partager et diffuser l'information

Wakelet permet de partager les informations avec toute personne, en lui envoyant un lien de connexion, ou en créant un groupe d'utilisateurs. (Share and publish)

On peut aussi commenter chaque article, en lui ajoutant un texte et une image, une appréciation. (Commentaire ajouté sous l'article MongoDB)

On peut même créer une Newsletter, afin de tenir son équipe informée des nouveautés. On peut choisir la fréquence d'envoi de la Newsletter (Hebdomadaire, mensuelle).





# Conclusion

Pour conclure, je dirai que cette veille fut très constructive. Elle m'a permis de découvrir la méthode Stack MERN dont j'ignorais l'existence, et qui correspond parfaitement au Projet Menu Maker by Qwenta.

Bien sûr, il existe d'autres méthodes et d'autres langages, qui pourraient être utilisés sur ce projet.

L'utilisation de méthodes agiles permet de gérer un projet en s'adaptant aux besoins, aux changements du client et aux technologies en vogue.

La veille est donc primordiale, pour rester informé des nouvelles publications.

