

# Document de Veille Technologique – Menu Maker by Qwenta

**Auteur :** Mr LAURENT

**Date :** 29/09/2025

**Projet :** Menu Maker by Qwenta

## 1. Introduction

La veille technologique est une étape essentielle dans le développement du projet *Menu Maker by Qwenta*.

Elle permet de rester informé des avancées, des outils récents et des bonnes pratiques qui pourraient être appliquées dans la mise en place du site.

Cette veille a été réalisée avec **Feedly**, un agrégateur de flux RSS qui centralise les articles issus de plusieurs sources pertinentes.

Lien Feedly:

<https://feedly.com/i/collection/content/user/f4dc1599-d860-40f8-b7af-4d0cc8ed60ac/category/87beda3a-a5ba-4f6a-b672-84ba216ecd23>

## 2. Objectif de la veille

- Identifier les technologies adaptées au projet.
- Suivre les évolutions du web (frameworks, bibliothèques, sécurité, UI/UX).
- Détecter des solutions techniques permettant de répondre aux besoins définis dans les **spécifications techniques** et le **Kanban**.

## 3. Outil utilisé



Feedly est un agrégateur qui permet de suivre les publications de plusieurs sources (blogs, sites spécialisés, newsletters).

J'ai configuré Feedly en ajoutant différentes sources liées :


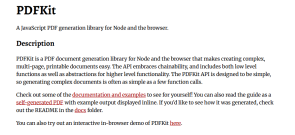
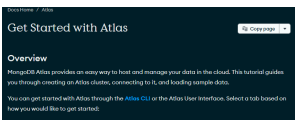
- Développement web (front-end & back-end)
- Sécurité & bonnes pratiques
- Tendances et innovations Tech



## 4. Sources principales suivies

Voici quelques exemples de sources ajoutées dans Feedly :

- **Smashing Magazine** – Articles sur le design web et les tendances front-end.
- **CSS-Tricks** – Ressources et astuces sur CSS et JavaScript.
- **Dev.to** – Communauté de développeurs partageant tutoriels et retours d'expérience.
- **Mozilla Developer Network (MDN)** – Documentation officielle pour HTML, CSS, JS.
- **Node.js Blog** – actualités [Node.js](#)
- **Zataz (FR)** – veille sécurité francophone

## 5. Résultats de la veille

Solution	Description	Justification	Source / Article Feedly
React + react-modal	Gestion interface et modales	Rendu dynamique + accessibilité	Article : <i>State management in React</i> 
jsPDF / pdfkit	Génération PDF	Compatible React/Node	Article : <i>How to generate PDFs in JS</i> 
MongoDB Atlas	DB Cloud	Flexible, scalable	Article : <i>Working with MongoDB Atlas</i> 

JWT + bcrypt + HTTPS	Auth sécurisée	Protection utilisateurs	Article : <i>JWT Auth in Node.js</i> 
CI/CD (Vercel/Heroku)	Pipeline automatisé	Déploiement fiable	Article : <i>Deploying with Vercel CI/CD</i> 

## 6. Conclusion

Cette veille technologique permet d'identifier rapidement les outils et bibliothèques les plus adaptés au projet *Menu Maker by Qwenta*.

En poursuivant cette démarche, nous pourrons :

- Optimiser le temps de développement.
- Améliorer la qualité et la maintenabilité du code.
- Suivre les évolutions pour garder le site compétitif.

## 7. Captures d'écran

# Feed d'actualité du Feedly Veille Technique:

Veille Technique

Most popular

AI

Do you want to track specific threats?

Critical Vulnerabilities

Threat actors

Malware Families

Cyberattacks

AMD and OpenAI announce strategic partnership to deploy 6 gigawatts of AMD GPUs

OpenAI partners with AMD for AI chips

42 Tendances & Innovation Tech / 6h

AMD and OpenAI have announced a multi-year partnership to deploy 6 gigawatts of AMD Instinct GPUs, beginning with 1 gigawatt in 2026, to power OpenAI's next-generation AI infrastructure and accelerate...

A 19-year-old nabs backing from Google execs for his AI memory startup, Supermemory

34 Tendances & Innovation Tech / 1h

19-year-old Dhravya Shah, who is originally from Mumbai, has built a memory layer for AI applications with Google's AI chief's backing.

Smashing Animations Part 5: Building Adaptive SVGs With '<symbol>', '<use>', And CSS Media Queries

Développement Web (Front-end & Back-end) / 2h

I've written quite a lot recently about how I prepare and optimise SVG code to use as static graphics or in animations . I love working with SVG, but there's always been something about them that bugs me. ...

# Image des différentes catégories:

Veille Technique

1K+

Tendances & Innovation Tech | The Verge

197

Cybersécurité & Bonnes pratiques | darkrea...

39

Cybersécurité & Bonnes pratiques | Have I B...

8

Cybersécurité & Bonnes pratiques | Krebs o...

6

Cybersécurité & Bonnes pratiques | ZATAZ

31

Développement Web (Front-end & Back-end...

12

Développement Web (Front-end & Back-end...

1K+

Développement Web (Front-end & Back-end...

3

Développement Web (Front-end & Back-end...

13

Développement Web (Front-end & Back-end...

15

Tendances & Innovation Tech | AI

23

Tendances & Innovation Tech | OpenAI

26

Tendances & Innovation Tech | TechCrunch

173

Images présentes dans le tableau en plus grande:

1.

[APPRENDRE REACT >](#)

## Gérer l'état

INTERMÉDIAIRE

Au fur et à mesure que votre application se développe, il est utile de réfléchir à la façon dont vous organisez votre état et à la façon dont les données circulent entre vos composants. Les états redondants ou dupliqués sont une source fréquente de bugs.

Dans ce chapitre, vous apprendrez à bien structurer votre état, à garder une logique de mise à jour de l'état maintenable, et à partager l'état entre des composants distants.

**Dans ce chapitre**

- Comment modéliser les changements d'interface en tant que changements d'état
- Comment bien structurer l'état
- Comment « faire remonter l'état » pour le partager entre les composants
- Comment contrôler si l'état est préservé ou réinitialisé
- Comment consolider une logique d'état complexe dans une fonction
- Comment transmettre l'information sans « faire percoler les props »
- Comment adapter la gestion d'état à la croissance de votre application

### Réagir à la saisie avec un état

Avec React, vous ne modifiez pas l'interface utilisateur directement à partir du code. Par exemple, vous

2.

## PDFKit

A JavaScript PDF generation library for Node and the browser.

### Description

PDFKit is a PDF document generation library for Node and the browser that makes creating complex, multi-page, printable documents easy. The API embraces chainability, and includes both low level functions as well as abstractions for higher level functionality. The PDFKit API is designed to be simple, so generating complex documents is often as simple as a few function calls.

Check out some of the [documentation and examples](#) to see for yourself! You can also read the guide as a [self-generated PDF](#) with example output displayed inline. If you'd like to see how it was generated, check out the README in the [docs](#) folder.

You can also try out an interactive in-browser demo of PDFKit [here](#).

### Installation

Installation uses the [npm](#) package manager. Just type the following command after installing npm.

```
npm install pdfkit
```

### Features

- Vector graphics
  - HTML5 canvas-like API
  - Path operations
  - SVG path parser for easy path creation
  - Transformations
  - Linear and radial gradients
- Text
  - Line wrapping (with soft hyphen recognition)
  - Text alignments
  - Bulleted lists

3.

## Get Started with Atlas

Copy page

### Overview

MongoDB Atlas provides an easy way to host and manage your data in the cloud. This tutorial guides you through creating an Atlas cluster, connecting to it, and loading sample data.

You can get started with Atlas through the [Atlas CLI](#) or the Atlas User Interface. Select a tab based on how you would like to get started:

Atlas CLI

Atlas UI

To create and authenticate with your Atlas account, create one free database, load sample data, add your IP address to your project IP access list, create a MongoDB user, and view your connection string using the Atlas CLI, run the following command:

```
atlas setup [options]
```

To learn more about the command syntax and parameters, see the Atlas CLI documentation for [atlas setup](#).

For step-by-step instructions on using this command, see [Get Started with Atlas from the Atlas CLI](#).

You can also run `atlas setup` if you have an Atlas account and an organization/project, but you

4.

DEV

Find related posts...


Powered by Algolia

56


0


35

...



**EidorianAvi**  
Posted on 14 déc. 2020

 53

 3

## Authentication and JWT in Node.js

[#auth](#) [#node](#) [#javascript](#)

Alright so this week I'm going to continue working with node. This one should be pretty short and sweet but I'd like to cover how to build out a login request and how to produce a token for verification in the front end. Let's get started.

### What's required

So for our form of authentication I'm going to be using Bcrypt and a JWT. What are those you ask? I'm glad you asked.


Bcrypt: A function that uses an algorithm to hash passwords. This is important for user security because if someone were to gain access to your database and the passwords are not hashed the users credentials are compromised.


JWT: JWT stands for JSON Web Token. It is a standard for authentication in applications. Upon a successful login the server sends a JWT to the client as proof of verification. Think of this as the ticket for a user to gain access to gated content or personal content.


5.

**DEV**

Find related posts...Powered by Algolia



**Matheus Bernardes Spilari**  
Posted on 16 sept. 2024

 3

## Github actions: Deploy on vercel using CI/CD

[#cicd](#) [#webdev](#) [#angular](#) [#githubactions](#)

In this article, we're going to cover what **CI** and **CD** are, and explain how easy it is to deploy an Angular or other framework app on Vercel using GitHub Actions.