

PRÉSENTATION

Menu Maker by Qwenta

Sommaire

1. Contexte du projet
2. Aperçu de la maquette
3. Méthodologie utilisée
4. Tableau Kanban
5. Spécifications techniques
6. Veille technologique
7. Conclusion
8. Questions

Contexte du Projet

Nom du projet : Menu Maker by Qwenta

Objectif :

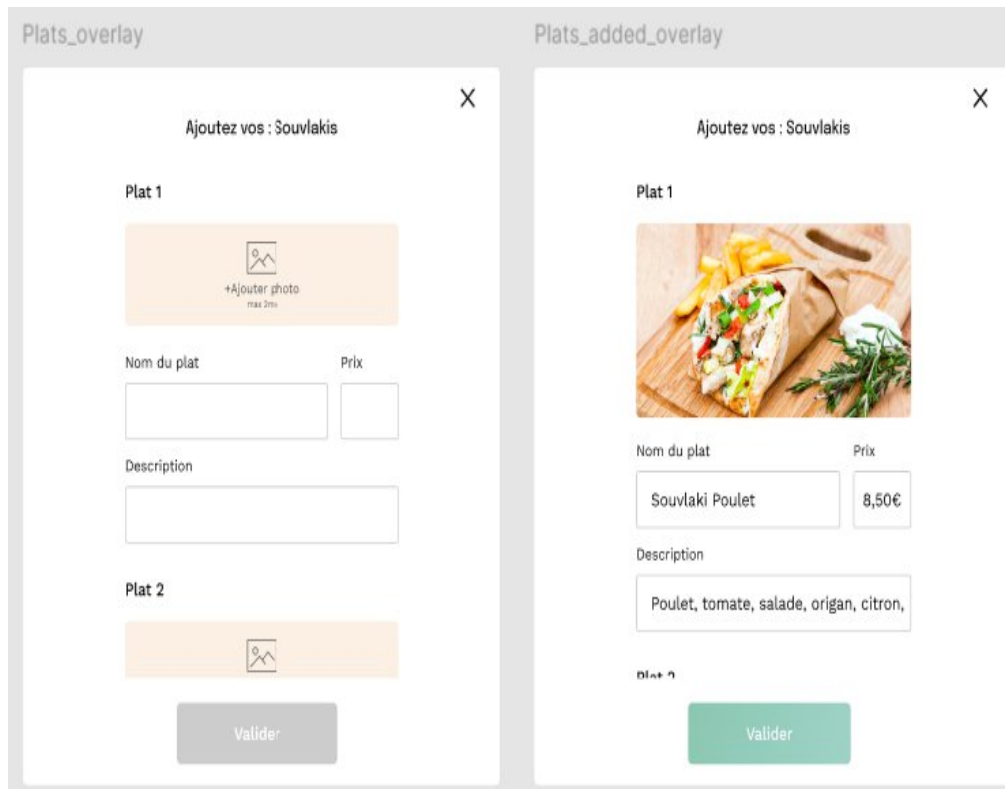
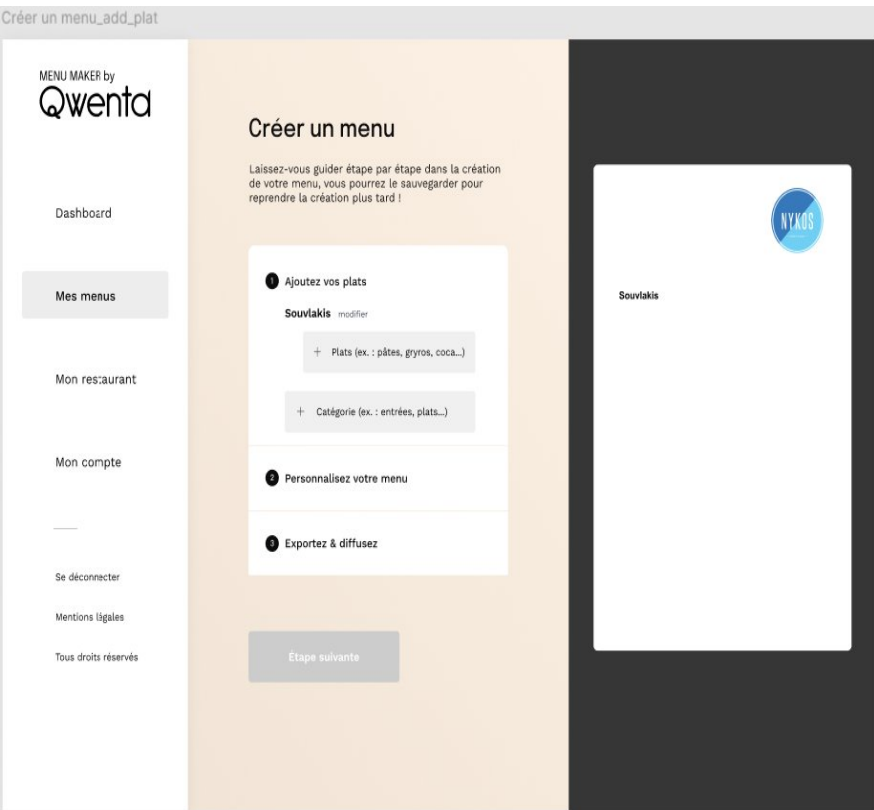
Personnalisation complète : Personnaliser le style visuel du menu en ajustant les couleurs, la typographie, et le branding (logo, couleurs du restaurant).

Multicanal : Offrir des solutions pour diffuser les menus sur plusieurs canaux, comme Deliveroo, Instagram, ou via des fichiers PDF pour l'impression physique.

Gestion simplifiée : Un tableau de bord permet de gérer toutes les étapes du processus, de la création à la diffusion, tout en conservant un historique des menus créés.

Exportation et diffusion rapides : En un seul clic, les restaurateurs peuvent partager leurs menus sur des plateformes partenaires ou les imprimer pour leur établissement.

Aperçu de la maquette



Créer un menu_font_color

MENU MAKER by
Qwenta

Dashboard

Mes menus

Mon restaurant

Mon compte

Se déconnecter

Mentions légales

Tous droits réservés

Créer un menu

Laissez-vous guider étape par étape dans la création de votre menu, vous pourrez le sauvegarder pour reprendre la création plus tard !

1 Ajoutez vos plats modifier

2 Personnalisez votre menu

Choisissez une typographie



Choisissez une couleur



3 Exportez & diffusez

Étape suivante

MENU

Souvlakis

Souvlaki/Poulet 8,50€
Poulet, tomate, salade, origan,
citron, tzatziki, jus de viande

Souvlaki/Poulet 8,50€
Poulet, tomate, salade,
origan, citron, tzatziki, jus
de viande

Souvlaki/Poulet 8,50€
Poulet, tomate, salade,
origan, citron, tzatziki, jus
de viande

Assiettes

Salade grecque 4,50€
Tomates, salade, feta,
concombre, huile d'olive

Gyros 11,50€
Agnou, tzatziki, pita,
frites & salade grecque

Koftedes 13,50€
Boulette de veau grecque,
servie avec frites & feta

Accompagnement

Frites à l'origan 3,50€

Olives Kalamata 3,50€

Boissons

Coca-Cola 33cl 2,50€

Citronnade maison 2,50€

Mythos 25cl 4,5% 4€

Thé glacé 3,50€

Ouzo 4cl 40% 8€

Jus d'orange frais 4,50€

Qwenta

Dashboard

Mes menus

Mon restaurant

Mon compte

Se déconnecter

Mentions légales

Tous droits réservés

Créer un menu

Laissez-vous guider étape par étape dans la création de votre menu, vous pourrez le sauvegarder pour reprendre la création plus tard !

1 Ajoutez vos plats modifier

2 Personnalisez votre menu modifier

3 Exportez & diffusez

Exporter en .pdf

Diffuser sur Deliveroo

Partager sur Instagram

Valider

MENU

Souvlakis

Souvlaki Poulet 8,50 €

Poulet, tomate, salade, origan, citron, tzatziki, jus de viande

Souvlaki Poulet 8,50 €

Poulet, tomate, salade, origan, citron, tzatziki, jus de viande

Souvlaki Poulet 8,50 €

Poulet, tomate, salade, origan, citron, tzatziki, jus de viande

Assiettes

Salade Grecque 4,50 €

Tomates, salades, feta, concombre, huile d'olive

Cyros 11,50 €

Agnou, tzatziki, pita, frites & salade grecque

Kefzedes 13,50 €

Boulette de veau grecque, servi avec des frites & feta

Accompagnement

Frites à l'origan 3,50 €

Olives Kalamata 3,50 €

Boissons

Coca-cola 33cl 2,50 €

Citronnade maison 2,50 €

Mythos 25cl 4,5% 4 €

Thé glacé 3,50 €

Ouzo 4cl 40% 8 €

Jus d'orange frais 4,50 €

Mes menus

MENU MAKER by
Qwenta

Dashboard

Mes menus

Mon restaurant

Mon compte

Se déconnecter

Mentions légales

Tous droits réservés

Mes menus



Créé le 3 juin

Modifier

Supprimer

+
Ajouter
un menu

Dashboard

MENU MAKER by
Qwenta

Dashboard

Mes menus

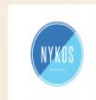
Mon restaurant

Mon compte

Se déconnecter

Mentions légales

Tous droits réservés



Bienvenue,
Nykos Greek Food

+ Créer un menu

+ Diffuser un menu

+ Imprimer un menu

Pour aller plus loin



Faites réaliser vos
supports par un
graphiste pro



Découvrez nos conseils
pour un menu réussi



Comment poster son
menu sur Instagram ?

Méthodologie utilisée

Méthodologie Agile & Scrum :

- Sprints de 2 semaines pour gérer les différentes fonctionnalités.
- Daily stand-ups pour suivre les progrès.
- Réunions de rétrospective pour améliorer le processus.

Collaboration en équipe : Interaction continue entre le Product Owner, les développeurs, et le QA Engineer.

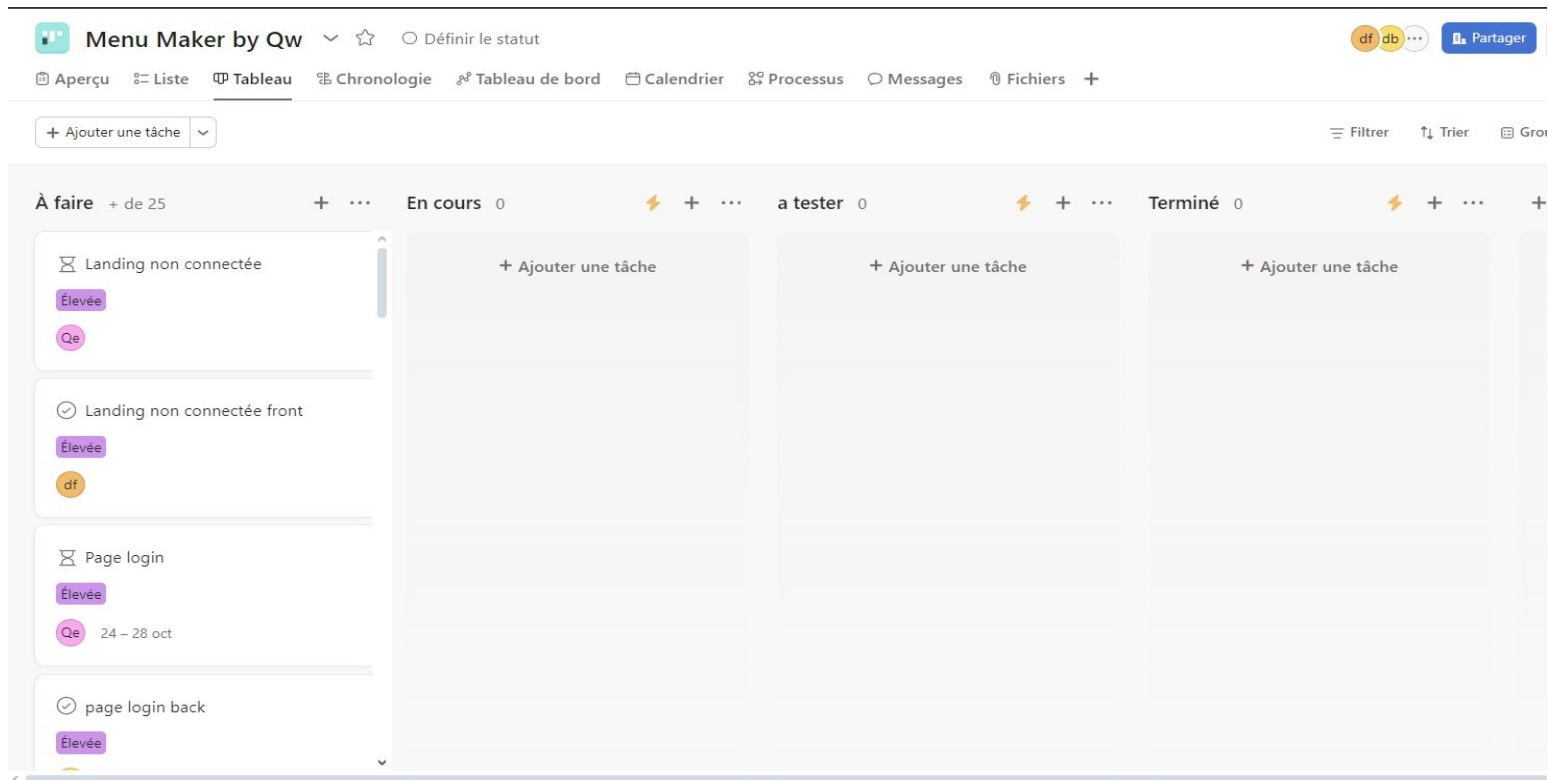
Outils utilisés : Notion pour la gestion des tâches, GitHub pour le versionnement.

Avantages dans le cadre de ce projet :

1. Flexibilité et adaptabilité :
 - la méthodologie Scrum a permis de s'adapter aux besoins changeants du client. Par exemple, des ajustements ont pu être faits en cours de développement.
2. Livraisons fréquentes :
 - Grâce à l'approche itérative, des versions fonctionnelles du produit ont été livrées régulièrement à la fin de chaque sprint. Cela a permis au client de tester les fonctionnalités en temps réel, d'identifier des ajustements ou des améliorations, et de fournir des retours concrets.
3. Collaboration continue :
 - Une communication constante avec le client et entre les membres de l'équipe. Les daily stand-ups permettent de résoudre les problèmes rapidement.
4. Amélioration continue :
 - Après chaque sprint, des rétrospectives permettent d'évaluer ce qui peut être amélioré. Ce qui permet d'améliorer la qualité du produit au fur et à mesure, tout en optimisant le travail de l'équipe.
5. Réduction des risques :
 - En divisant le projet en plusieurs sprint, les problèmes ou obstacles peuvent être identifiés et corrigés rapidement.
6. Implication du client :
 - Le client, Qwenta, est impliqué régulièrement. Ses retours peuvent être intégrés en temps réel, garantissant que le produit final répondait exactement à ses besoins.

Suivi du projet avec le Kanban

- Capture(s) d'écran du tableau Kanban.



The screenshot shows a Kanban board titled "Menu Maker by Qw". The interface includes a top navigation bar with various icons and a "Partager" button. Below the navigation bar, there are tabs for "Aperçu", "Liste", "Tableau" (selected), "Chronologie", "Tableau de bord", "Calendrier", "Processus", "Messages", and "Fichiers". A dropdown menu for "Ajouter une tâche" is visible on the left. The main area displays four columns: "À faire" (with 25 items), "En cours" (0 items), "a tester" (0 items), and "Terminé" (0 items). Each column has a header with a lightning bolt icon, a plus sign, and a three-dot menu. The "À faire" column contains four tasks: "Landing non connectée" (marked "Élevée" and "Qe"), "Landing non connectée front" (marked "Élevée" and "df"), "Page login" (marked "Élevée" and "Qe" with a date range "24 - 28 oct"), and "page login back" (marked "Élevée").

Suivi du projet avec le Kanban

Explication du Kanban

- **À faire** : Liste initiale des tâches.
- **Commencer** : Indique qu'une tâche est en cours de développement.
- **À tester** : Tâches prêtes pour les tests, avec notification automatique au QA.
- **Terminé** : Tâches complètes, prêtes à l'intégration finale.

1. Tâche principal

- **User Story** : Besoin utilisateur formulé comme "En tant que [utilisateur], je veux [fonctionnalité] afin de [bénéfice]".
- **Succès** : Conditions pour considérer la tâche comme réussie.
- **Scope** : Limites et fonctionnalités incluses pour éviter les dérives.
- **Côté QA Engineer** : Types de tests et checklist QA avec temps estimé.
- **Améliorations** : Suggestions d'ajouts futurs pour améliorer le produit.
- **Temps Total Estimé** : Durée totale pour achever la tâche.

2. Sous tâche :

- **Temps Estimé** : Durée totale pour achever la tâche.
- **Front-End** : Spécifications techniques front-end et estimation de temps.
- **Back-End** : Détails back-end, incluant les API et les validations.

Suivi du projet avec le Kanban

Le tableau Kanban facilite le suivi et la coordination de l'équipe grâce à plusieurs points clés :

1. **Visibilité claire** : Chaque tâche est divisée en sous-tâches front-end et back-end, offrant à chaque membre une vision précise de ce qu'il doit faire.
2. **Dépendances** : Les tâches principales sont bloquées jusqu'à ce que les sous-tâches front et back soient terminées, garantissant que toutes les étapes soient complétées avant les tests.
3. **Automatisation** : Les règles d'automatisation déplacent les tâches dans les sections correspondantes, réduisant les erreurs humaines et accélérant le processus de validation. Validation fluide et efficace avant la finalisation des tâches.
4. **Suivi en temps réel** : Les mouvements automatiques des tâches à travers les colonnes permettent à l'équipe de suivre l'avancement du projet en temps réel sans avoir à échanger constamment sur le statut de chaque tâche.

Ce tableau optimise donc la productivité et la communication entre les membres tout en assurant une progression fluide du projet.

lien du tableau : <https://app.asana.com/0/1208614697735217/1208614706853152>

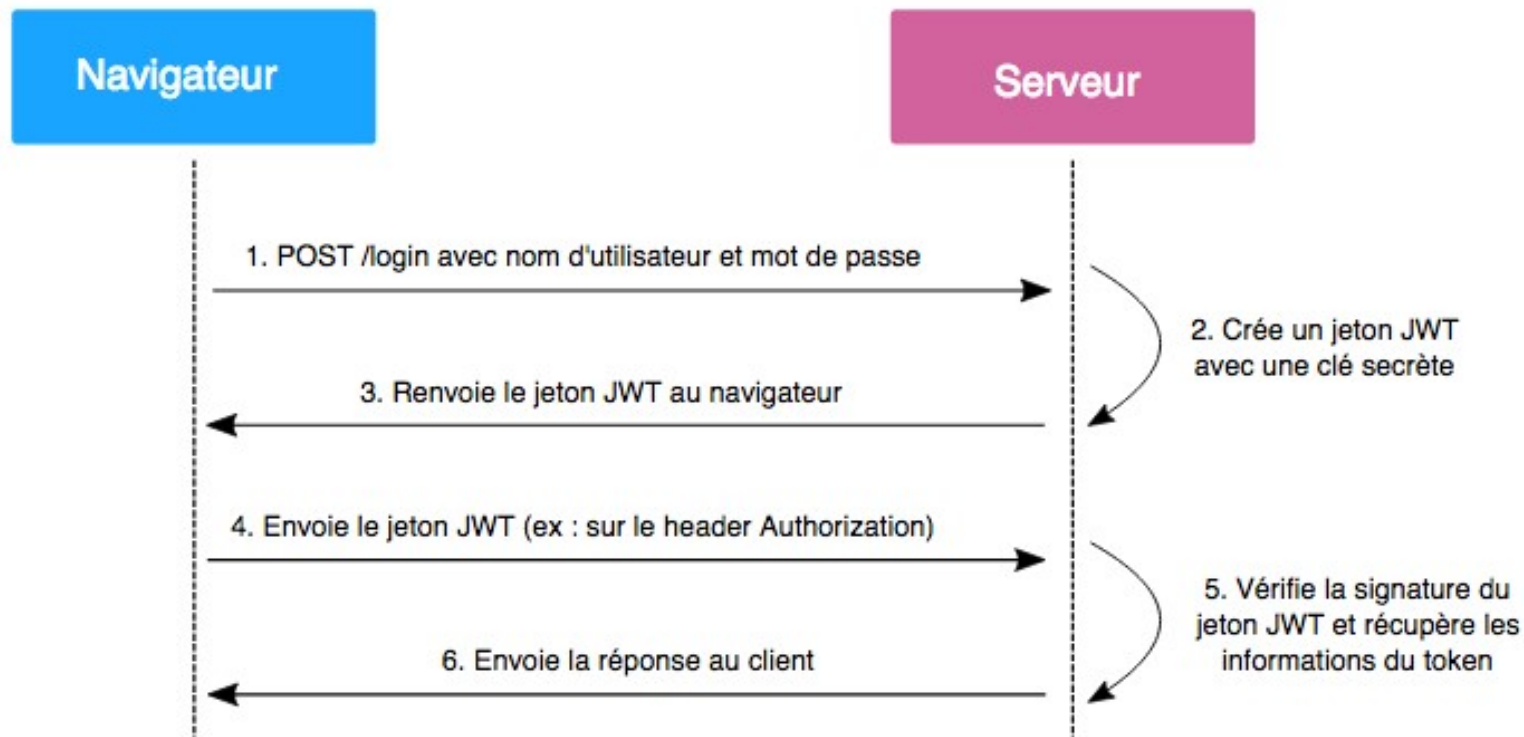
Spécifications techniques

- React pour le front-end avec la librairie react-modal.
- Node.js pour le back-end, intégrant des solutions comme html-pdf pour l'export PDF et JWT pour l'authentification.
- MongoDB avec Mongoose pour la gestion flexible des données.
- Deliveroo API pour la diffusion des menus sur Deliveroo.
- JWT pour les sessions utilisateurs et bcrypt pour le hashing des mots de passe.
- Utilisation de express-rate-limit et de Helmet.js pour renforcer la sécurité des requêtes et des en-têtes HTTP.
- Compatibilité avec les navigateurs principaux (Chrome, Firefox, Safari) et support des lecteurs d'écran (NVDA, VoiceOver).
- Gestion des thèmes avec Sass et pour l'ajustement des couleurs et du contraste.






Vulgarisation d'une spécification clé : Authentification avec JWT et bcrypt

- Lorsqu'un restaurateur se connecte, son identité est validée via bcrypt, qui compare son mot de passe crypté (hashé) avec celui stocké de manière sécurisée.
- Une fois validé, un JWT (JSON Web Token) est créé et envoyé au navigateur de l'utilisateur. Ce jeton sert de preuve que l'utilisateur est bien authentifié.
- Le JWT est léger et sécurisé, et permet à l'utilisateur de rester connecté en évitant de devoir entrer son mot de passe à chaque action.

Spécifications techniques



Veille Technologique

JavaScript Weekly

PortSwigger Blog

The A11Y Project

The OWASP Podcast Series

WebAIM Blog

Menu Maker Technologies

CSS-Tricks

DEV Community: javascript

DEV Community: react

Node.js Blog

The New Stack | DevOps, Open Source, a...

Vue.js Feed

deliveroo.engineering

MongoDB | Alerts

React

Release notes from express

Sass Blog

Tailwind CSS Blog

Create New Folder

BOARDS

Create New Board

Recently Read

Blog

Learn and Get Support



New Release 5.0.1

5.0.1

What's Changed

- remove --bail from test script by @jonchurch in #5962
- Nominate @bjohansebas to the triage team by...



What's Changed

- remove --bail from test script by @jonchurch in #5962
- Nominate @bjohansebas to the triage team by @UlisesGascon in #6009
- Link and update captains by @blakeembrey in #6013
- Update cookie semver lock to address CVE-2024-47764 by @joshbucker in #6017
- Release: 5.0.1 by @UlisesGascon in #6032

Full Changelog: [v5.0.0...5.0.1](#)

[Visit Website](#)

Feedly AI

Integrations & API

FEEDS

All1K+

✓ Développement web en général1K+

Articles on Smashing Magazine — For We...8

DEV Community1K+

JavaScript Weekly5

PortSwigger Blog

The A11Y Project

The OWASP Podcast Series

WebAIM Blog

✓ Menu Maker Technologies1K+

CSS-Tricks21

DEV Community: javascript1K+

DEV Community: react536

MongoDB | Alerts1

Node.js Blog4

Release notes from express2

The New Stack | DevOps, Open Source, a...144

Vue.js Feed3

4 more feeds

Create New Folder

BOARDS

Create New Board




MongoDB Server secondaries may crash due to forced index constraints (CVE-2024-8305)

✓ CVE-2024-8305 • MongoDB | Alerts / Oct 21, 2024 at 8:08 PM // keep unread // hide



Feedly detected 1 CVE. Automatically tag, enrich, and export CVEs, Threat Actors, Malware Families, TTPs, and IoCs

[Learn More](#)

prepareUnique index may cause secondaries to crash due to incorrect enforcement of index constraints on secondaries, where in extreme cases may cause multiple secondaries crashing leading to no primaries. This issue affects **MongoDB**  Server v6.0 versions prior to 6.0.17, MongoDB Server v7.0 versions prior to 7.0.13 and MongoDB Server v7.3 versions prior to 7.3.4

CVSS Score has been rated as 6.5

[Visit Website](#)

Feedly AI

Integrations & API

FEEDS

All1K+

Développement web en général1K+

Articles on Smashing Magazine — For We...7

DEV Community1K+

JavaScript Weekly5

PortSwigger Blog

The A11Y Project

The OWASP Podcast Series

WebAIM Blog

Menu Maker Technologies1K+

CSS-Tricks20

DEV DEV Community: javascript1K+

DEV DEV Community: react530

Node.js Blog4

The New Stack | DevOps, Open Source, a...144

Vue.js Feed3

deliveroo.engineering

MongoDB | Alerts

React

Release notes from express

Sass Blog

Tailwind CSS Blog

Create New Folder

Decoding WCAG: “Change of Context” and “Change of Content”

WebAIM Blog by Jon Whiting / Jul 31, 2024 at 7:11 PM // keep unread // hide

Introduction

As was mentioned in an earlier [blog post](#) on “Alternative for Time-based Media” and “Media Alternative for Text,” understanding the differences between terms in the Web Content Accessibility Guidelines (WCAG) is essential to understanding the guidelines as a whole. In this post, we will explore two more WCAG terms that are easily confused—change of context and change of content.

“Change of Context” and “Change of Content” in WCAG

WCAG defines changes of context as “major changes that, if made without user awareness, can disorient users who are not able to view the entire page simultaneously.” The most common examples are changing focus or creating a popup window.

Changes of context relate to four WCAG success criteria:

- 3.2.1 On Focus (Level A) – A change of context does not automatically occur when an element receives focus.
- 3.2.2 On Input (Level A) – A change of context does not automatically occur when changing an interactive component.
- 3.2.5 Change on Request (Level AAA) – Changes in context are always user-

Veille Technologique

Objectif de la veille :

S'assurer que Menu Maker intègre les meilleures pratiques en accessibilité, sécurité et performance, et reste à jour avec les tendances du développement web.

Méthode de Classification des Sources :

La sélection des sources repose sur quatre critères principaux :

1. Fiabilité et crédibilité : Choix de sources reconnues (ex. : Smashing Magazine, Node.js Blog) pour garantir l'exactitude des informations.
1. Pertinence pour le projet : Chaque source couvre un besoin essentiel, comme l'accessibilité ou la performance.
1. Actualité : Les sources publiant assurent régulièrement que les informations sont récentes.
1. Spécialisation par axe de veille : Organisation en deux axes de veille pour une couverture adaptée : *Développement web en général et Technologies spécifiques à Menu Maker* .

Veille Technologique

Exemple et explication du choix d'une source pour chaque axe :

1. Axe Développement Web en Général
 - Source : Smashing Magazine
 - Pourquoi : Articles pertinents pour l'accessibilité et l'UX, aidant à intégrer des pratiques comme la navigation au clavier et le support des lecteurs d'écran.
2. Axe Technologies de Menu Maker
 - Source : Blog Node.js
 - Pourquoi : Suivi des mises à jour de Node.js, essentiel pour les spécifications de performance et de sécurité du back-end.

Contribution de la Veille aux Spécifications Techniques :

La veille technologique joue un rôle essentiel dans l'élaboration des spécifications techniques en permettant d'anticiper les besoins et d'aligner le projet avec les normes de l'industrie. La veille garantit que les spécifications techniques intègrent les dernières améliorations. La veille permet le suivi des tendances et les recommandations officielles dans l'élaboration des Spécifications Techniques.

Conclusion

- Objectif : Digitaliser et personnaliser les menus pour les restaurateurs avec diffusion multicanal (Deliveroo, Instagram, PDF) via une interface simple.
- Méthodologie : Sprints Agile & Scrum avec Kanban pour le suivi des tâches (« À faire », « En cours », « À tester », « Terminé »). Résultat : livraisons fréquentes, flexibilité et amélioration continue.
- Technologies : React, Node.js, MongoDB, JWT et bcrypt (sécurité), Sass pour les thèmes, compatible principaux navigateurs et lecteurs d'écran.
- Veille Technologique : Sources fiables (ex. Smashing Magazine pour l'accessibilité, blog Node.js pour le backend), assurant une mise à jour continue avec les normes du secteur.

Conclusion : Projet aligné sur les besoins clients, progression fluide et optimisation des performances et de la sécurité.

QUESTIONS ?