

Rapport de testing

I. Recommandations

A. Tests à automatiser

Les deux fonctionnalités pour lesquelles j'ai retenu l'automatisation des tests sont :

1. La connexion
2. Le panier

J'ai sélectionné la fonctionnalité de connexion car elle permet d'accéder à la fonctionnalité primordiale du site, à savoir l'achat de produits. Sans s'être connecté au préalable, l'utilisateur n'a pas accès à son panier.

J'ai sélectionné la fonctionnalité du panier car en cas de dysfonctionnement de celle-ci, l'entreprise peut avoir des préjudices financiers. En effet, si un utilisateur commande des produits qui sont en réalité en rupture de stock, l'entreprise devra rembourser l'utilisateur car la livraison de sa commande ne pourrait être menée à bien. Outre le préjudice financier lié à cette conjecture, un tel cas est susceptible d'engendrer une forte insatisfaction de l'utilisateur et pourrait mener celui-ci à donner un mauvais avis sur le site (ou toute autre plateforme d'avis en ligne), ce qui impacterait négativement l'image de marque de l'entreprise.

A contrario, un problème décelé au niveau de l'affichage des produits n'engendrerait pas de préjudice financier, et serait moins coûteux à solutionner.

B. Préconisations pour la suite

Pour continuer à augmenter la couverture des tests, je préconise de tester plus de scénarios négatifs sur les points de terminaison API, comme suit :

- **POST /reviews** : tester l'entrée de valeurs invalides sur les différents champs du formulaire, avec une attention particulière sur le textarea afin de s'assurer qu'aucune faille XSS ou autre type de faille n'existe sur le site.
- **POST /login** : tester également que l'API renvoie une erreur en cas de données invalides pour s'assurer encore une fois qu'il n'y ait pas de faille de sécurité.


Automatiser ces tests API permettra de gagner du temps lors de l'exécution des tests et d'être plus confiants quant à la sécurité du site, d'autant que leur maintenance n'est pas très coûteuse.

Je préconise également d'ajouter le cas de test où l'utilisateur souhaite ajouter un produit à son panier avec la quantité 0. Ce cas de test pourrait être automatisé afin d'intégrer la suite de tests fonctionnels qui sont faits sur le panier d'une part, et d'autre part car la gestion des stocks et des commandes sont des fonctionnalités critiques du site. L'envoi d'une commande qui contiendrait un produit en quantité 0 pourrait créer de la confusion et faire perdre du temps et des ressources à l'entreprise.

Les tests fonctionnels sur l'affichage des produits sont moins urgents bien qu'aussi importants. Je préconise qu'ils soient automatisés afin de s'assurer que la vitrine du site contienne bien tous les produits vendus par l'entreprise et que les utilisateurs puissent bien prendre connaissance de chacun de ceux-ci. En outre, l'automatisation de ces tests n'est pas très complexe et permettra de gagner du temps lors de leur exécution.

II. Bilan de campagne de validation : tests automatisés

Document revu par :

Nom	Fonction	Version	Signature
Amandine	<i>Testeur manuel</i>	<i>1.0.0</i>	

Contexte :

Eco Bliss Bath sort sa première version de site vitrine en ligne de produits de beauté écoresponsables dont le produit principal est un savon solide.

Dans ce contexte, des premiers tests manuels ont été effectués par Marie, qui a suggéré d'en automatiser certains.

Les tests suggérés ont donc été automatisés et exécutés.

Objectif(s) :

- Valider le fonctionnement du site Web : s'assurer que le site fonctionne correctement et réponde aux attentes des utilisateurs.
- Déceler les anomalies et les corriger : identifier les problèmes et les anomalies du site Web pour qu'ils soient corrigés avant la mise en production.

Tests effectués :

* **Test 1** : Les smoke tests

- vérifie la présence du bouton de connection dans le header et des champs du formulaire de connection sur la page de connection
- vérifie la présence du bouton « Ajouter au panier » sur la fiche descriptive du produit lorsque l'utilisateur est connecté
- vérifie la présence du champs de disponibilité sur la fiche descriptive du produit

* **Test 2** : Les tests API

- Point de terminaison : **/login** :

[Scénarios positifs](#)

■ **POST** :

- connecter un utilisateur enregistré avec des identifiants valides :
 - « testApi@test.fr »
 - « testApi »

[Scénarios négatifs](#)

■ **POST** :

- tenter de connecter un utilisateur inexistant (attend une erreur 401) :
 - « doNotExist@test.fr »
 - « do*not*exist »

- Point de terminaison : **/orders** :

[Scénarios positifs](#)

Au préalable : L'utilisateur doit être connecté et le panier doit contenir un produit.

■ **GET** :

- récupérer le panier d'un utilisateur connecté

■ **PUT /add** :

- ajouter un produit disponible dans le panier

- **PUT /{id}/change-quantity :**
 - modifier la quantité d'un produit dans le panier
- **DELETE /{id}/delete :**
 - retirer un produit du panier
- **POST :**
 - envoyer une commande

Scénarios négatifs

Au préalable : Un nouvel utilisateur est enregistré (à l'aide de la création d'informations aléatoires) puis connecté au site

- **GET :**
 - tenter de récupérer le panier d'un utilisateur qui vient seulement d'être enregistré et qui n'a encore rien ajouté à son panier (attend une erreur 404)
 - tenter de récupérer le panier d'un utilisateur non connecté (attend une erreur 403)
- **PUT /add :**

Au préalable : modifier la quantité d'un produit pour 0

 - tenter d'ajouter un produit en rupture de stock au panier (attend une erreur 404)

- Point de terminaison : **/products :**

Scénarios positifs

- **GET :**
 - récupérer la liste des produits
- **GET /random :**
 - récupérer trois produits aléatoirement
- **GET /{id} :**
 - récupérer la fiche descriptive d'un produit

- Point de terminaison : **/reviews :**

Scénarios positifs

Au préalable : Connecter l'utilisateur

- **GET :**
 - récupérer la liste des avis
- **POST :**
 - poster un avis

- Point de terminaison : **/register :**

Scénarios positifs

Au préalable : Générer aléatoirement les données d'un utilisateur

- **POST :**

- enregistrer un nouvel utilisateur

- Point de terminaison : **/me** :

- [Scénarios positifs](#)

- Au préalable : Connecter l'utilisateur*

- **GET :**

- récupérer les informations de l'utilisateur connecté

✱ **Test 3 :** Les tests fonctionnels

- La connexion

- [Scénario positif](#)

- vérifie qu'un utilisateur enregistré est bien connecté après avoir correctement renseigné ses identifiants :

- cliquer sur le bouton de connexion
- la page de connexion avec le formulaire s'affiche
- entrer "test2@test.fr" dans le champ de l'email
- entrer "testtest" comme mot de passe
- vérifier la présence du token dans le local storage
- vérifier la présence du bouton « Mon panier » dans le header

- Le panier

- [Scénarios positifs](#)

- Au préalable : Connecter l'utilisateur*

- vérifie que le produit ajouté apparaît bien dans le panier de l'utilisateur :

- accéder à la page des produits (/products)
- cliquer sur le bouton « Consulter » du premier produit de la liste (« Sentiments printaniers »)
- vérifier la quantité de produits en stock
- cliquer sur le bouton « Ajouter au panier »
- vérifier la présence du produit dans le panier

- vérifie que le stock du produit a bien diminué d'un

- accéder à la page des produits (/products)
- cliquer sur le bouton « Consulter » du premier produit de la liste (« Sentiments printaniers »)
- vérifier que la quantité de produits en stock a diminué d'un

- vérifie que le stock d'un produit qui est retiré du panier revient bien à son état originel

- accéder au panier

- cliquer sur l'icône de corbeille du produit « Sentiments printaniers »
- accéder à la page des produits (/products)
- cliquer sur le bouton « Consulter » du premier produit de la liste (« Sentiments printaniers »)
- vérifier que la quantité de produits en stock est revenue à son état originel

Scénarios négatifs

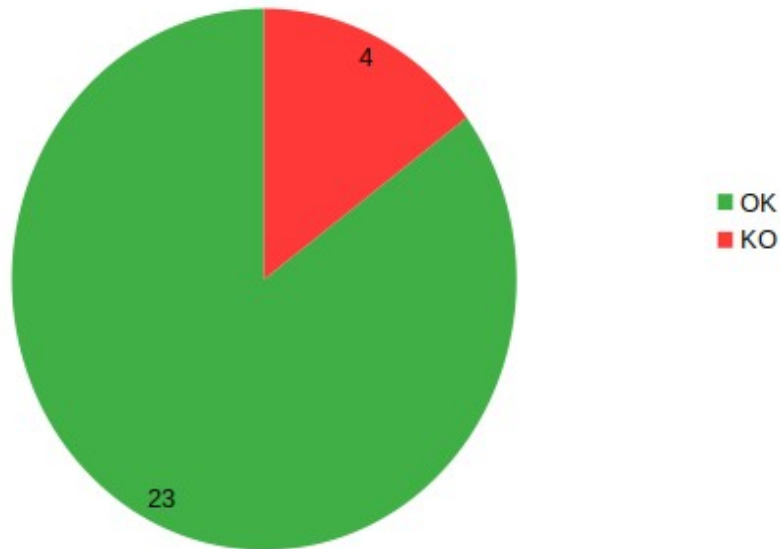
Au préalable : Connecter l'utilisateur

- vérifie qu'un produit dont le stock est inférieur à 1 ne peut pas être ajouté au panier :
 - modifier la quantité du produit 3 pour 0
 - accéder à la page des produits (/products)
 - cliquer sur le bouton « Consulter » du premier produit de la liste (« Sentiments printaniers »)
 - vérifier que le champs de disponibilité affiche « En rupture de stock »
 - vérifier que le bouton « Ajouter au panier » est désactivé
- vérifie qu'on ne peut pas ajouter moins de 0 produit dans le panier :
 - accéder à la fiche descriptive du produit 3 (/products/3)
 - modifier la quantité pour « -1 »
 - vérifier que l'input qui permet de modifier la quantité est invalide
- vérifie qu'on ne peut pas ajouter plus de 20 instances d'un même produit dans le panier :
 - accéder à la fiche descriptive du produit 3 (/products/3)
 - modifier la quantité pour « 21 »
 - vérifier que l'input qui permet de modifier la quantité est invalide

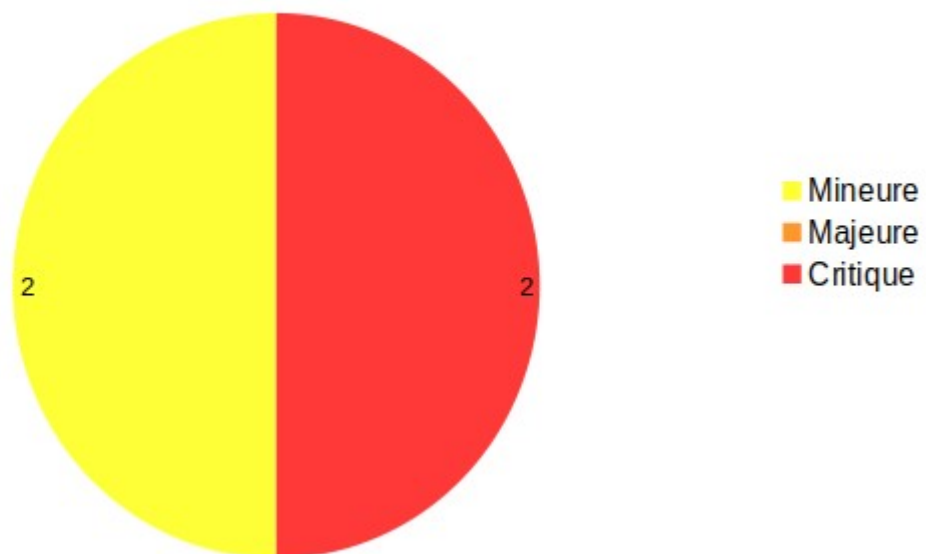
Résultats des tests

Principaux résultats - résumé

Résultats des tests



Criticité des anomalies



- Fonctionnalités en échec :

- API /orders
- panier

- Principaux bugs :

- EBB-1 Un produit en rupture de stock peut être ajouté au panier
- EBB-2 : Appel API /orders/add retourne 200 quand le produit est en rupture de stock

Résultats complets

- API /orders

- EBB-2 : Appel API /orders/add retourne 200 quand le produit est en rupture de stock

- **Description :**

Lorsqu'on fait un appel API pour essayer d'ajouter un produit en rupture de stock dans le panier, le serveur nous renvoie une réponse dont le code de statut est 200 (bien effectuée) au lieu de 404 (non trouvé).

- **Etapes de reproduction :**

1. changer le stock du produit 3 pour le mettre à 0
2. se connecter avec les identifiants suivants :
 - "testApi@test.fr"
 - "testApi"
3. appeler l'API pour ajouter ce produit au panier, avec la méthode "PUT" sur la route "/orders/add" avec :
 - headers = "{ 'Authorization': 'Bearer {token}' }"
 - body = "{ 'product': 3, 'quantity': 1 }"
4. récupérer le code de statut de la réponse

- **Résultat attendu :**

Le code de statut retourné par le serveur est 404 (non trouvé).
Le corps de la réponse contient un message "En rupture de stock".

- **Résultat obtenu :**

Le code de statut retourné par le serveur est 200 (bien effectuée).

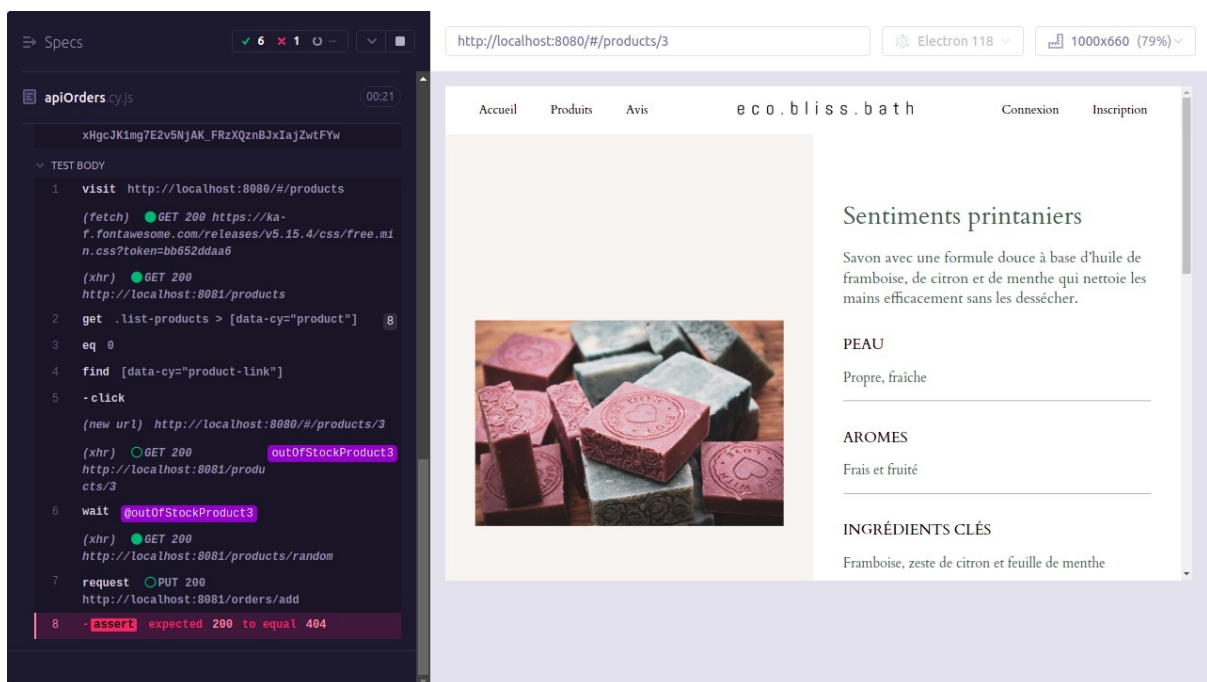
- **Analyse :**

Le corps de la requête ne contient que l'id du produit et la quantité à ajouter au panier. Si au lieu de ne contenir que l'id, le corps de la requête contenait le produit complet, peut-être le

serveur pourrait-il vérifier que le “availableStock” est supérieur à 0 pour répondre avec un code de statut 200 ?

Egalement, l’API ne prévoit pas de code et de message d’erreur pour ce cas. Il serait justicieux d’ajouter un code 404 accompagné d’un message “En rupture de stock”.

- **Criticité :** Critique
- **Priorité :** Majeure
- **Niveau :** Back-end
- **Captures d’écran :**



- EBB-4 : Appel API /orders retourne 401 quand l'utilisateur n'est pas connecté

- **Description :**

Lorsqu'on fait un appel API pour essayer de récupérer le panier d'un utilisateur qui n'est pas connecté, le serveur nous renvoie une réponse dont le code de statut est 401 (non authentifié) au lieu de 403 (non autorisé).

- **Étapes de reproduction :**

1. appeler l'API avec la méthode “GET” sur la route “/orders” sans header “Authorization”
2. récupérer le code de statut de la réponse

- **Résultat attendu :**

Le code de statut retourné par le serveur est 403 (non autorisé).

- **Résultat obtenu :**

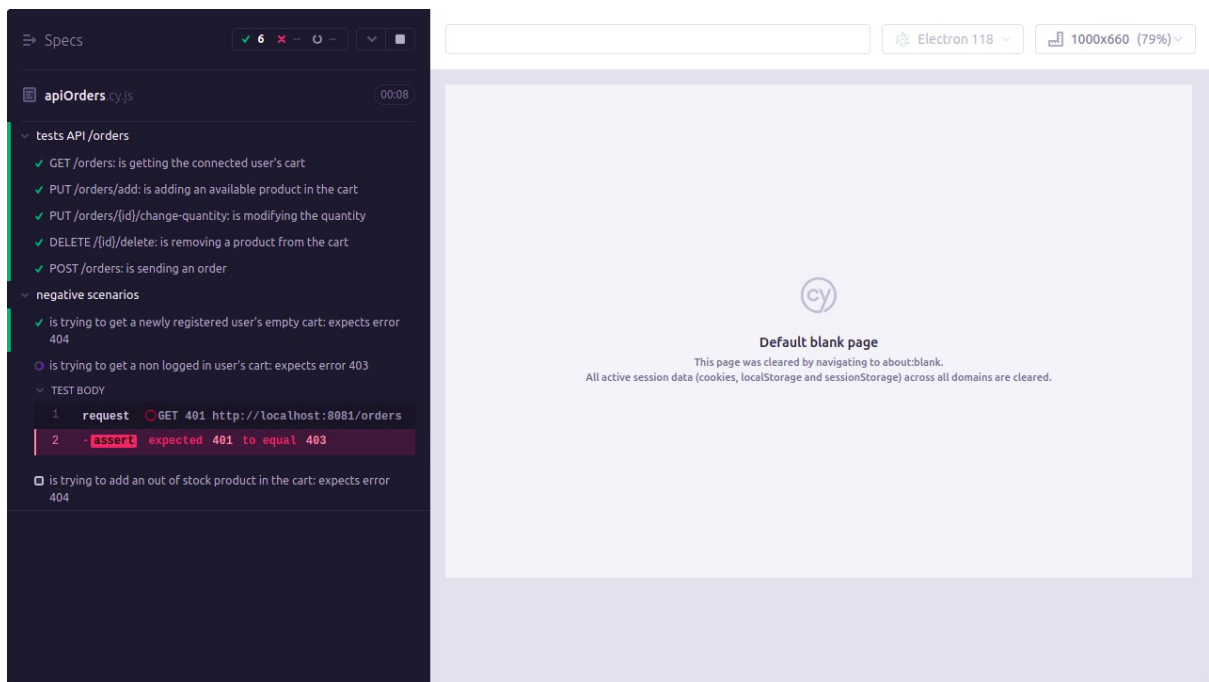
Le code de statut retourné par le serveur est 401 (non authentifié).

- **Criticité :** Mineure

- **Priorité :** Mineure

- **Niveau :** Back-end

- **Captures d'écran :**



- panier

- EBB-1 Un produit en rupture de stock peut être ajouté au panier

- **Description :**

Le stock d'un produit continue à diminuer indéfiniment. Ainsi, sur la page de description d'un produit, le stock peut être un nombre négatif (ex : -10), et l'utilisateur peut continuer à ajouter de ce produit dans son panier.

- **Etapes de reproduction :**

- 1. changer le stock du produit 3 pour le mettre à 0

- 2. se connecter avec les identifiants suivants :

- "test2@test.fr"

◦ “testtest”

3. aller sur la page de ce produit : /products/3

4. vérifier l’affichage de la disponibilité du produit

5. cliquer sur le bouton “Ajouter au panier”

■ **Résultat attendu :**

Le champs de disponibilité du produit affiche “En rupture de stock” et le bouton “Ajouter au panier” est désactivé, de sorte à ce que l’utilisateur ne puisse pas ajouter de ce produit à son panier.

■ **Résultat obtenu :**

La section de disponibilité du produit affiche “0 en stock” et l’utilisateur peut ajouter de ce produit à son panier en cliquant sur le bouton “Ajouter au panier”.

■ **Criticité :** Critique

■ **Priorité :** Majeure

■ **Niveau :** Front-end

■ **Captures d’écran :**

The screenshot displays a web browser window showing the product page for "Sentiments printaniers" on the website "ecobliss.bath". The product is priced at 60,00 € and has a stock status of "-265 en stock". The "Ajouter au panier" button is visible. The browser address bar shows "http://localhost:8080/#/products/3".

Below the browser window, a terminal window shows the Cypress test results. The test steps include:

- GET 200 http://localhost:8081/products
- GET 200 http://localhost:8081/products/3
- GET 200 http://localhost:8081/products/random
- GET [data-cy=detail-product-add]
- Assertion failure: `expected <button> to be disabled`

The terminal also lists several assertions that passed, such as "An order line with the added product is created in the cart" and "The added product stock amount has decreased by one".

60,00 €
-265 en stock

1

Ajouter au panier

- EBB-3 Un utilisateur peut ajouter un produit en quantité supérieure à 20 dans son panier

- **Description :**

Un utilisateur peut ajouter plus de 20 instances d'un même produit dans son panier.

- **Etapes de reproduction :**

1. se connecter avec les identifiants suivants :

- "test2@test.fr"
- "testtest"

2. aller sur la page du produit 3 : /products/3

3. changer la quantité pour "21"

4. cliquer sur le bouton "Ajouter au panier"

- **Résultat attendu :**

Le bouton "Ajouter au panier" est désactivé afin d'empêcher l'utilisateur d'ajouter une si grande quantité d'un même produit dans son panier.

- **Résultat obtenu :**

L'utilisateur peut ajouter 21 instances du produit dans son panier en cliquant sur le bouton "Ajouter au panier".

- **Criticité :** Mineure

- **Priorité :** Moyenne

- **Niveau :** Front-end

- **Captures d'écran :**

The screenshot displays a Cypress test runner on the left and a web application on the right. The test runner shows a test for adding products to a cart, with a failure on the 7th step. The web application shows the product page for 'eco.bliss.bath' with a quantity of 21 selected and the 'Ajouter au panier' button.

Test Runner (Cypress):

- Specs: 5 passed, 1 failed, 0 pending, 0 skipped, 0 disabled, 0 errored, 0 pending, 0 skipped, 0 disabled, 0 errored
- Test Name: cannot add more than 20 products in the cart
- BEFORE EACH: 1. request POST 200 http://localhost:8081/login, 2. window
- TEST BODY: 1. visit http://localhost:8080/#/products/3, 2. get [data-cy=detail-product-quantity], 3. - assert expected <input.ng-untouched.ng-pristine.ng-valid> to have value '1', 4. - assert expected <input.ng-untouched.ng-pristine.ng-valid> to have class ng-valid, 5. - clear, 6. - type 21, 7. - assert expected <input.ng-untouched.ng-dirty.ng-valid> to have class ng-invalid

Web Application (eco.bliss.bath):

- Navigation: Accueil, Produits, Avis, Mon panier, Déconnexion
- Product Image: A stack of colorful soaps.
- Product Name: PEAU
- Description: Propre, fraîche
- Aromes: Frais et fruité
- Ingrédients Clés: Framboise, zeste de citron et feuille de menthe
- Price: 60,00 €
- Quantity: 21
- Button: Ajouter au panier

Confiance

La version actuelle du site n'est pas en état d'être mise en production car le site contient deux anomalies critiques à corriger en priorité qui concernent le stock de produits. En effet, le fait que les utilisateurs puissent ajouter à leur panier des produits en rupture de stock peut entraîner de sérieux impacts sur les finances et la réputation de l'entreprise.