

ACTION	FONCTION TESTÉE	RESULTAT	Problème POSSIBLE	FICHIER	LIGNES
Démarrage du serveur					
Connexion au serveur par le biais de la console : npm node.js	/	Réponse de la console : "Listening on port 3000"	Vérifier que la réponse de la console est bien "Listening on port 3000"	/	/
Page index.html					
Récupération des informations des produits	GET/	Le serveur retourne la liste des produits de l'API en format JSON	Une erreur apparait sur la console si la récupération des données a échouée	script.js	1 à 5
Affichage des informations des produits	productsData(allProducts)	Le serveur retourne la liste des produits de l'API en format JSON	Aucun produit ne s'affiche sur la page	script.js	8 à 69
Page products.html					
Récupération des informations du produit	GET/\${productId}	Le serveur retourne les informations du produit récupéré dans l'API à l'aide de son ID	Une erreur apparait sur la console si la récupération des données a échouée	product-page.js	1 à 20
Affichage des informations du produit	productsData(currentProducts)	Les informations du produit sur lequel l'utilisateur a cliqué sont affichées	Les informations du produit sur lequel l'utilisateur a cliqué ne s'affichent pas	product-page.js	13 à 62
Construction du panier	addToCartProduct()	Création d'un objet "productInfo"	Les informations du produits ne sont pas ajoutés dans le LocalStorage	product-page.js	79 à 82
Clic sur "ajouter au panier" lorsque ce dernier est vide	addEventListener ("click")	Ajoute l'objet sur lequel on a cliqué dans le localStorage	L'objet ne s'ajoute pas lorsque le panier est vide	product-page.js	85 à 88
Clic sur "ajouter au panier" lorsque le panier n'est pas vide et le produit sur lequel on a cliqué est identique à un produit du panier	addEventListener ("click")	Ajoute la quantité sélectionnée avec l'ancienne située dans le Panier sans créer de nouveau produit	La quantité ne s'ajoute pas et un nouveau produit est créé	product-page.js	91 à 101
Clic sur "ajouter au panier" lorsque le panier contient au minimum 1 article et que un de ces derniers possède le même ID mais une couleur différente du produit que l'on souhaite ajouter	addEventListener ("click")	Le produit est ajouté dans le panier sous la forme d'un nouveau produit même si l'ID est le même qu'un produit déjà présent dans ce même panier	Aucun produit n'est créé lorsque l'ID est identique, même si la couleur est différente	product-page.js	103 à 108
Clic sur "ajouter au panier" lorsque le panier contient au moins 1 article mais aucun de ces derniers n'a le même ID que le produit que l'on souhaite ajouter	addEventListener ("click")	Le produit est ajouté dans le panier sous la forme d'un nouveau produit	Aucun produit n'est ajouté au panier	product-page.js	103 à 108

Création d'un message d'erreur si la quantité ou la couleur ne sont pas renseignées	addEventListener ("click") orderValidIf ()	Message d'alerte "Veuillez choisir une couleur et/ou une quantité"	Le produit est quand même ajouté au panier	product-page.js	115 à 117
Création d'un message d'erreur si la quantité est supérieur à 100 ou inférieur à 0	addEventListener ("click") orderValidIf ()	Message d'alerte "Veuillez choisir une quantité comprise entre 1 et 100"	Le produit est quand même ajouté au panier	product-page.js	119 à 121
Création d'une alerte lorsque le produit a bien été ajouté au panier	addEventListener ("click") orderValidIf ()	Message d'alerte "Votre article a bien été ajouté au panier"	Le produit n'a pas été ajouté au panier	product-page.js	123 à 125
Page cart.html					
Récupération des produits présents dans le LocalStorage	let productInCart	Les produits du localStorage ont bien été récupérés	Les produits n'ont pas été récupérés	cart.js	1 à 3
Masque le formulaire si le panier est vide	productOrder() if (productInCart === null)	Masque le formulaire lorsque le panier est vide	Le formulaire est toujours présent même lorsque le panier est vide	cart.js	8 à 15
Affichage des informations des produits dans le panier	else productInCart.forEach()	Les produits du localStorage s'affiche sur la page panier	Les produits du localStorage ne s'affiche pas	cart.js	18 à 209
Récupération des informations n'étaient pas renseignées dans le localStorage pour les produits dans le panier	GET/\${product.id}	Toutes les informations des produits du localStorage ont été récupérées via l'API	Seulement les informations présentes dans le LocalStorage s'affichent	cart.js	22 à 25
Affiche la quantité total des produits du panier	changeQuantityProduct()	La quantité totale est affichée	La quantité totale ne s'affiche pas	cart.js	123 à 128
Affiche le prix total des produits du panier	changeCartPrice()	Le prix total est affiché	Le prix total ne s'affiche pas	cart.js	130 à 135
La quantité totale change selon le nombre d'articles dans le panier, de plus, le prix totale change selon le nombre d'articles dans le panier	updatePrice() .addEventListener("change")	Le nombre de produit total change instantanément est doit être identique au nombre de produit dans le panier, de plus, le prix total change instantanément si la quantité d'un produit change	Le prix et la quantité totale des articles du panier ne se met pas à jour	cart.js	149 à 157

Clic sur le bouton "Supprimer"	<pre>.addEventListener("click", () => deleteProductInCart(), deleteArticle() deleteProductInStorage () removeForm ()</pre>	Le produit est supprimer du LocalStorage et de la page panier Si le panier est vide supprime le formulaire	Le produit est toujours présent dans le localStorage et/ou sur la page panier	cart.js	190 à 204
Clic sur le bouton "Envoyer"	<pre>.addEventListener('click', (stopRefresh) => submitForm(stopRefresh))</pre>	La page ne se rafraichit plus lors du clic sur le bouton "Envoyer"	La page se rafraichit lors du clic sur le bouton "Envoyer"	cart.js	190 à 194
Clic sur le bouton "Envoyer"	<pre>.addEventListener('click', (stopRefresh) => submitForm(stopRefresh))</pre>	Si le localStorage est vide ajout d'une alerte "Veuillez choisir un produit"	Le local storage n'est pas vide	cart.js	217 à 220
Clic sur le bouton "Envoyer"	<pre>formValid() if (isFormInvalid() isFirstNameInvalid() isLastNameInvalid() isAdressInvalid() isCityInvalid() isAdressInvalid() isEmailInvalid() === true) return</pre>	Si une des fonctions renvoie 'true' alors la requête ne se 'execute pas et l'order n'est envoyé à l'API	La requête a été envoyé et nous sommes redirigés à la page suivante	cart.js	231 à 347
Clic sur le bouton "Envoyer"	<pre>pushIdForPost()</pre>	Récupère l'ID de tous les objets présent dans le localStorage	L'ID n'a pas été récupéré	cart.js	300 à 306
Clic sur le bouton "Envoyer"	<pre>makeRequest ()</pre>	Création d'un objet "contact" et d'un objet "product" qui seront envoyé via une requête POST	Les objets n'ont pas été crée	cart.js	285 à 298
Clic sur le bouton "Envoyer"	<pre>POST/</pre>	L'ID du produit et les informations du formulaire sont envoyées à l'API via une requête POST	Les données n'ont pas pu être envoyé	cart.js	208 à 222
Confirmation.html					
Récupère l'URL de l'order	<pre>getOrder()</pre>	Récupère l'ID de l'order via l'URL de la page	L'order ID n'a pas pu être récupéré	confirmation.js	5 à 10

Affiche l'order	displayOrder()	Affiche l'ID de l'order récupéré sur la page	L'order ID ne s'affiche pas	confirmation.js	12 à 16
Vide le panier	removeCart()	Supprime les articles du LocalStorage	Les produits n'ont pas été supprimés du LocalStorage	confirmation.js	18 à 23

Nombre de ligne de code	Nombre de ligne testées	Pourcentage de code testé
605	592	97%

Légende
Action de l'utilisateur
Action automatique