

Plan test d'orinoco

N°	Page testée	Test	Ligne testées	Résultat attendu	Résultat
1	index.html	L'utilisateur se connecte sur index.html Les caméras s'affichent avec la description, l'image, le titre, le prix	index.js ligne 57 products().then(function(products)	Les produits s'affichent avec le l'image , le nom et le prix	OK
2	index.html	Chaque appareil photo est muni d'un bouton qui renvoie vers un lien unique et qui renvoie sur la page products.html	on récupère dans le script"produits js" de ligne 18 à 38	On obtient l'id de la caméra qui apparaît dans l'URL et il n'y a qu'une caméra présente avec son image, titre, prix, description, bouton ajout au panier, menu déroulant des lentilles	OK
3	products.html	Quand l'utilisateur click sur le bouton "ajouter au panier" dans le local storage un tableau apparaît contenant un ID correspondant à l'id dans l'URL	products.js : "ajoutPanier,addEventListener('click, function()'" ligne 110 à 118	Dans le LocalStorage un tableau contenant un seul id correspondant à l'id de l'URL	OK
4	products.html	L'utilisateur qui a maintenant un produit dans son panier, si l'utilisateur souhaité rajouter un autre produit celui-ci doit retourner dans la page index.html et de rajouter un autre produit	products.js : fonction nombreProduit() ligne de 121 à 132	dans le LocalStorage tableau contenant deux id correspondant à l'id de l'URL avec en dernier l'id correspondant à l'id dans l'URL	OK
5	panier.html	Le panier présente les caméras (image, prix unitaire et total et quantité)	panier.js : ligne de 24 à 45	Deux articles sont présents	OK
6	panier.html	L'utilisateur arrivé sur le panier avec 2 caméras il souhaite supprimer la dernière caméra qu'il l'ajoute on appelle la fonction deleteButton	panier.js ligne de 104 à 156	La quantité pour chaque objet est passé de 2 à 1 le total est mis à jour	OK
7	panier.html	Si l'utilisateur actualise la page avec les deux caméras doit être toujours présent	panier.js :fonction chargementPanier ligne de 3 à 13	rien ne doit changer	OK
8	panier.html	Si l'utilisateur supprime tous ses articles à l'aide de la fonction delete Button	panier.js ligne de 104 à 156	la page doit afficher "votre panier est vide" à la place des articles	OK
9	panier.html	Si l'utilisateur clique sur valider "mon panier "alors que le panier est vide une alerte doit s'afficher	panier.js : fonction achat ligne de 163 à 172	Un message d'alerte s'affiche "votre panier est vide, impossible de passer une commande"	OK
10	panier.html	Si l'utilisateur à un objet dans son panier et souhaite valider son panier une alerte de formulaire doit s'afficher en rouge	panier.js: ligne 251 à 278 " return event.preventDefault"	Indication disant "format de votre prénom incorrect..." affiché en rouge"	OK
11	panier.html	L'utilisateur valide le formulaire et clique sur valider mon panier	panier.js: variable request ligne 283 à 296	l'utilisateur doit être rédigé vers la page de confirmation.html	OK
12	confirmation.html	Une fois que l'utilisateur à valider son panier	confirmation.js : fonction commande: ligne de 4 à 35	Un id de commande et total payé doit apparaître	OK
13	confirmation.html	Une requête post doit s'afficher dans la session Storage avec un order	panier.js: ligne 283 à 300	Dans la session Storage nous devons avoir un tableau avec la récupération des données du formulaire ainsi le récapitulatif de la commande	OK
14	confirmation.html	Après avoir passé une commande, l'utilisateur clic sur le "retour au site " le panier doit être vide ainsi que dans le local storage et session Storage	confirmation.js: fonction retour() ligne 38 à 42	Vérifier que le panier est vide ainsi que le local storage et session Storage	OK
15	index.html	Si le serveur est déconnecté	index.js ligne 20 à 31	Une boîte de dialogue doit s'afficher alert "Nous sommes désolés, le serveur ne répond pas ! "	OK