

	Functionalité	Action	Result expected	Result observed
1	Une page d'accueil montrant (de manière dynamique) tous les articles disponibles à la vente.	Ouvrir sur la page d'accueil du site web dans un navigateur	Affichage de l'ensemble des produits	OK / Error
2	fetchMeuble	Récupération des données avec l'API	Retour du fichier res.json	Promise / Error
3	meubleDisplay	Récupération de la promesse de la fonctionnalité fetchMeuble permettant d'ajouter l'id, l'image, le nom et la description du produit pour chaque produit dans le DOM	Ajout de l'id, l'image, le nom et la description de chaque produit	OK / Error
4	button	Exécute la fonction button en boucle à chaque clique	Redirige vers la page product.html	OK / Error
5	Une page "produit" qui affiche (de manière dynamique) les détails du produit	Cliquer sur une carte d'un produit depuis la page d'accueil	Affichage les détails du produit	OK / Error
6	getProduct	Requête auprès de l'API puis ajout de l'id à la fin de l'URL	Retour du fichier a.json	OK / Error
7	renderProduct	Récupération de la promesse getProduct et intégration dans le HTML	Affichage les détails du produit dans le HTML	OK / Error
8	Depuis cette page "produit", l'utilisateur peut sélectionner une quantité, une couleur	Définir un nombre dans l'input et une couleur dans le menu déroulant	Implémenter ces options dans le localStorage	OK / Message d'erreur
9	productQuantity / productColor	Récupération des options via le ElementById et identification des alertes	Affichage d'alertes si une ou des option(s) ne sont pas sélectionnées	OK / Message d'erreur
10	ajouter le produit à son panier.	Définir les options et cliquer sur le bouton "ajouter au panier"	Ajouter le produits et ces options dans le localStorage	OK / Message d'erreur
11	addToCart	Récupération des options du produit dans le HTML puis ajout au panier au clique du btn_addToCart	Définir ses options puis ajouter au panier	OK / Error

12	btn_sendInCart	Récupération de l'id #addToCart dans le HTML puis ajout d'un event au clique	Ajouter le produit au panier au clique si toutes les conditions sont valides	OK / Message d'erreur
13	confirmation	créer une fenêtre de confirmation avec un message défini	Affiche "L'article a été envoyé dans le panier" quand le produit est ajouté au panier	OK / Message d'erreur
14	<i>Un résumé des produits dans le panier, le prix total</i>	<i>Afficher les différents produits et multiplier le prix des produit selon les quantités puis additionner avec les différents modèles</i>	<i>Affichage des produits ajoutés au panier et calcule du prix total</i>	<i>OK / Error</i>
15	displayCart	Récupération des données de l'API et intégration de ces données dans le HTML	Affichage des produits ajoutés au panier avec leurs nom, description, image et options	OK / Error
16	totalPrice	Récupération de la valeur price dans le localStorage, += le prix du produit x la quantité	À l'ajout d'un produit au panier, le prix total est calculé	OK / Error
17	totalQuantity	Récupération de la valeur quantity dans le localStorage	input servant à ajouter la quantité des produits ajoutés dans le panier	OK / Error
18	<i>la possibilité de modifier la quantité d'un produit sélectionné ou bien de supprimer celui-ci.</i>	<i>Possibilité de suppression ou de changer la quantité de chaque produits dans le localStorage</i>	<i>Modification des données dans le localStorage</i>	<i>OK / Error</i>
19	valueNewQuantity	Condition if si l'input est remplis avec un nombre inférieur à 1 / Condition else si l'input est remplis avec un nombre supérieur à 100	Afficher une alerte en fonction de la conditions	OK / Message d'erreur
20	removeFromCart	Récupération dans le localStorage et ajout d'un event au clique qui delete le produit du panier et qui reload la page	Produit supprimé du panier au clique sur le bouton supprimé	OK / Error
21	<i>Un formulaire permettant de passer une commande. Les données du formulaire doivent être correctes et bien formatées avant d'être renvoyées au back-end.</i>	<i>Mettre des vérification sur chaque input pour vérifier que l'user remplis correctement les données demandées</i>	<i>Avoir des données d'envoi correctes</i>	<i>OK / Message d'erreur</i>
22	firstName / Name / ...	Récupération des input dans le HTML via des getElementById	Attribuer les vérification sur les input	OK / Error

23	firstNameErrorMsg / NameErrorMsg / ...	Afficher un message si l'input est mal remplis	Affichage d'un message personnalisé sur l'input où l'erreur à été faite	OK / Message d'erreur
24	btn_sendOrder	Ajout d'un event avec une condition if qui récupère tous les input remplis, qui renvoie à la page de confirmation, qui nettoie le localStorage et else qui envoi une alerte au clique sur le bouton "Commander !"	if = Redirection vers la page de confirmation else = alerte que le formulaire a mal été remplis	OK / Message d'erreur
25	<i>Un message de confirmation de commande, remerciant l'utilisateur pour sa commande, et indiquant l'identifiant de commande envoyé par l'API.</i>	<i>Ajout d'un message avec un id comme identifiant personnalisé envoyé dans le HTML</i>	<i>Affichage d'un message de confirmation de commande avec un identifiant</i>	<i>Message / Error</i>
26	confirmOrder	Récupération de l'id du produit puis envoi de l'id dans HTML	Affichage de l'id sur le message de confirmation de commande	Message / Error