ORITEDDY’S TEST PLAN

Test plan objective :

Our test plan will allow us to set up the unit tests necessary to verify the proper functioning of our « Oriteddy » application.

Test environment :

Unit tests will be made with the JavaScript tests framework : Jest.
All unit tests will be stored in a separate folder named « tests » at the root of the project.

UNIT TESTS

Test N°1 : Check Fetch API

Description : Check the fetchAPI function get a positive response from the API.

Parameters : url + endpoint ‘/teddies’, method, body

Pass if returns : A 201 response contains list of products.

Test N° 2 : Check Fetch API error

Description : Check the fetchAPI function get an error in the response from the API.

Parameters : url, method, body

Pass if returns : 400, 404 or 500 http error code.

Test N°3 : Check TeddyBear's ID is saved

Description : Check new TeddyBear instance saves a new ID.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear with the new ID.

Test N°4 : Check TeddyBear's image is saved

Description : Check new TeddyBear instance saves a new image.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear with the new image.

Test N°5 : Check TeddyBear's name is saved

Description : Check new TeddyBear instance saves a new name.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear with the new name.

Test N°6 : Check TeddyBear's description is saved

Description : Check new TeddyBear instance saves a new description.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear with the new description.

Test N°7 : Check TeddyBear’s array of colors is saved

Description : Check new TeddyBear instance saves a new array of colors.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear with the new array of colors.

Test N°8 : Check TeddyBear’s default color does not exist

Description : Check new TeddyBear instance don’t have a default color.

Parameters : new TeddyBear(id, image, name, description, colors = [], price, quantity)

Pass if returns : teddyBear don’t have color saved.

Test N°9 : Check TeddyBear’s price is saved

Description : Check new TeddyBear instance saves a new price.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear with the new price.

Test N°10 : Check TeddyBear has default quantity

Description : Check new TeddyBear instance has a default quantity.

Parameters : new TeddyBear(id, image, name, description, colors, price, quantity)

Pass if returns : teddyBear has a default quantity = 0.

Test N°11 : Check TeddyBear can set a new ID

Description : Check new TeddyBear instance can set a new ID.

Parameters : teddyBear.Id = « new Id ».

Pass if returns : teddyBear has the new ID.

Test N°12 : Check TeddyBear can set a new image

Description : Check new TeddyBear instance can set a new image.

Parameters : teddyBear.Image = « new Image ».

Pass if returns : teddyBear has the new image.

Test N°13 : Check TeddyBear can set a new name

Description : Check new TeddyBear instance can set a new name.

Parameters : teddyBear.Name = « new Name ».

Pass if returns : teddyBear has the new name.

Test N°14 : Check TeddyBear can set a new description

Description : Check new TeddyBear instance can set a new description.

Parameters : teddyBear.Description = « new Description ».

Pass if returns : teddyBear has the new description.

Test N°15 : Check TeddyBear can set a new color

Description : Check new TeddyBear instance can set a new color.

Parameters : teddyBear.ChosenColor = « new Color ».

Pass if returns : teddyBear has the new color.

Test N°16 : Check TeddyBear can set a new price

Description : Check new TeddyBear instance can set a new price.

Parameters : teddyBear.Price = « new Price ».

Pass if returns : teddyBear has the new price.

Test N°17 : Check TeddyBear can set a new quantity

Description : Check new TeddyBear instance can set a new quantity.

Parameters : teddyBear.Quantity = « new Quantity ».

Pass if returns : teddyBear has the new quantity.

Test N°18 : Check Append Items list in Dropdown Menu

Description : Check the appendItemInViewDropdownMenu returns items list in dropdown menu

Parameters : item, url

Pass if returns : items list in dropdown menu.

Test N° 19 : Check Append items in index page

Description : Check the appendItemInViewIndex function displays items in index page.

Parameters : item

Pass if returns : items number in index page > 0.

Test N°20 : Check Get Url Parameter

Description : Check the getUrlParam function returns url parameter contains product's ID

Parameters : url parameter name

Pass if returns : product's ID from Url Parameter

Test N°21 : Check Fetch API to get product's ID

Description : Check the fetchAPI function get a positive response from the API contains product's information.

Parameters : url + endpoint '/ID', method, body

Pass if returns : A JSON response contains product's information in relation to its ID.

Tests N°22 : Check Append item's information to product page

Description : Check the appendItemInfosInViewProduct function displays item's information in the product page.

Parameters : product

Pass if returns : list of item's information in product page.

Test N°23 : Check Cart sent to localStorage

Description : Check the sendCartToLocalStorage function sends array containing products to localStorage when user click on « Add to cart » button.

Parameters : object, localStorage.key, localStorage.value

Pass if returns : localStorage.key and localStorage.value != empty.

Test N°24 : Check success message

Description : Check the successMessage function adding success message in div container

Parameters : messageContainer, message

Pass if returns : true.

Test N°25 : Check appear success message on cart page

Description : Check the appearSuccessMessageAddToCart function adding to div « item-confirm » a confirmation message.

Parameters : messageContainer

Pass if returns : true.

Test N°26 : Check number of items in cart

Description : Check the numberOfItemsInCart function displaying number of products in cart on item menu « cart ».

Parameters : localStorageKey.length

Pass if returns : number in item menu « cart » == number of products in localStorage « cart ».

Test N°27 : Check get Cart from LocalStorage

Description : Check the getCartFromLocalStorage function get the content of localStorage and applys a new TeddyBear class forEach product.

Parameters : localStorage.key, TeddyBear's Class.

Pass if returns : array of TeddyBears.

Test N°28 : Check Append cart content in cart page

Description : Check the appendProductsInViewCart function displays list of products contained in the cart.

Parameters : array of TeddyBears.

Pass if returns : products list contained in cart to the cart page.

Test N°29 : Check addition item's total cost

Description : Check the appendCartTotalCostInViewCart function additioning cost of each product and display the total cost in the cart page.

Parameters : itemsTotalPrice, itemTotalPriceNumber

Pass if returns : itemsTotalPrice > 0.

Test N°30 : Check delete product on cart

Description : Check the deleteItemOnCart function deleting a product in the cart page.

Parameters : array of TeddyBears.

Pass if returns : true.

Test N°31 : Check form is valid

Description : Check the postOrderToAPI function continues when the form is valid.

Parameters : object Contact

Pass if returns : the posrtOrderToAPI returns a 201 response.

Test N°32 : Check firstName in form is invalid

Description : Check the postOrderToAPI function displaying an error message when the firstName's input is invalid.

Parameters : firstName's input, nameRegex.

Pass if returns : an error message when the firstName's input is invalid.

Test N°33 : Check lastName in form is invalid

Description : Check the postOrderToAPI function displaying an error message when the lastName's input is invalid.

Parameters : lastName's input, nameRegex.

Pass if returns : an error message when the lastName's input is invalid.

Test N°34 : Check address in form is invalid

Description : Check the postOrderToAPI function displaying an error message when the address input is invalid.

Parameters : address input, addressRegex.

Pass if returns : an error message when the address input is invalid.

Test N°35 : Check city in form is invalid

Description : Check the postOrderToAPI function displaying an error message when the city input is invalid.

Parameters : city input, cityRegex.

Pass if returns : an error message when the city input is invalid.

Test N°36 : Check email in form is invalid

Description : Check the postOrderToAPI function displaying an error message when the email input is invalid.

Parameters : email input, emailRegex.

Pass if returns : an error message when the email input is invalid.

Test N°37 : Check post order to API

Description : Check the postOrderToAPI function sending the order correctly to the server.

Parameters : localStorage, order object

Pass if returns : a 201 response of the API containing products information and orderID.

Test N° 38 : Check orderId and totalCost appear

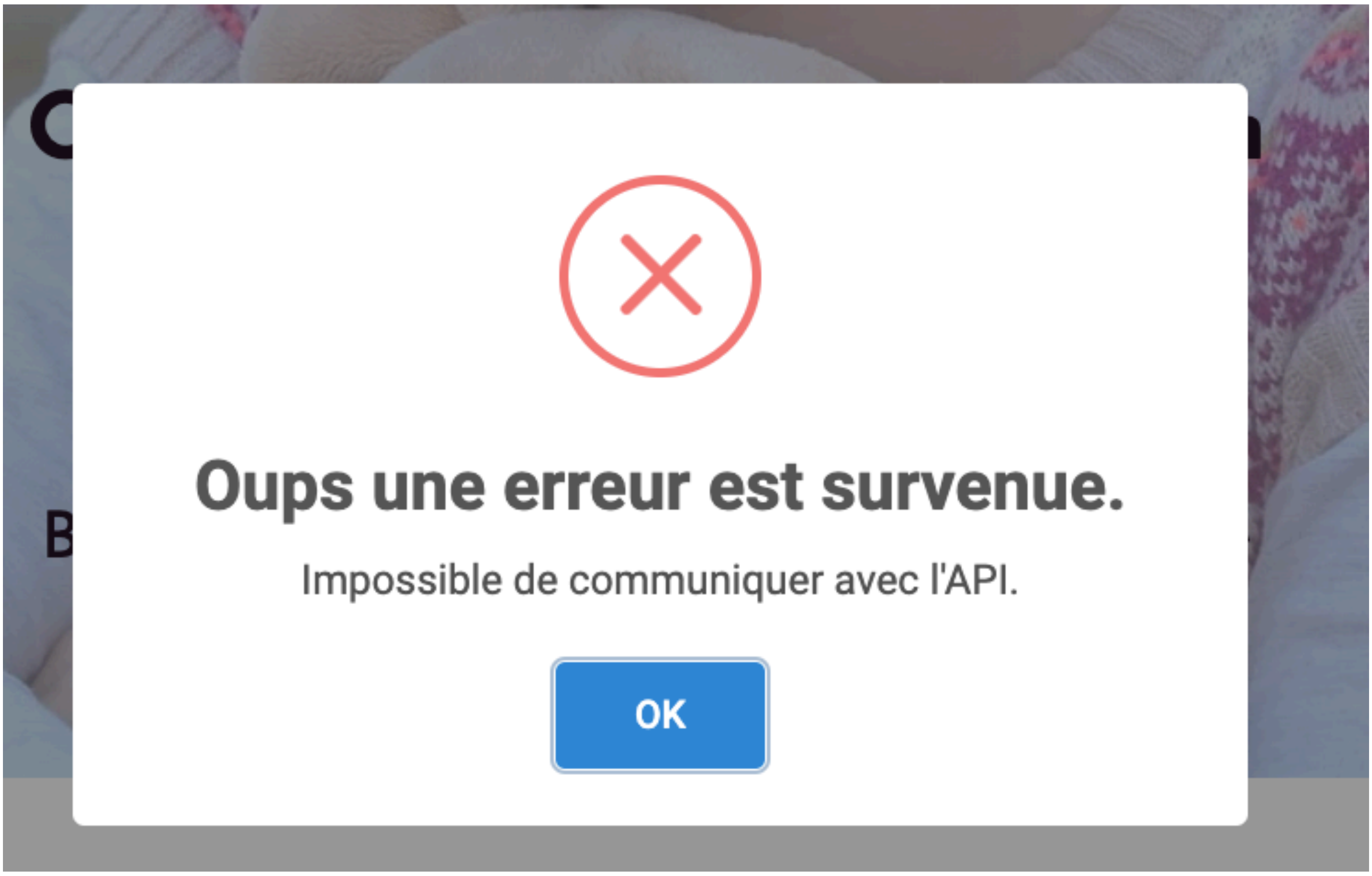
Description : Check the confirmedOrder function displaying the order id and total cost in the order page.

Parameters : orderId, totalCost

Pass if returns : true.

INTEGRATION TESTS

Test N° 1 : Check error message when bad communication with API

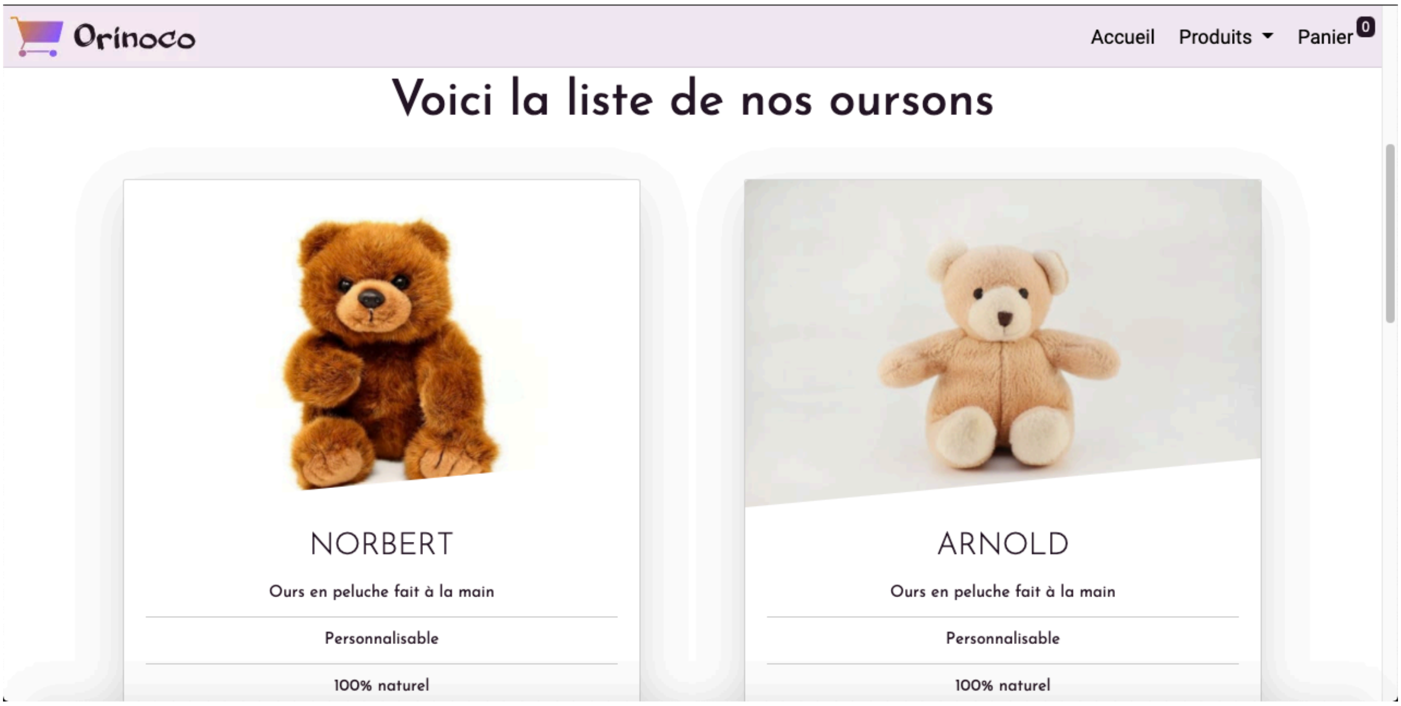


Oups une erreur est survenue.

Impossible de communiquer avec l'API.

Check if an error message appears when the project have a bad communication with the API.

Test N° 2 : Check list of products when positive response of API



Orinoco

Accueil Produits Panier 0

Voici la liste de nos oursons

NORBERT

Ours en peluche fait à la main

Personnalisable

100% naturel

ARNOLD

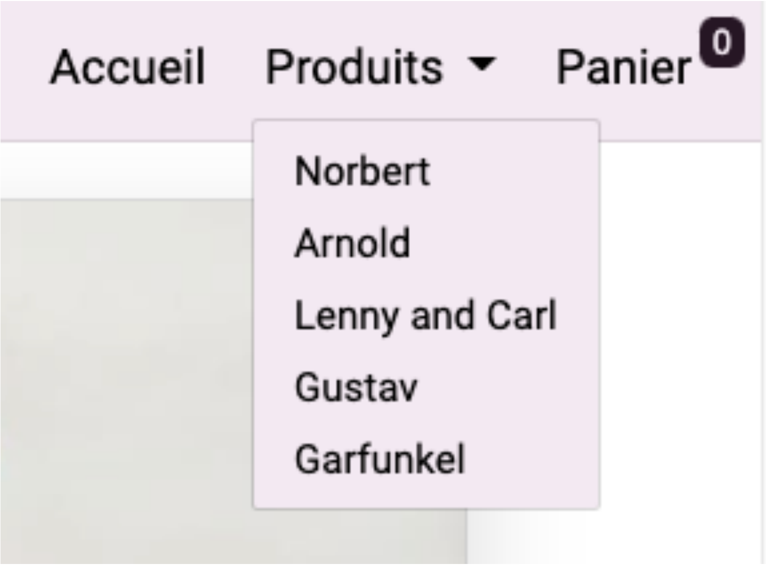
Ours en peluche fait à la main

Personnalisable

100% naturel

Check if a list of the products appears when the API returns a positive response.

Test N°3 : Check list of products in menu



Accueil Produits Panier 0

Norbert

Arnold

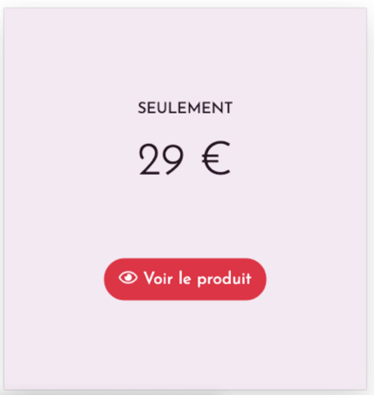
Lenny and Carl

Gustav

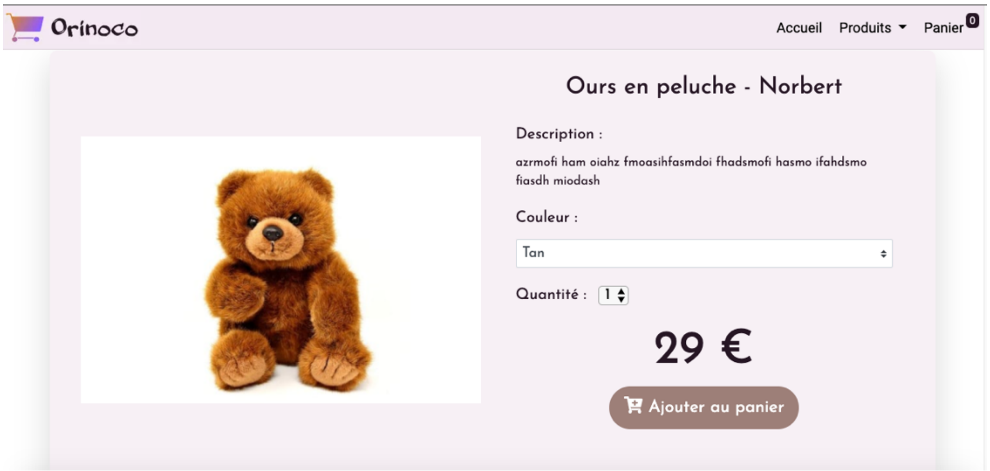
Garfunkel

Check if a list of the products appears in dropdown menu « Produits ».

Test N° 4 : Check focused product appears in product page


SEULEMENT

29 €



Ours en peluche - Norbert

Description :

azrmofi ham oiahz fmoasihfasmdoi fhadsmofi hasmo ifahdsmo fiasdh miodash

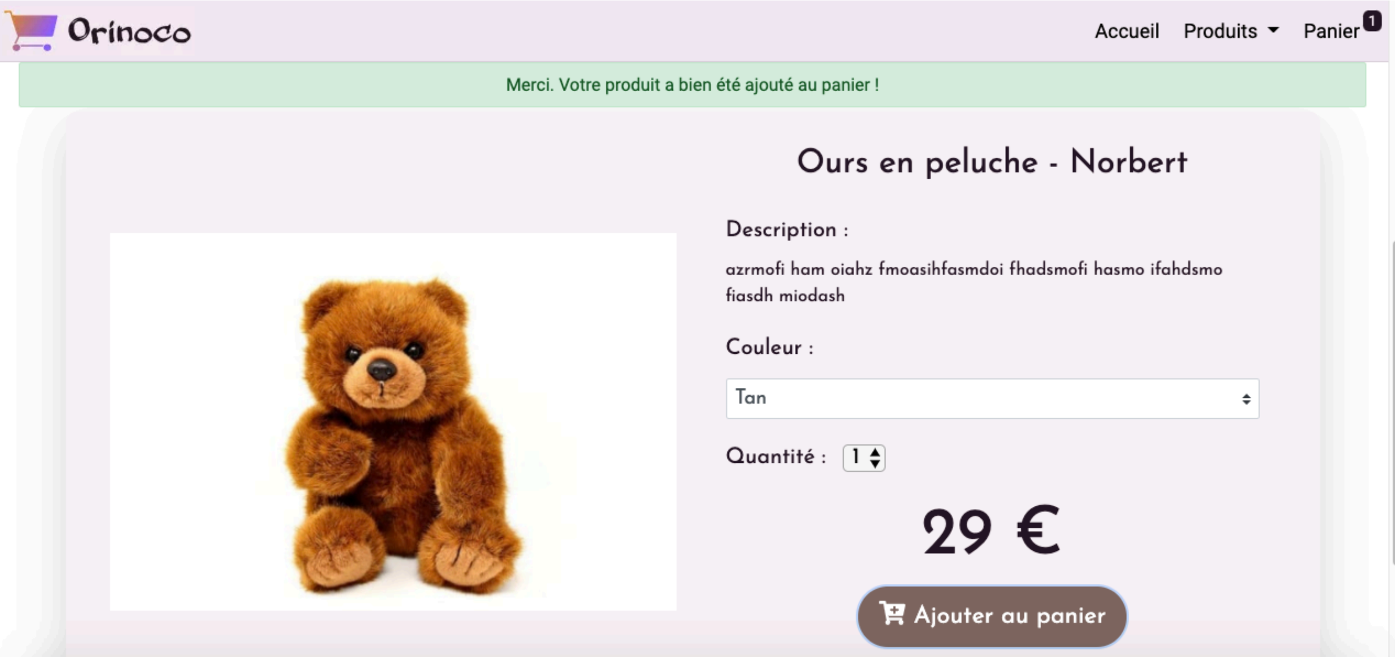
Couleur : Tan

Quantité : 1

29 €

Check if the product clicked on the index page is displayed on the product page.

Test N° 5 : Check success message appears when product is added to cart



Merci. Votre produit a bien été ajouté au panier !

Ours en peluche - Norbert

Description :

azrmofi ham oiahz fmoasihfasmdoi fhadsmofi hasmo ifahdsmo fiasdh miodash

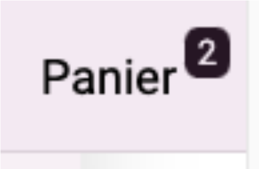
Couleur : Tan

Quantité : 1

29 €

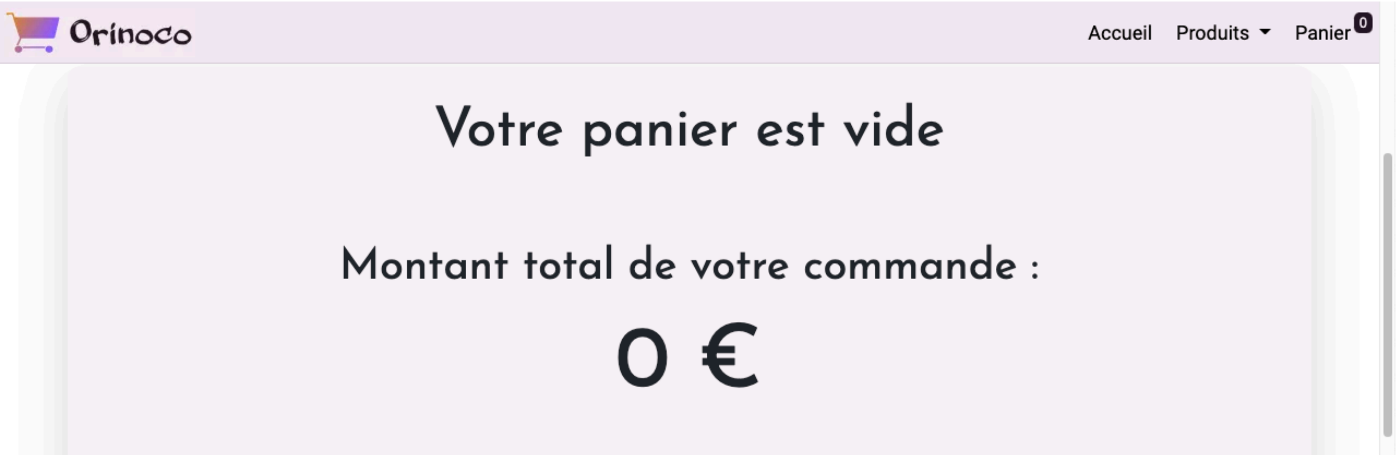
Check if a success message appears on the top of the screen when a product is added to the cart.

Test N° 6 : Check number of items is displayed



Check if the number of products in the cart is displayed next to the « Panier » menu.

Test N° 7 : Check display when the cart is empty



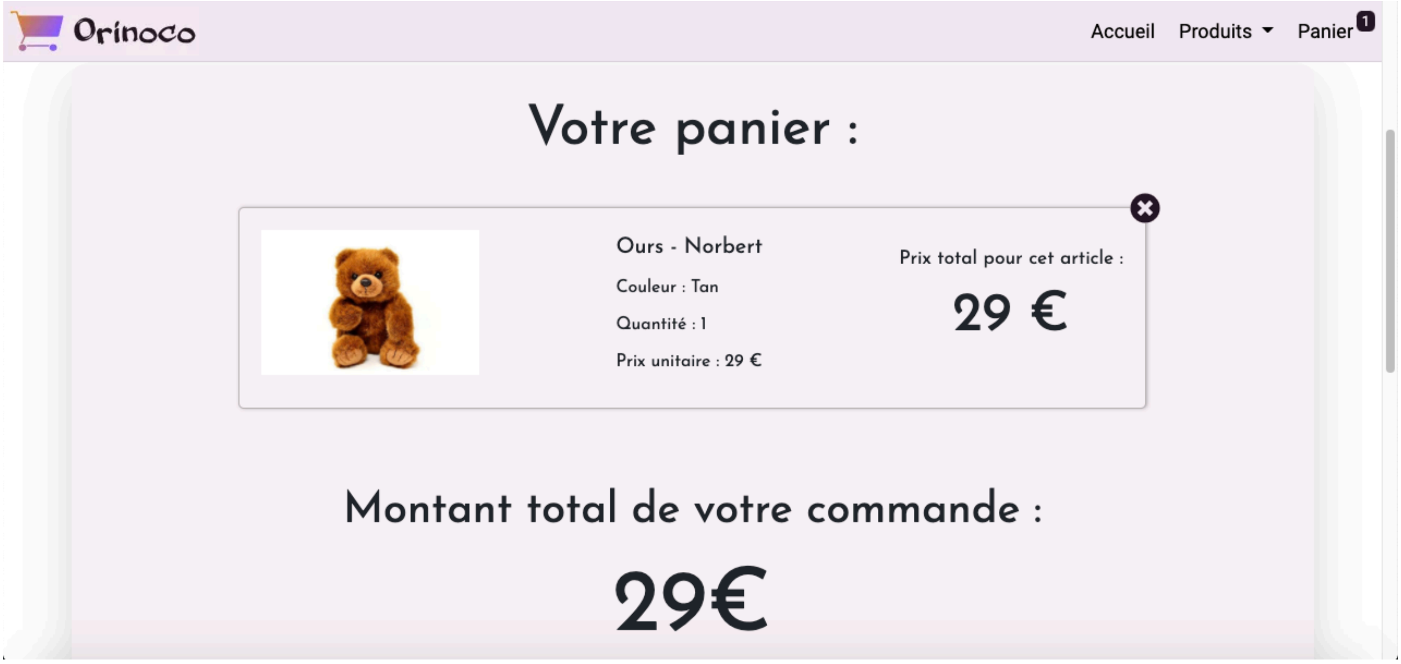
Votre panier est vide

Montant total de votre commande :

0 €

Check if a message appears on the cart page when the cart is empty.

Test N° 8 : Check display when the cart is not empty



Votre panier :

Ours - Norbert

Couleur : Tan

Quantité : 1

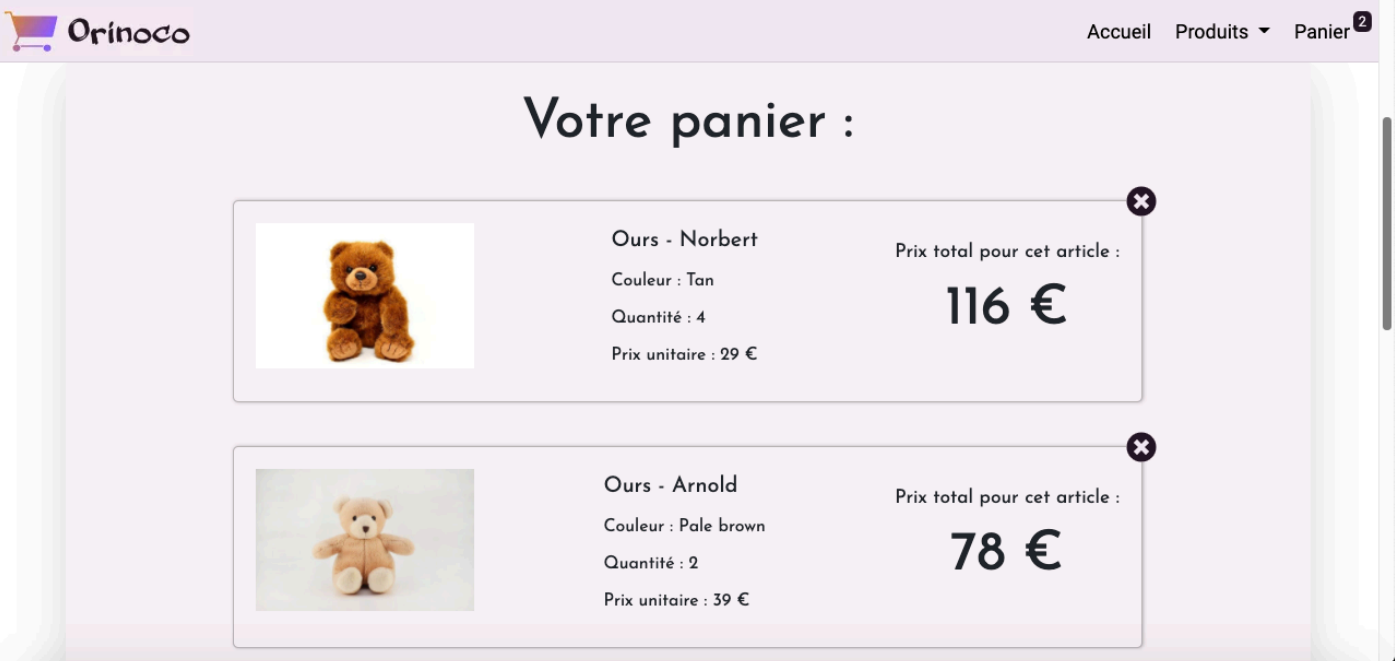
Prix unitaire : 29 €

Prix total pour cet article :

29 €

Montant total de votre commande :

29€



Votre panier :

Ours - Norbert

Couleur : Tan

Quantité : 4

Prix unitaire : 29 €

Prix total pour cet article :

116 €

Ours - Arnold

Couleur : Pale brown

Quantité : 2

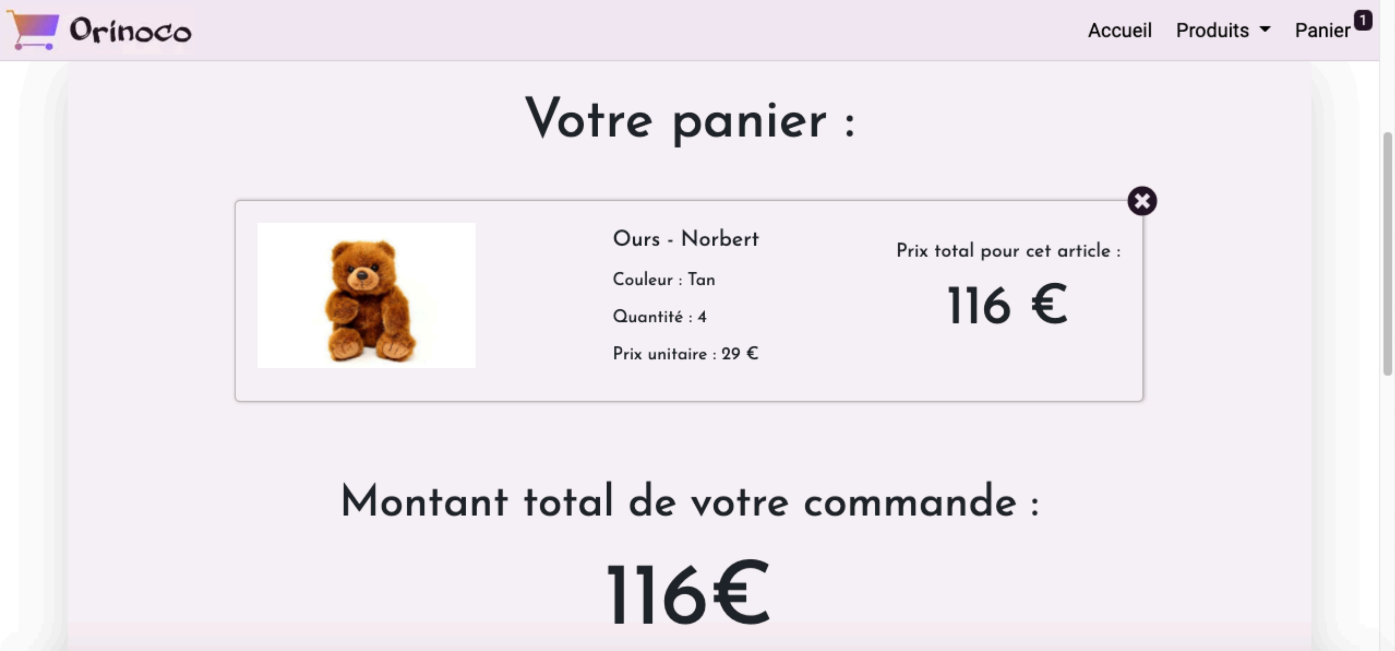
Prix unitaire : 39 €

Prix total pour cet article :

78 €

Check if the list of products appears on the cart page when the cart is not empty.

Test N° 9 : Check the fusion of the quantities of a product



Votre panier :

Ours - Norbert

Couleur : Tan

Quantité : 4

Prix unitaire : 29 €

Prix total pour cet article :

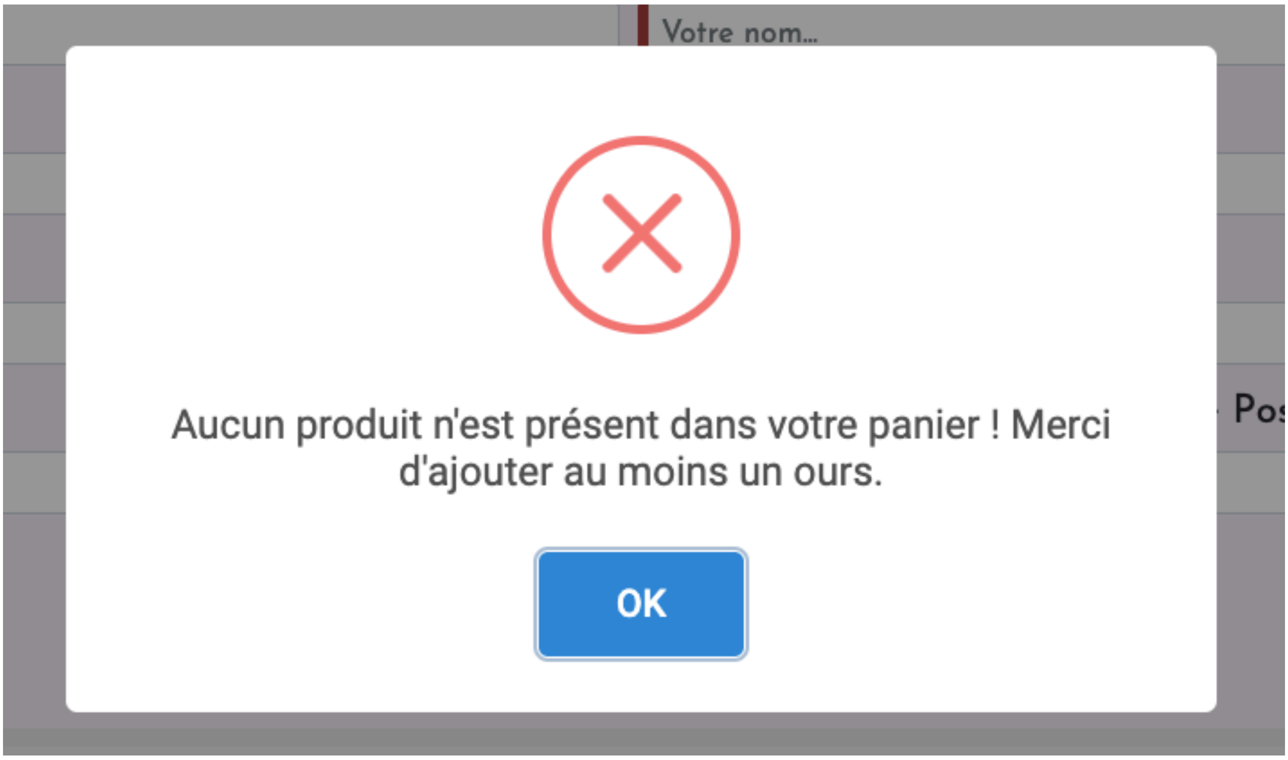
116 €

Montant total de votre commande :

116€

Check if the quantities of identic products added to the cart merge.

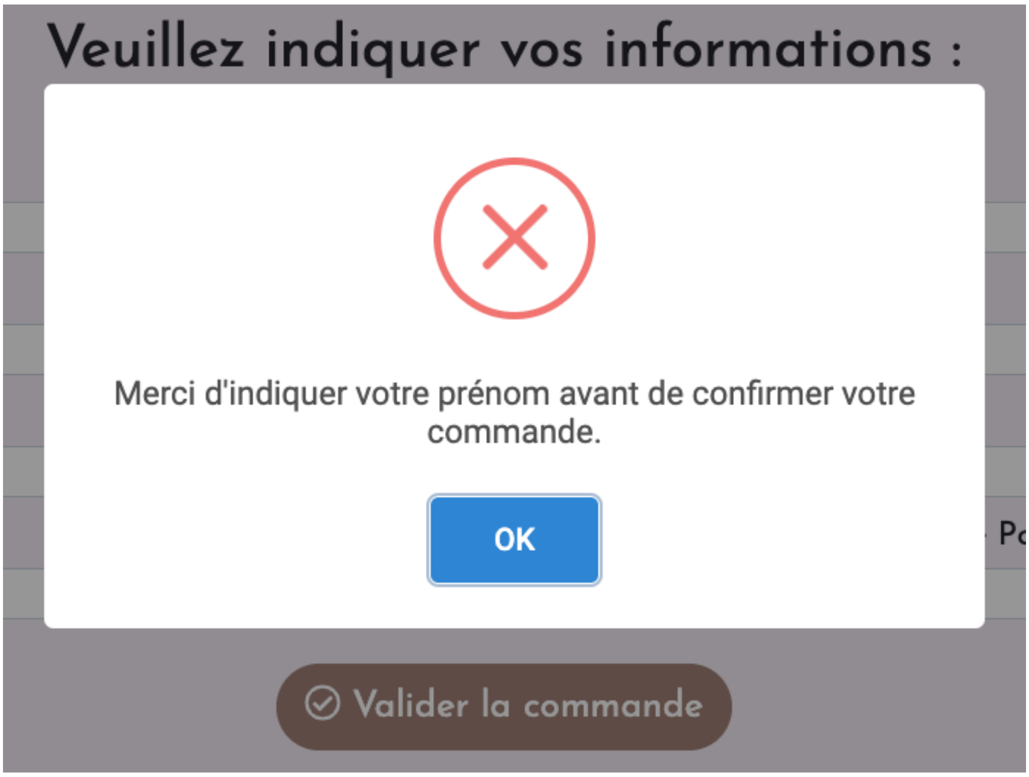
Test N° 10 : Check error message when try confirmation without product


Votre nom...

Aucun produit n'est présent dans votre panier ! Merci d'ajouter au moins un ours.

Check if an error message appears when the user want to confirm order without product in the cart.

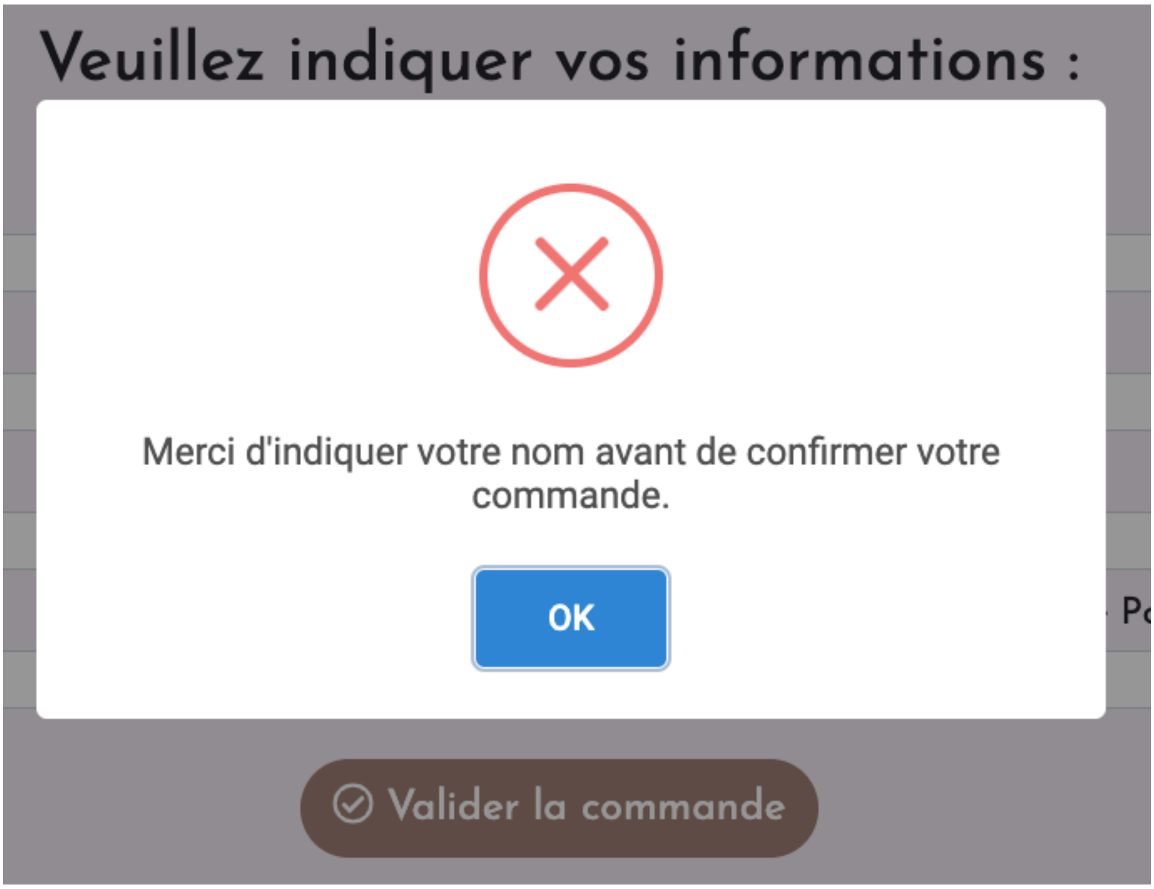
Test N° 11 : Check error message when the firstname is missing or wrong


Veuillez indiquer vos informations :

Merci d'indiquer votre prénom avant de confirmer votre commande.

Check if an error message appears when the firstname is missing or wrong in the form while the user want to confirm the order.

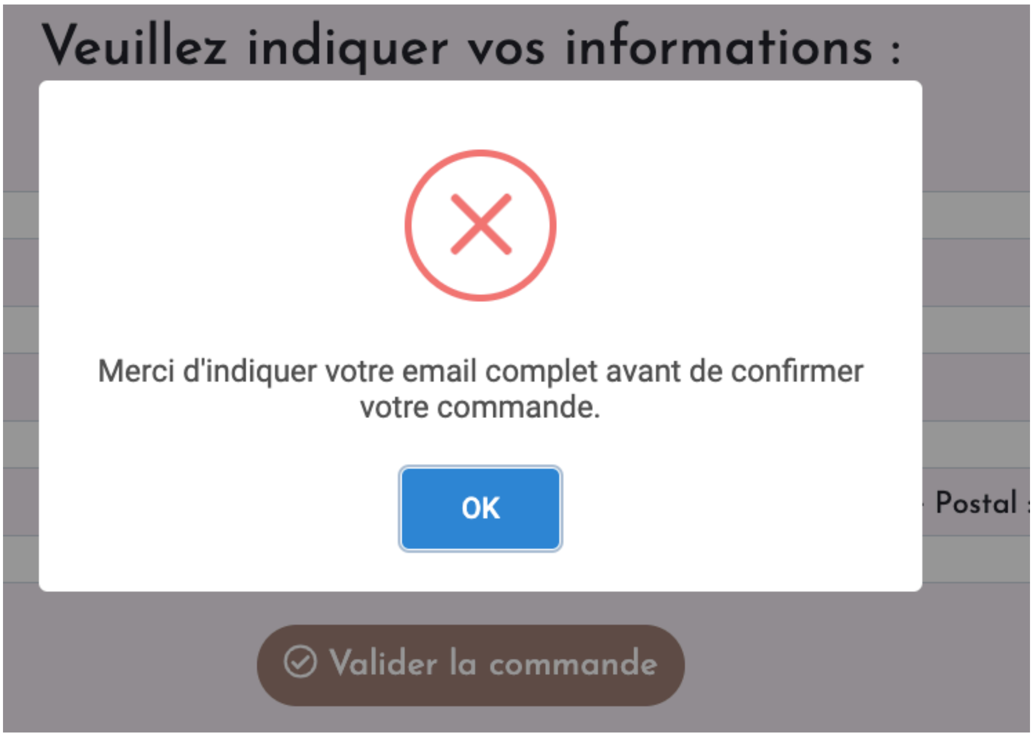
Test N° 12 : Check error message when the lastname is missing or wrong


Veuillez indiquer vos informations :

Merci d'indiquer votre nom avant de confirmer votre commande.

Check if an error message appears when the lastname is missing or wrong in the form while the user want to confirm the order.

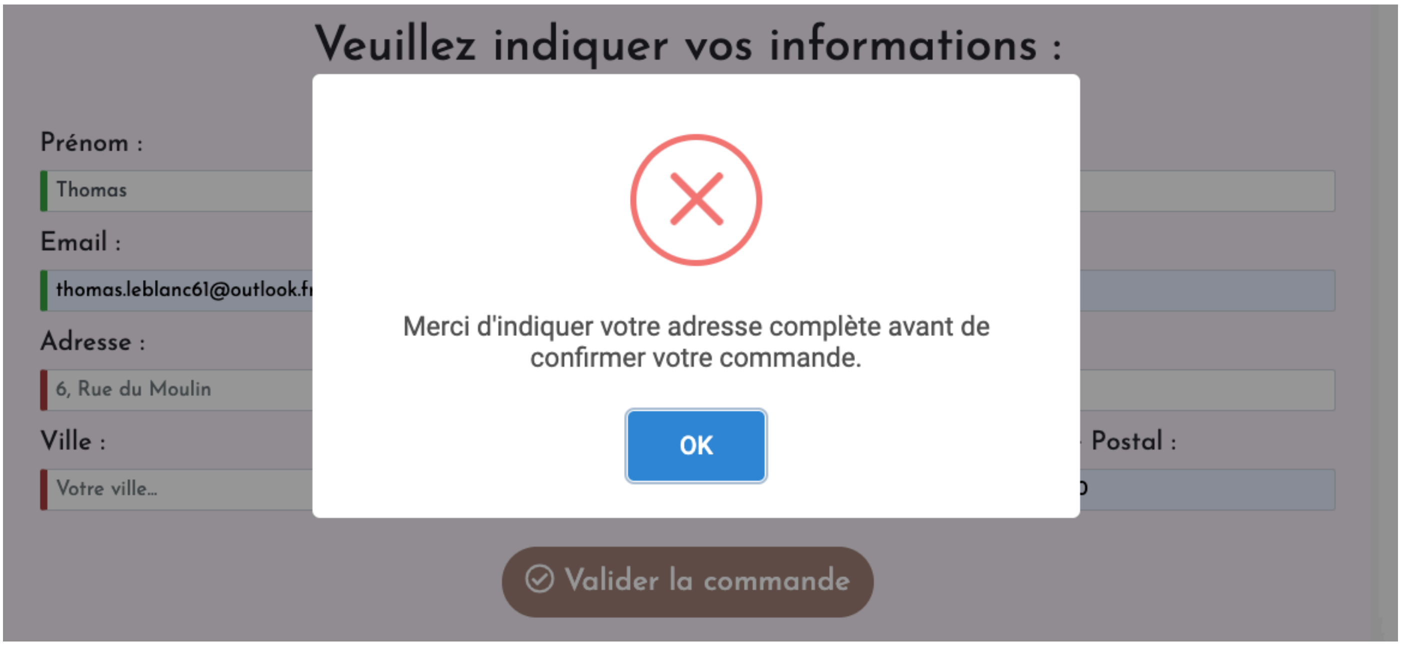
Test N° 13 : Check error message when the email is missing or wrong


Veuillez indiquer vos informations :

Merci d'indiquer votre email complet avant de confirmer votre commande.

Check if an error message appears when the email is missing or wrong in the form while the user want to confirm the order.

Test N° 14 : Check error message when the address is missing or wrong



Veuillez indiquer vos informations :

Prénom : Thomas

Email : thomas.leblanc61@outlook.f

Adresse : 6, Rue du Moulin

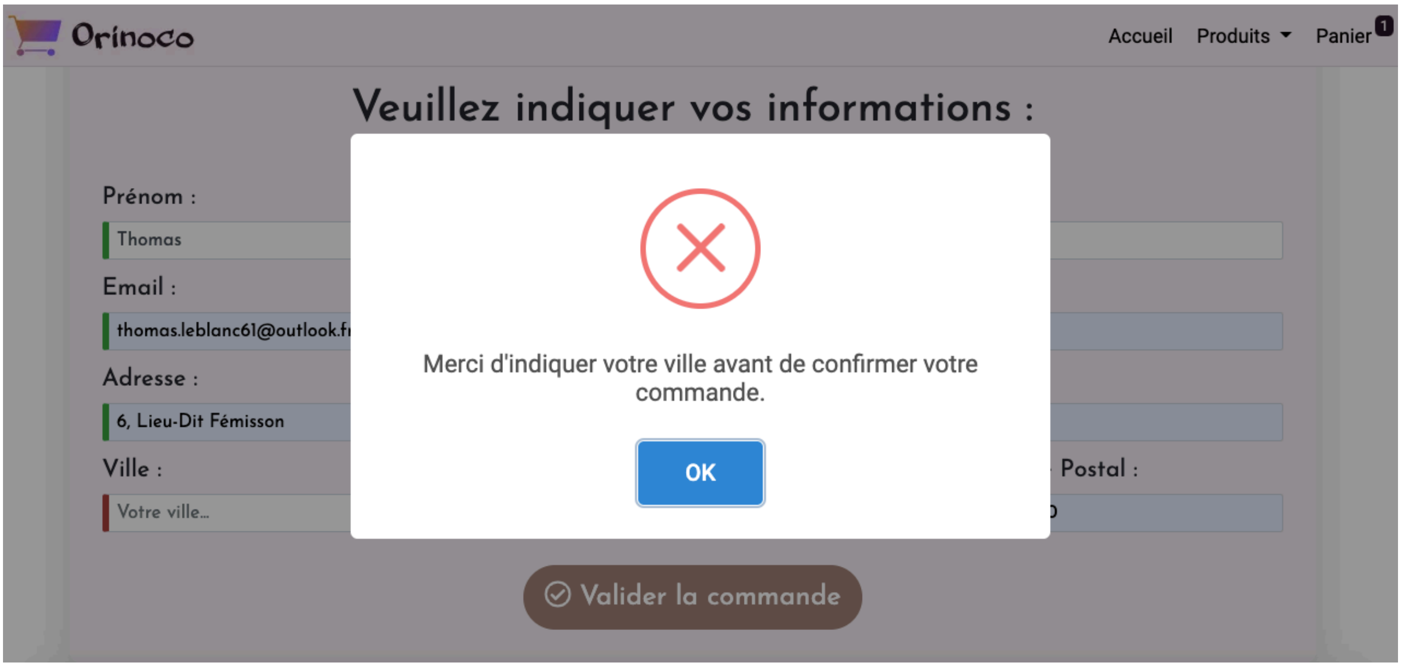
Ville : Votre ville...

Postal :

Merci d'indiquer votre adresse complète avant de confirmer votre commande.

Check if an error message appears when the address is missing or wrong in the form while the user want to confirm the order.

Test N° 15 : Check error message when the city is missing or wrong



Veuillez indiquer vos informations :

Prénom : Thomas

Email : thomas.leblanc61@outlook.f

Adresse : 6, Lieu-Dit Fémisson

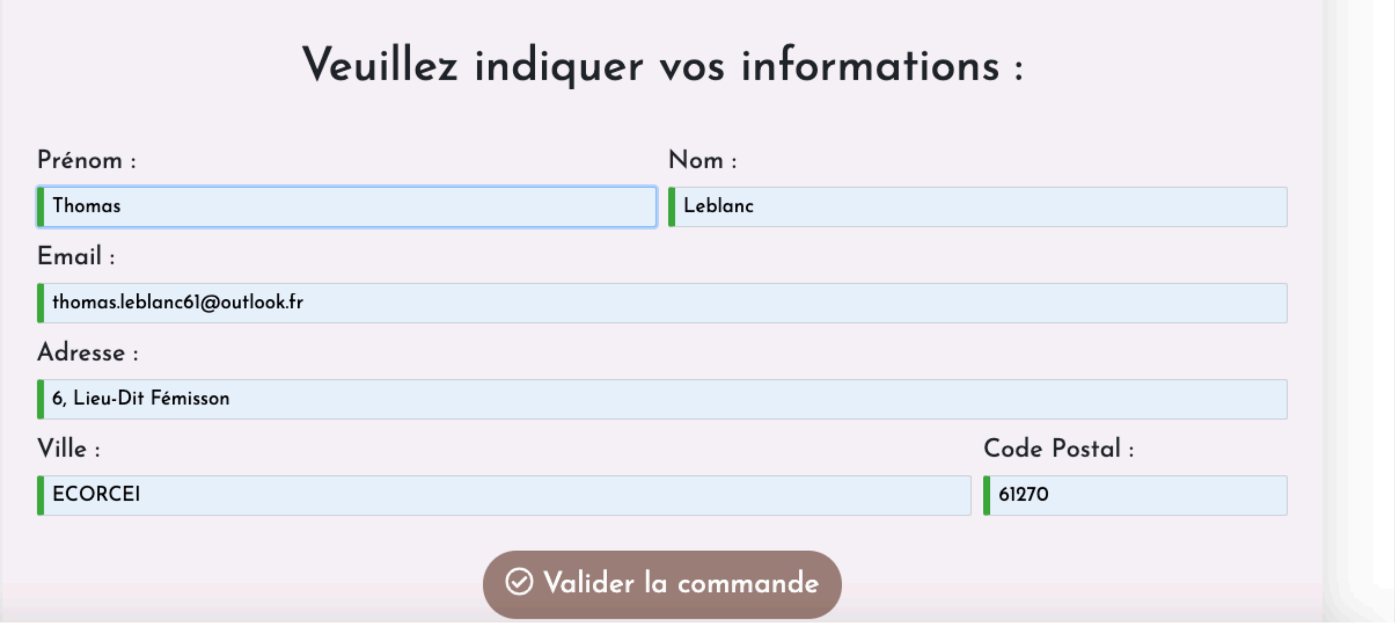
Ville : Votre ville...

Postal :

Merci d'indiquer votre ville avant de confirmer votre commande.

Check if an error message appears when the city is missing or wrong in the form while the user want to confirm the order.

Test N° 16 : Check if the form is correctly filled



Veuillez indiquer vos informations :

Prénom : Thomas

Nom : Leblanc

Email : thomas.leblanc61@outlook.fr

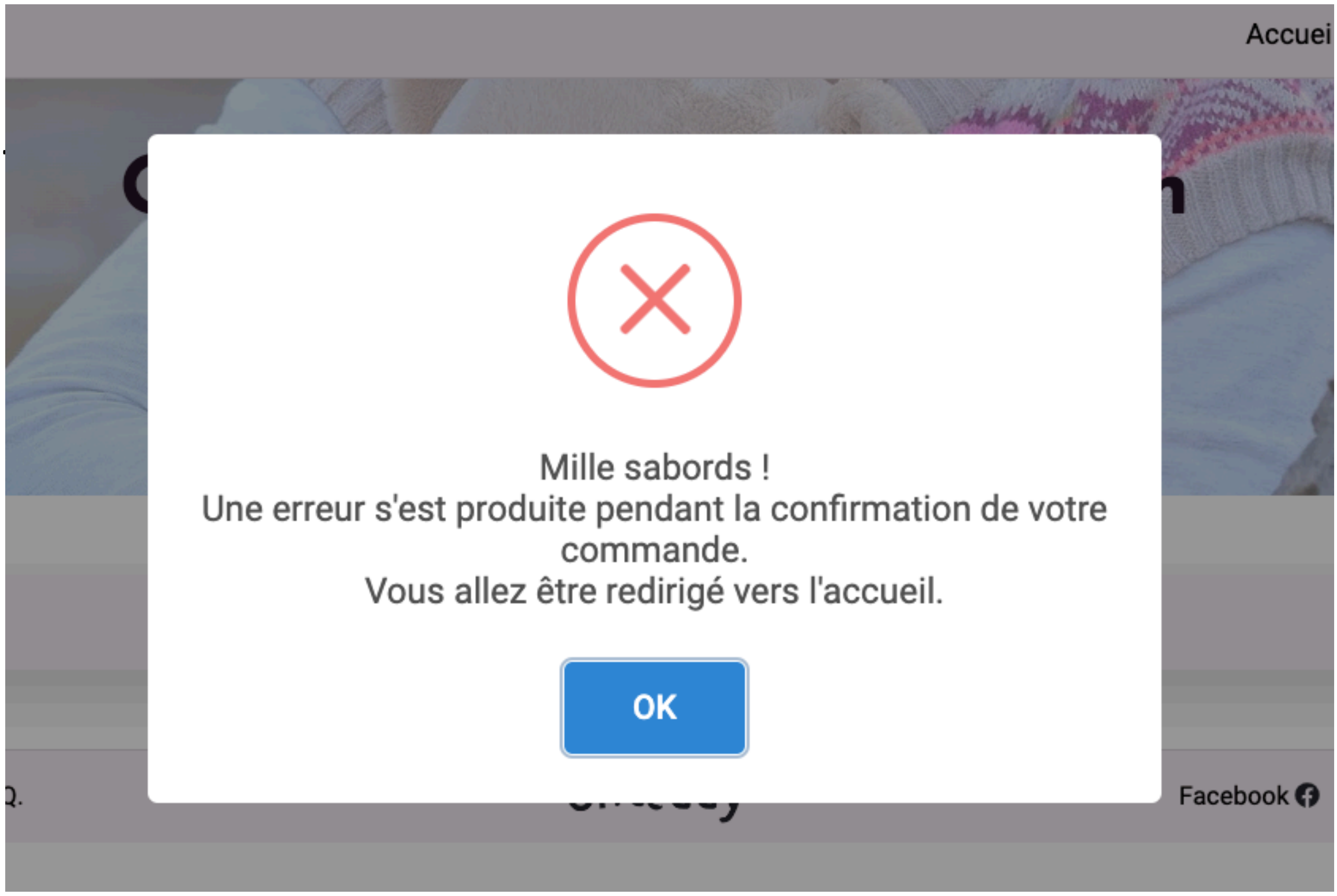
Adresse : 6, Lieu-Dit Fémisson

Ville : ECORCEI

Code Postal : 61270

Check if the form is correctly filled with the inputs who are displayed in green color.

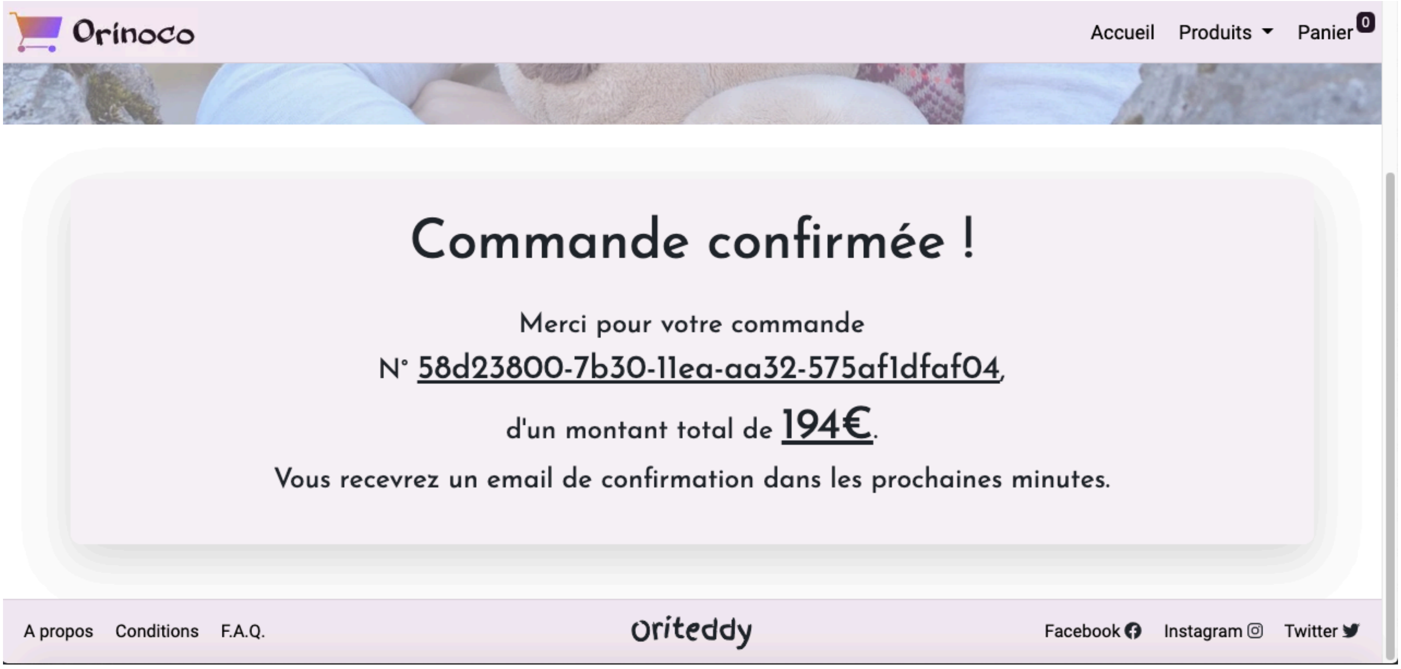
Test N° 17 : Check error message if an error occured when the order confirmation is not confirmed



Mille sabords !
Une erreur s'est produite pendant la confirmation de votre commande.
Vous allez être redirigé vers l'accueil.

Check if an error message appears when the order confirmation is not confirmed.

Test N° 18 : Check confirmed message of the order confirmation



Commande confirmée !

Merci pour votre commande
N° 58d23800-7b30-11ea-aa32-575af1dfaf04,
d'un montant total de 194€.
Vous recevrez un email de confirmation dans les prochaines minutes.

Check if the user is redirected to the order page and a confirmed message appears with the orderId and the total cost of the order.

<u>END OF TESTS.</u>