

# ¡Javascript!

guayerd beta hub

# Agenda del día



# 01

## Introducción

Repaso Clase 22

# 02

## Cart

localStorage.  
Contador.  
Agregar producto.

# 03

## Ejercitación

Funcionalidad  
contador, agregar  
productos y cart  
widget.

## Cart Widget

Agregar funcionalidad  
en nav.



# daily

¿Cómo venimos?

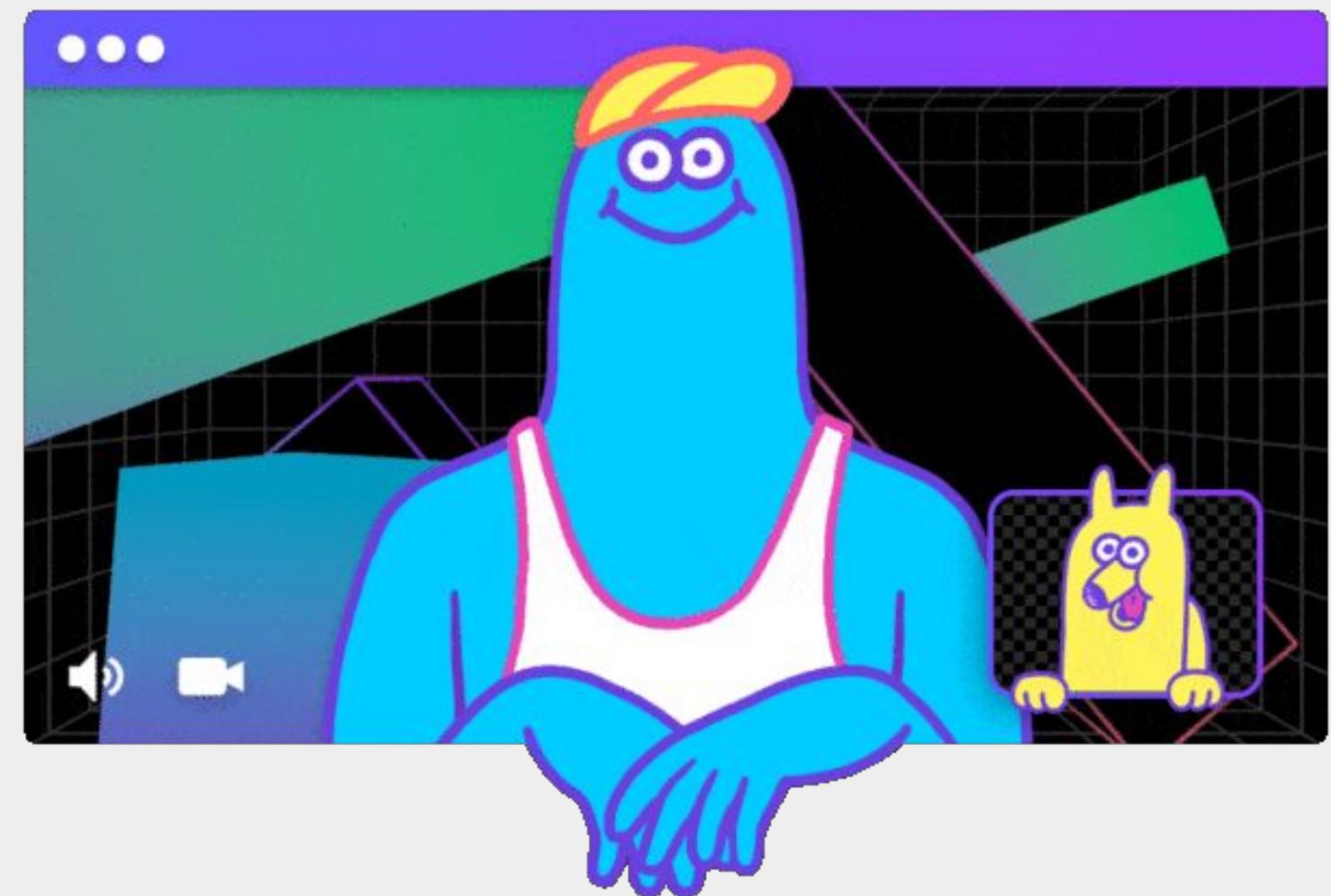
¿Algo nos bloquea?

¿Cómo seguimos?



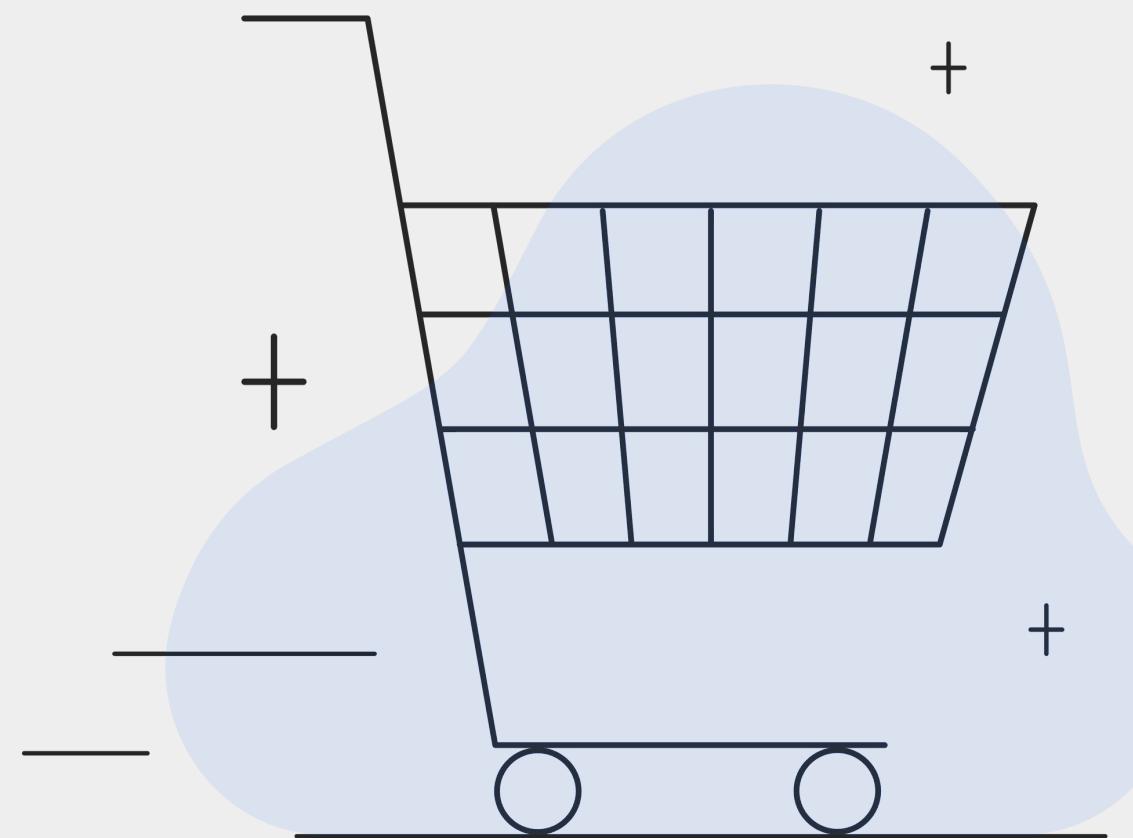
# Cart

Sesión



## SPRINT 3

# Cart



La **idea** es guardar en una variable de almacenamiento local los productos que vamos agregando al carrito.

# Crear ítem cart

En js/login.js, dentro de la función handleSubmitLogin, si el usuario inicia sesión agregamos el ítem “cart”. Cómo es un array, lo convertimos en string con JSON.stringify.

```
const handleSubmitLogin = ev => {
    ev.preventDefault()
    let emailForm = ev.target.elements.email.value
    let passwordForm = ev.target.elements.password.value

    if (emailForm === USER_LOGIN.email && passwordForm === USER_LOGIN.password) {
        localStorage.setItem("email", emailForm)
        localStorage.setItem("cart", JSON.stringify([]))
        location.href = "./index.html"
    } else {
        message.innerText = "Por favor introduce credenciales válidas."
    }
}
```

# js/producto.js

Agregamos a los botones eventos y controladores de eventos.



```
<div class="input-group">
  <button class="btn btn-danger" type="button" onclick="increaseItem()">+</button>
  <input type="number" class="form-control" value="0">
  <button class="btn btn-danger" type="button" onclick="decreaseItem()">-</button>
</div><a href="#" class="btn btn-primary col-12" onclick="addItems()">Comprar</a>
```

# js/producto.js

Agregamos las funciones.

```
● ● ●  
const counter = document.querySelector("#producto .input-group input")  
  
function increaseItem() {  
    counter.value = Number(counter.value) + 1  
}  
  
function addItems() {  
    const cart = JSON.parse(localStorage.getItem("cart"))  
  
    const idProduct = Number(window.location.search.split("=")[1])  
  
    cart.push({ id: idProduct, quantity: counter.value })  
  
    localStorage.setItem("cart", JSON.stringify(cart))  
}
```

# Carrito en nav

Avantt Cars [Home](#)

Hola, rick@morty.com |  0 | [Cerrar sesión](#)

# Carrito en nav

Agregamos la imagen y llamamos a un ítem de localStorage que aún no creamos en js/nav.js.



```
<span>|</span><li><b>${localStorage.getItem("quantity")}</b></li><span>|</span>
```

# Creamos el ítem cart al iniciar sesión

```
● ● ●  
const handleSubmitLogin = ev => {  
  ev.preventDefault()  
  let emailForm = ev.target.elements.email.value  
  let passwordForm = ev.target.elements.password.value  
  
  if (emailForm === USER_LOGIN.email && passwordForm === USER_LOGIN.password) {  
    localStorage.setItem("email", emailForm)  
    localStorage.setItem("cart", JSON.stringify([]))  
    localStorage.setItem("quantity", "0")  
    location.href = "./index.html"  
  } else {  
    message.innerText = "Por favor introduce credenciales válidas."  
  }  
}
```

# Al agregar productos modificamos quantity

Primero sumamos todas las cantidades de los productos que quiero comprar en producto.js. Guardo la cantidad en localStorage y actualizo el tag #quantity del nav.js.

```
function addItems() {
    const cart = JSON.parse(localStorage.getItem("cart"))

    const idProduct = Number(window.location.search.split("=")[1])

    cart.push({ id: idProduct, quantity: Number(counter.value) })

    localStorage.setItem("cart", JSON.stringify(cart))

    let quantity = cart.reduce((acumulado, actual) => acumulado + actual.quantity, 0)
    localStorage.setItem("quantity", quantity)
    const quantityTag = document.querySelector("#quantity")

    quantityTag.innerText = quantity
}
```





¡Manos a la obra!

SPRINT 3

# Cart 1

Entregable

Hacer todo lo visto en clase.



- **IMPORTANTE:** Funcionalidad de agregar producto y contador.
  - increaseItem tiene como máximo el stock del producto.
  - decreaseItem tiene como mínimo 1.
  - addItems debe funcionar así: si el item ya existe en el array cart, en ese caso no hace un push, sólo actualiza la cantidad. Además, debemos agregar el producto completo, no sólo el id. La cantidad mínima a agrega al carrito es 1.

URL CART:

[https://github.com/julioavantt/guayerd\\_login/blob/main/cart.png](https://github.com/julioavantt/guayerd_login/blob/main/cart.png)

# 🔍 retro

¿Cómo nos fué?

¿Qué cosas no quedaron claras y  
necesitamos repasar la próxima?

