STEP1. CHOOSING THE PRODUCT

- The page starts with a bootstrap navbar (navbar, navbar-expand-lg, navbar-dark, bg-dark) containing the links to donation process, about us, and how donations help.

- A header section that i used bootstrap classes (py-5, text-center, display-4, lead) used to write the purpose of imc krems charity

- A bootstrap progress bar (progress, progress-bar, progress-bar-striped,progress-bar-animated) indicates the current step you are for 1 to 4.

- The products are displayed using the bootstrap grid system (container, row, col-12 col-md-6 col-lg-3, g-3) to make sure i have a responsive layout.

- Each product is displayed inside a bootstrap card (card, card-body, card-img-top, h-100) including an image, description, price, and action button.

- The add to cart buttons are bootstrap buttons (btn, btn-primary, btn-sm) and trigger the addtocart(name, price) JS function using onclick.

- Bootstrap badges (badge bg-primary, bg-secondary, bg-success) display real-time cart information: item count, total amount, and discount.

Step2. Review cart

- With JS i hide step 1 and i show step 2 using the bootstrap utility class d-none via the showstep(2) function.

- The navbar with aboutus, how donationhelps, etc.. is hidden in steps 2–4 using nav.classList.add("d-none").

- The cart contents are shown inside a bootstrap table (table, table-striped) wrapped in a responsive container (table-responsive).

- The table is generated dynamically in JS by the rendercarttable() function, which creates <tr> and <td> elements based on the cart[] array.

- Each cart item includes action buttons (-, +, remove) styled with bootstrap buttons (btn btn-sm btn-outline-secondary, btn-outline-danger).

- I use the changequantity(index, delta) increases or decreases the item quantity and removes the item if the quantity reaches zero using cart.splice().

- i use the removecartitem(index) for removing a product entirely from the cart.

- A cart summary is displayed using bootstrap utility classes (p-4, bg-light, rounded) showing total amount, discount, and final total.

- The discount calculation is handled by calculatetotals(), applying a 10% discount when three or more items are added, using DISCOUNT_ITEM_LIMIT and DISCOUNT_RATE.

STEP3. BUYER DETAILS

- The form uses bootstrap form components (form-label, form-control) and the layout is structured with the bootstrap grid (row, col-md-6, col-md-4) for responsive web.

- The native browser validation is disabled with novalidate, and form submission is handled by JS using onsubmit="handlecheckout(event)".

- I put handlecheckout(event) to prevent the page to reload with preventDefault() and validates all input fields using validatecheckoutform().

- Jthe validatecheckoutform() checks name, email, address, ZIP code, city, country, and phone number (using the regex /^[0-9]+$/ to allow digits only).

- When an input is invalid, JS applies the bootstrap validation class is-invalid and inserts the error message into the corresponding div.invalid-feedback.

- If validation fails i put a general error message which is displayed using a bootstrap alert (alert alert-danger) inside #formmessagebox.

- When entering step3 previous validation errors are cleared using clearbuyerformvalidation() to avoid showing old messages.

- The "Back" and "Go to step4" buttons are aligned using Bootstrap utilities (d-flex, justify-content-between).

Your data was accepted. Please review the confirmation.

IMC Krems Charity

Choose donation packages, review your cart, enter your details, and confirm.

Step 4 of 4: Confirmation

Step 4: Confirmation

Your donation has been successfully registered! Thank you for your generosity.

Order Summary

Family Food Box × 1	€12.00
Senior Care Package × 1	€13.00
Baby Supplies × 1	€14.00

Family Food Box × 1	€12.00
Senior Care Package × 1	€13.00
Baby Supplies × 1	€14.00
Emergency Tent × 1	€40.00

Total: €79.00
Discount: €7.90
Final Total: €71.10

Buyer Information

JUAN
IMC 33
2346 POLAND
SOUTH AGRIVS
Email: AHUAO@JAUL.COM
Phone: 5739101

Start new donation

STEP4.CONFIRMATION

- The page displays a bootstrap success alert (alert alert-success) confirming that the donation has been successfully registered.

- Yhe bootstrap progress bar at 100% indicates that the user has reached the final step of the process.

- The order summary is generated dynamically using JS and displayed inside a bootstrap list group (list-group, list-group-item, d-flex, justify-content-between).

- Each list item shows the product name, quantity, and subtotal calculated in JS.

- The donation totals section is styled using bootstrap utility classes (p-4, bg-light, rounded) to clearly highlight the total amount, discount and final amount

- Discount values are calculated in JS using calculatetotals() and displayed in real time.

- The buyer Information section is filled dynamically using JS and injected into the page using innerHTML.

- Line breaks (<br>) are used to format buyer details such as name, address, email, and phone number.

- The "Start new donation" button is a bootstrap button (btn btn-primary) and calls resetprocess() to restart the entire donation flow.

- JS resets the cart, clears the form data, updates the UI, and returns the user to Step 1

SUMMARY

1. NAVBAR
Use: Top navigation bar with a responsive menú
Classes: navbar, navbar-expand-lg, navbar-dark, bg-dark, navbar-   brand, navbar-nav, nav-link, navbar-toggler, navbar-toggler-icon, collapse, navbar-collapse
Location: Top of the page
Functionality: Collapsible menu for mobile devices

2. CONTAINER
Use: Main container that centers and limits the width of the content
Classes: container, text-center
Location: Wrapping the main content and inside the navbar

3. GRID SYSTEM
Use: Grid system to organize products in responsive rows and columns
Classes: row, g-3 (gap), col-12, col-md-6, col-lg-3
Location: In all product sections (Step 1)
Functionality: 4 columns on large screens, 2 on medium screens, 1 on mobile screens

4. CARDS
Use: Cards to display each product with image, name, description, and price
Classes: card, h-100, card-img-top, card-body, card-title, card-text
Location: Each product in Step 1

5. BUTTONS
Use: Buttons for actions (add to cart, step navigation, etc.)
Classes: btn, btn-primary, btn-secondary, btn-success, btn-sm, btn-outline-secondary, btn-outline-danger, btn-lg

6. BADGES
Use: Small labels to display cart information (items, total, discount)
Classes: badge, bg-primary, bg-secondary, bg-success
Location: Step 1, below the title
Functionality: Shows the cart summary in real time

7. TABLE
Use: Table to display cart content with products, quantities, prices, and actions
Classes: table, table-striped, table-responsive
Location: Step 2
Functionality: Shows all cart items with options to change quantity or remove ítems

8. ALERTS
Use: Informational messages (success, error, warning, information)
Classes: alert, alert-success, alert-danger, alert-warning, alert-info, d-none

9. FORMS
Use: Form to collect buyer information (Step 3)
Classes: form-label, form-control, invalid-feedback, is-invalid, mb-3
Location: Step 3
Fields: Full name, Email, Address, ZIP code, City, Country, Phone
Functionality: JavaScript validation with visual feedback

10. LIST GROUP
Use: List to display the product summary in Step 4
Classes: list-group, list-group-item, d-flex, justify-content-between, align-items-center
Location: Step 4, in the confirmation summary
Functionality: Shows each product with its quantity and subtotal

11. PROGRESS BAR
Use: Progress bar that shows the current step of the process
Classes: progress, progress-bar, progress-bar-striped, progress-bar-animated, bg-primary
Location: Above all steps
Functionality: Visually shows progress (25%, 50%, 75%, 100%) and descriptive text

12. COLLAPSE (the three stripes when you go to the pone view)
Use: Component to show/hide the navigation menu on mobile devices
Classes: collapse, navbar-collapse
Location: Inside the navbar
Functionality: The menu collapses on small screens

HOW I THOUGHT THE LOGIC AND MY JS:
- The donation process is divided into four steps.
- JavaScript shows and hides each step without reloading the page.
- Products added to the cart are stored in a simple list (array).
- The cart automatically updates when items are added or removed.
- A discount is applied automatically when enough items are added (3) and gone if you remove some.
- The form checks if the user entered valid information.
- At the end, a confirmation page shows the order summary. The user can restart the process without refreshing the page