Exercise Sheet 5

(Web-Shop – Final Tasks)

Submission until: 20.12.2025 23:59h **Presentation:** 22.12.2025

Introduction

In this last task sheet 5, we aim to finalize our web-shop. In particular, we will implement our shopping cart, the collection of all orders, and further interesting features.

If you must save data for your implementation, you can use JSON-files or if you are already an advanced user, you can also use a database, i.e., MariaSQL.

Task 1 (Shopping Cart)

In the previous task sheet, you have prepared options to implement a shopping cart. Now, it is time to implement it for your web-shop. Therefore, you must develop at least the following features:

- A symbol for a shopping cart (empty cart) is shown at the beginning somewhere in your web-shop, e.g., in the top-right corner.
- When products have been selected, they will be added to the shopping cart. The symbol must show another icon, i.e., representing a cart with items.
- When you click on the shopping cart symbol, the corresponding page opens, e.g., shoppingCart.php
- This page must contain at least a list of all selected items, their amount, the price for one item, and the total price for all items. Don't forget to show the taxes as well.
- It must be possible to update the list, i.e., to remove items, and to change the amount of one certain product.
- Only authenticated users can perform an order, otherwise, an authentication option should be given. After the authentication, the user must be "logged-in" at the shoppingList page.
- When an order is done, it will be saved persistently.

Task 2 (Managing Orders)

Now, we need a management interface for our orders.

Customers should be able to view their orders containing a current state, i.e., finished, ordered, shipped but not received, or canceled. As long as the order has not been sent, the customer must be able to cancel the order.

The administrator's user interface has different views. A view for new orders, orders in process, rejected, and finished orders.

After logging in, a list of new orders is presented. The administrator can click on orders that have been sent. That will change the state for the customer as well. Obviously, orders can be rejected. In this case, a reason (e.g., not available products) must be added which will be shown at the customer's side. Furthermore, the administrator can open the list of customers and block/unblock them. Blocked customers are not able to make orders for a shopping cart, e.g., the button is disabled with a suitable comment, e.g., "your account is blocked by the administrator".

Task 3 (Discounts)

After implementing the shopping cart, you can add an interesting new feature. That will be the discount option. Generally, every 10^{th} order can be discounted by 10% and every 20^{th} order can be discounted by 20%. You can optionally(!) enrich this function by the administrator's settings.

Task 4 (Freestyle)

In this task you can add at least three further individual functionalities that let your web-shop stand out from the rest. Generally, you can implement everything you like, but at least one function must consider the customer, e.g., making combined orders – common shopping carts.

Task 5 (Final Steps)

Now, test your final solution. If you find bugs, try to solve them, and improve everything that can interfere the usage of your shop. Prepare your final result for the presentation to the class. We will present the final web-shops to the class. That's your stage! We look forward to see your launch!

Submission hints:

- Upload a solution **ZIP document** containing all solution files (i.e. html, css, js, and pdf).
- Please write all names of the team that participated in the solving process of this task in the PDF file.
- All team members must be present for a certificate in the following course. For a successful acceptance, the proposed solution must be explained by all team members.