Technical Test

I. Create a project that has the following characteristics:

- Any Flutter version
- 2 views, each one with its own route
    - My orders → /my-orders
    - Add/Edit order → /add-order/:id

II. 1st view: My Orders

- This view must include:
    - Title, “My Orders”
    - Table to show the orders created
        - Columns: ID, Order #, date, # Products, Final price, Options
        - The options per row are:
            - Edit Order: Redirect to the Add/Edit Order view (sending the ID)
            - Delete Order: Show a confirmation modal and delete the order
    - Button to add a new order, this will redirect to the Add/Edit Order view

III. 2nd view: Add/Edit Order

- This view must include
    - Title, “Add Order” or “Edit Order” depending on exists ID
    - Form to create/edit the order
        - Order #
        - Date (disabled, auto completed, current date with format)
        - # Products (disabled, auto completed, count of products selected)
        - Final Price (disabled, auto completed, sum of all product’s prices)
    - Button to add a new product to the order, this will open a modal with a simple form

        - ▪ Select to choose the product
        - ▪ Input number with the qty required
        - ▪ Button to confirm and save
    - o Table to list the available products (3 products)
        - ▪ Columns: ID, Name, Unit Price, Qty, Total Price, Options
        - ▪ The options per row are:
            - Edit product: Open a modal to edit the product added in the order (not the product general info)
            - Remove product: Show a confirmation modal and remove the product of the order
    - o Button to save and create the order
- This view must be used for “Create/Add” and “Edit”; the difference will be the exists of ID in the route params

IV. Extra points:

a. Create a view to list, add, delete, and edit the products.

b. Add a new option in the orders table to change the order status

    i. Pending

    ii. InProgress

    iii. Completed

c. Add validations to no edit or modify the completed orders

d. If You will use and SQL DB, work your business rules in Store Procedures

V. Technical Test

1. Create an API (C# or Java Spring boot or Express js or Pyhton)
2. Create a connection to any DB (MYSQL / MongoDB)
4. It’s not needed a login or token to access to the endpoints
5. Use the best practices that you want