Flower Delivery App

Web application where users can order flowers delivery

Overview

There are three complexity levels: **base, middle** and **advanced**.
The base level is sufficient to pass the test task stage. Still, accomplished tasks with a harder level of complexity will highly increase your chance of getting to the school.

Description

The Flower Delivery app should consist of two pages:

- the flower shops page
- the shopping cart page
- order details page

Requirements

Project Documentation

- Add the level of accomplished tasks in the **README.md** file.
- Upload the source code to **GitHub**, **BitBucket**, or **GitLab** and share the **repository link**.
- Host the application in any preferred way and **provide a public URL** for access. Projects without a publicly accessible URL will not be considered (no one will run your project from localhost). The same applies if you host it on your own machine and share the link with us.

Front-End

- Must be implemented using **HTML/CSS/TypeScript** (with or without a framework).
- Any design style may be used (with or without a design framework).

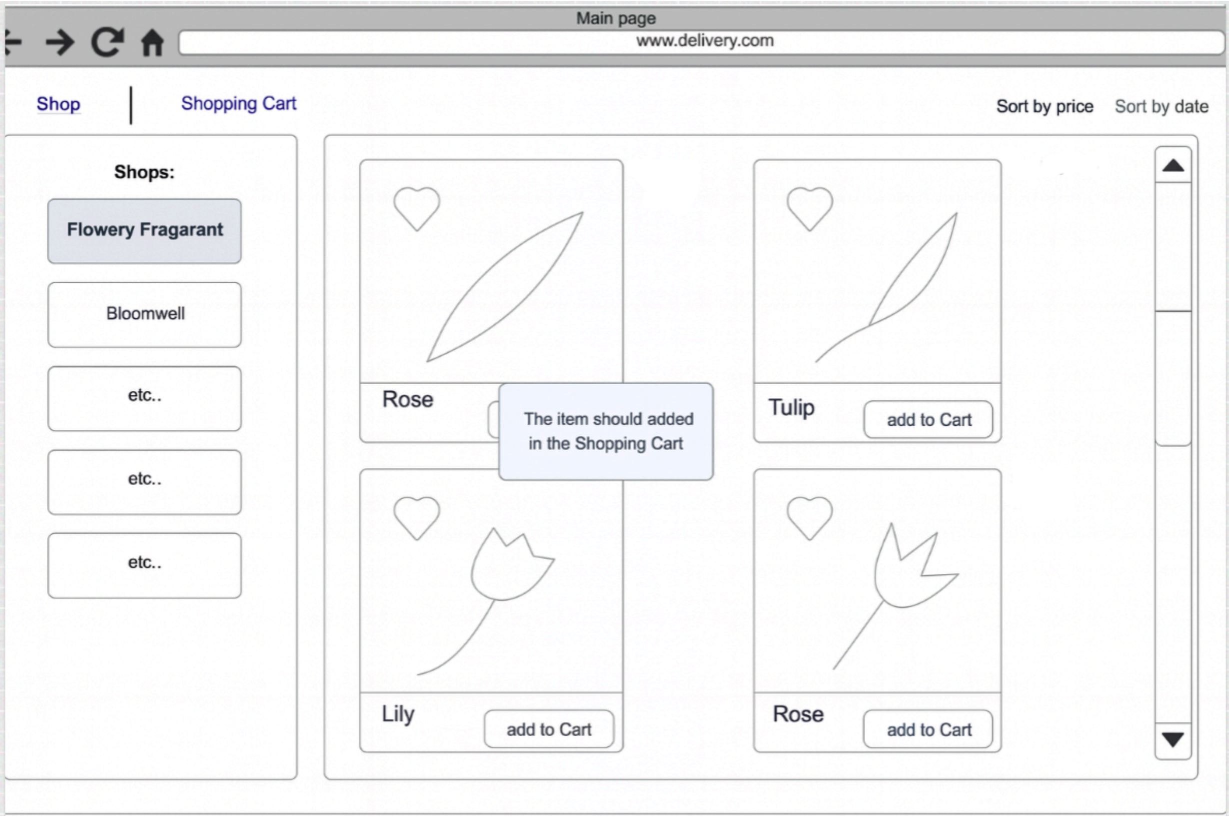
Back-End

- The back-end part can be done with or without any framework. Firebase and similar services are not accepted, as the backend must be built and hosted by you
- Should connect to either a **relational** or **non-relational database** of your choice. Databases in the form of a JSON file or similar, stored inside the project, are not accepted

Base level

Flower shops page:

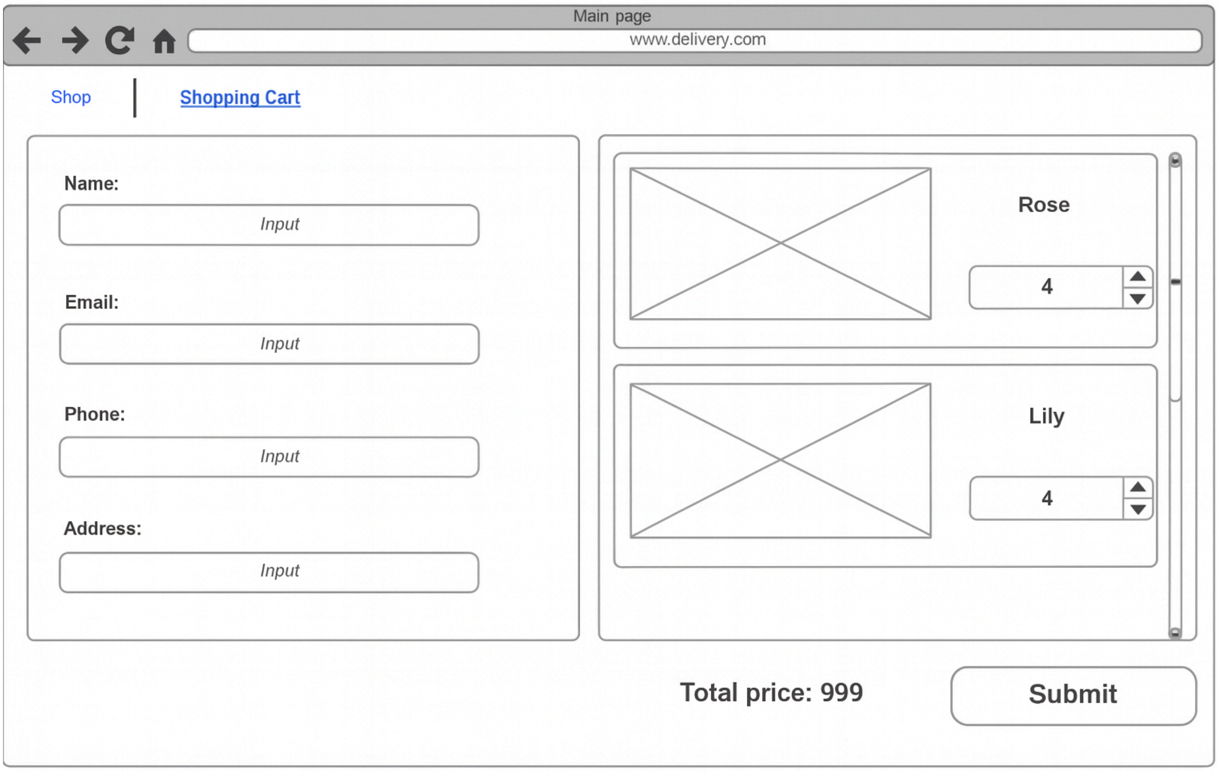
- The page where users can choose a flower shop, then add bouquets or single flowers to the cart (get data from the database).



The item should added in the Shopping Cart

(for example, shop page can look like this)

Shopping cart page:



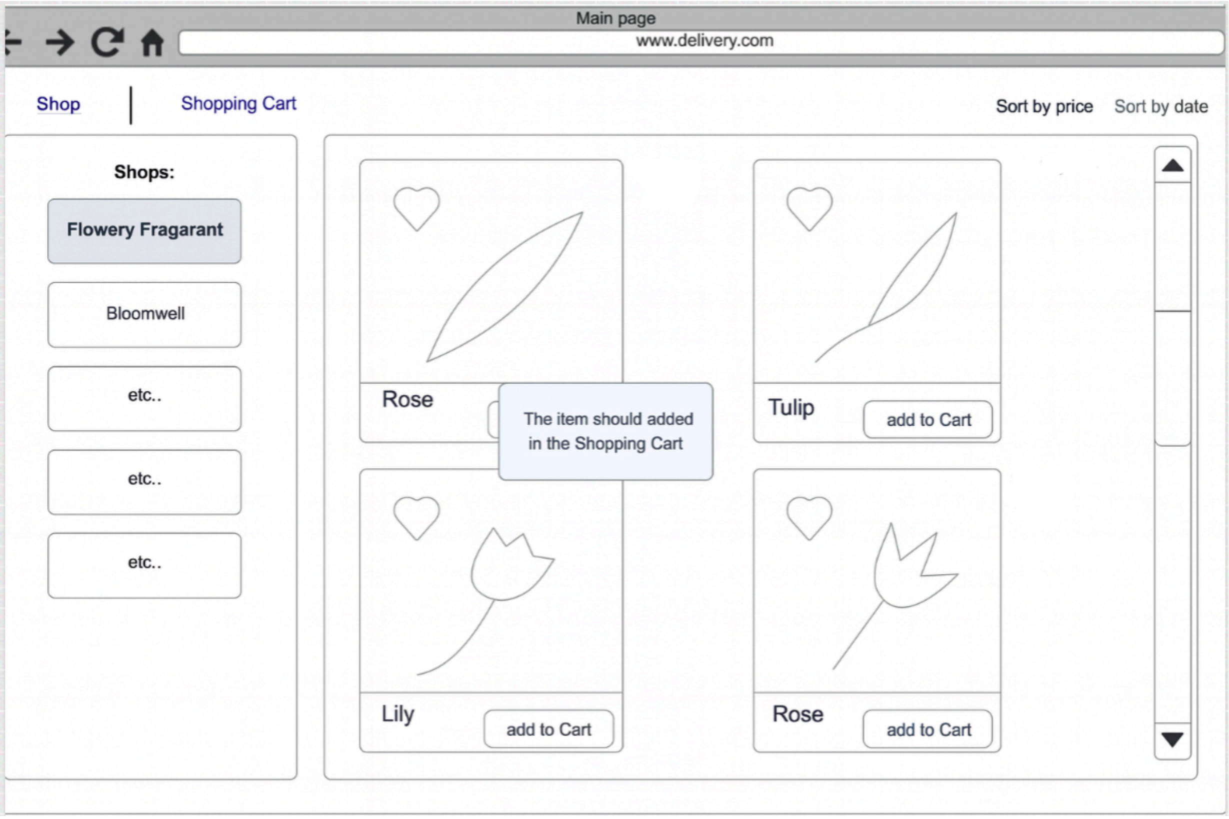
(for example, Shopping cart page can look like this)

- The page where the user can check all added products, remove some of them or change the count.
- Add inputs for email, phone number, and delivery address.
- The order should be saved in the database after the user clicks the “Submit” button.

Middle level

Flower shops page:

- Everything from the base level.
- Add the ability to sort flowers by price and/or date added.
- Add the ability to mark bouquets as favorites. Favorite flowers should be displayed first when sorting.



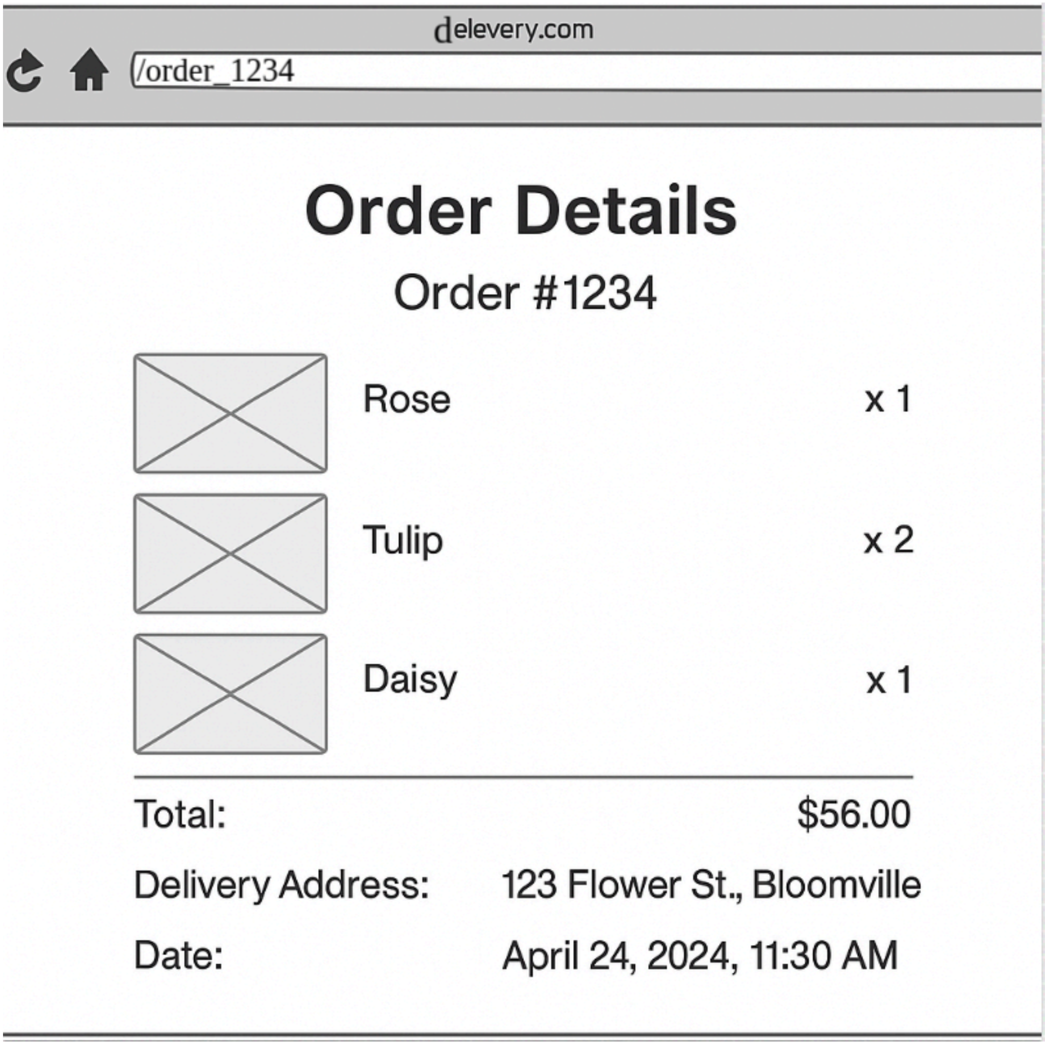
The item should added in the Shopping Cart

(for example, shop page can look like this)

Shopping cart page:

- Everything from the base level.
- The cart should be saved in local storage.
- When creating an order, include **date and time** information that correctly accounts for the customer's **time zone**.

Order details page:



delevery.com

/order_1234

Order Details

Order #1234

Rose	x 1
Tulip	x 2
Daisy	x 1

Total:	$56.00
Delivery Address:	123 Flower St., Bloomville
Date:	April 24, 2024, 11:30 AM

(for example, order page can look like this)

- After successfully placing an order, the user should be automatically redirected to an **order details page**

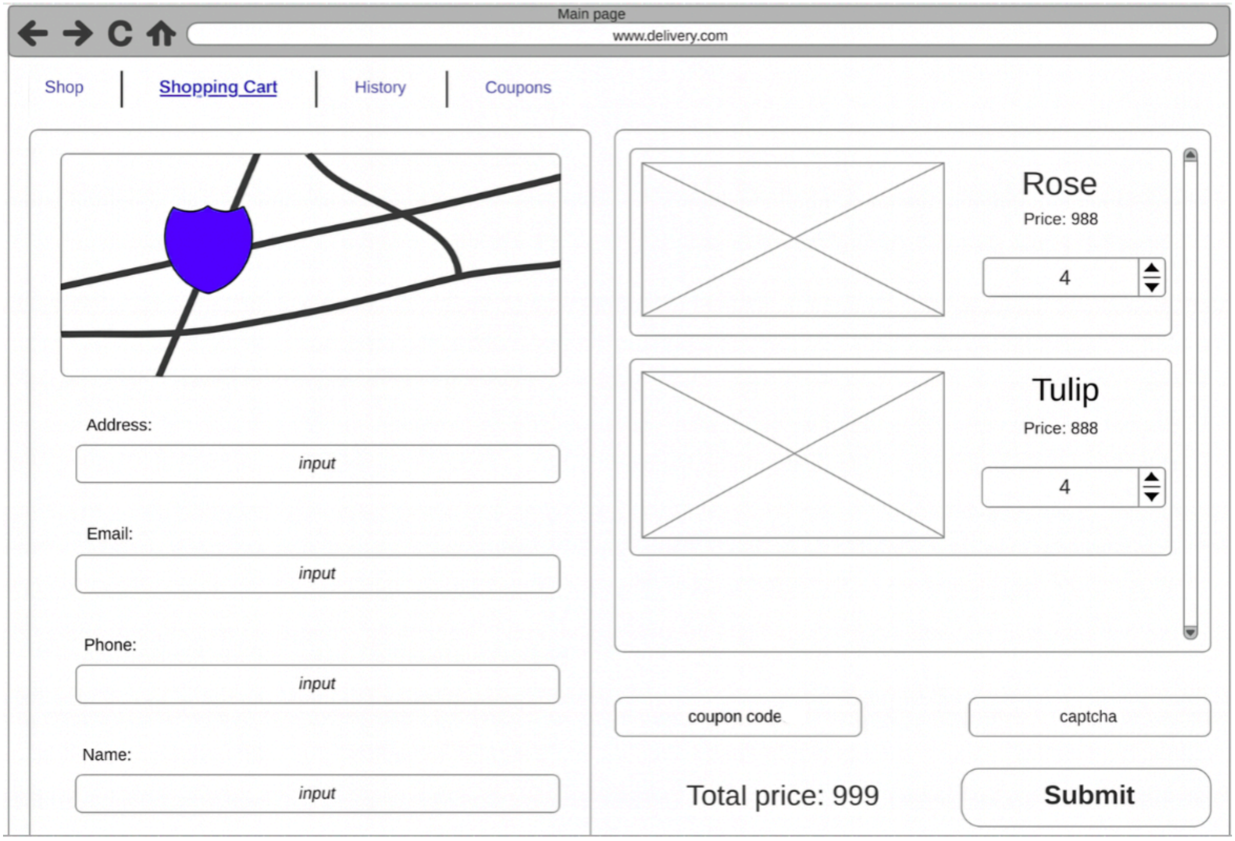
- The order details page must display the following information:
  - Unique order id
  - List of products
  - Total price
  - Delivery address
  - Date and time (adjusted to the user's time zone)

Advanced level

Flower shops page:

- Everything from the middle level.
- Add pagination to make browsing the list of flowers more convenient and optimized for large catalogs.

Shopping cart page:



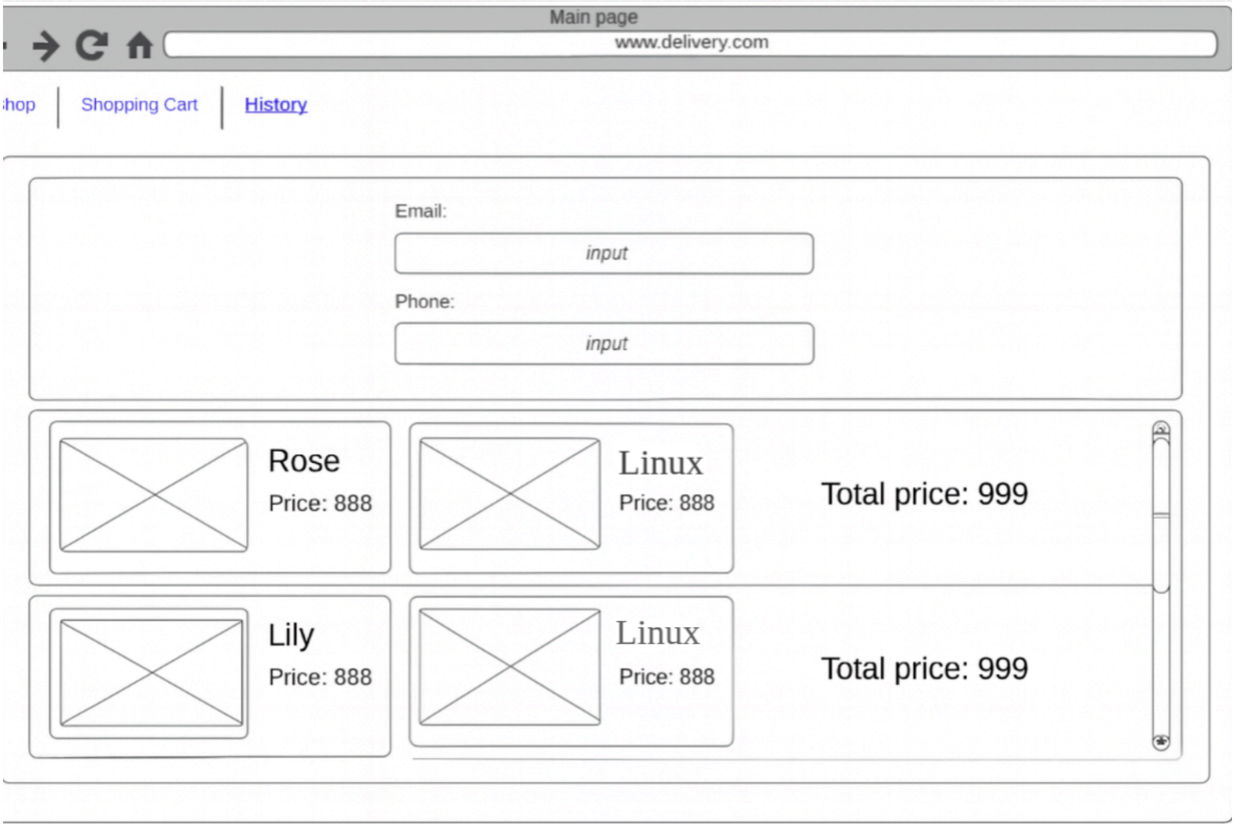
(for example, Shopping cart page can look like this)

- Everything from the middle level.
- Add Google Maps:
  - Users can choose their delivery address using a pin on the map or just enter an address, and it will be shown on the map.

- Show the flower shop from which the user ordered flowers on the map.
- *(extra)* Show the route from shop to user's address and approximate delivery time.
- *(extra)* Ask to enter a captcha after clicking on the "Create order" button.

Additional ideas for your inspiration:

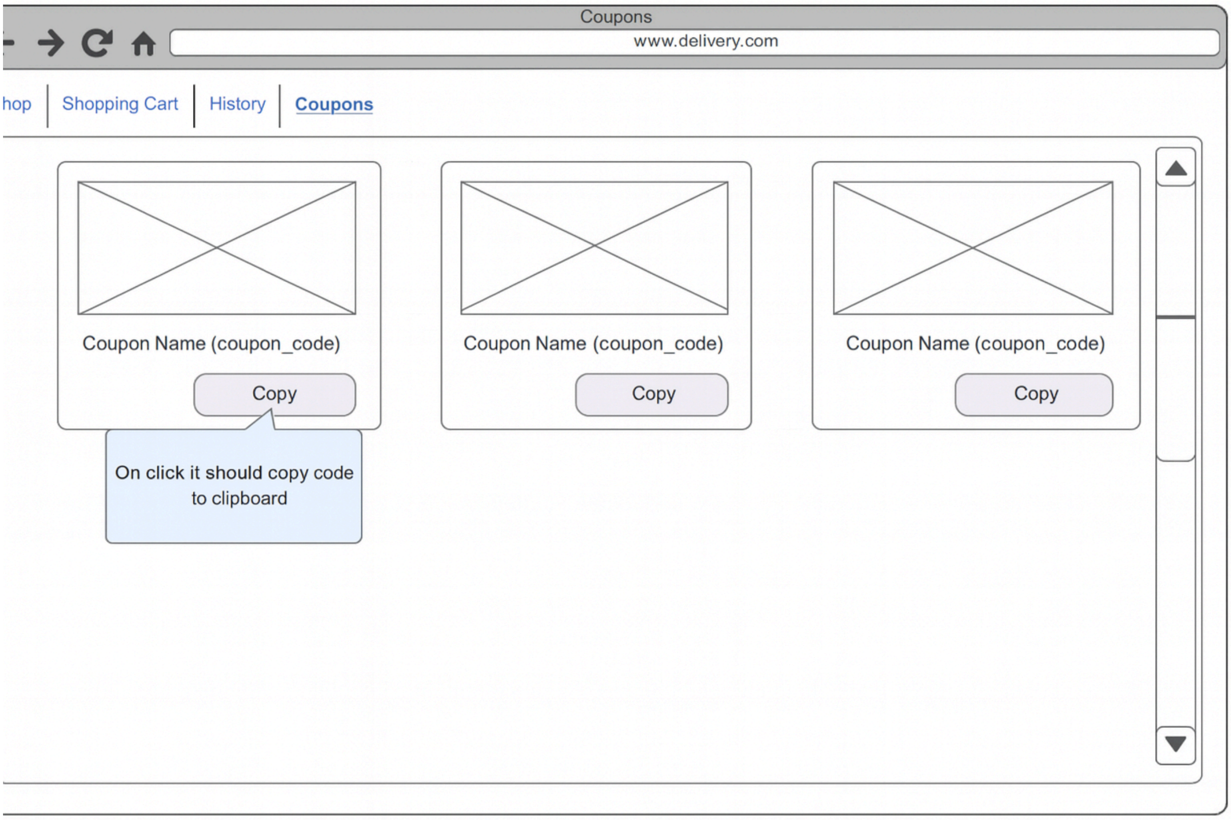
Orders history page:



(for example, history page can look like this)

- Users can find their orders on this page using their email and phone number, or order ID.

Coupons page:



Coupons
www.delivery.com
hop
Shopping Cart
History
Coupons
Coupon Name (coupon_code)
Copy
On click it should copy code to clipboard
Coupon Name (coupon_code)
Copy
Coupon Name (coupon_code)
Copy

(for example, coupons page can look like this)

- Create a page where users can see all coupons (coupons will be used for discounts, users can apply them on the shopping cart page).