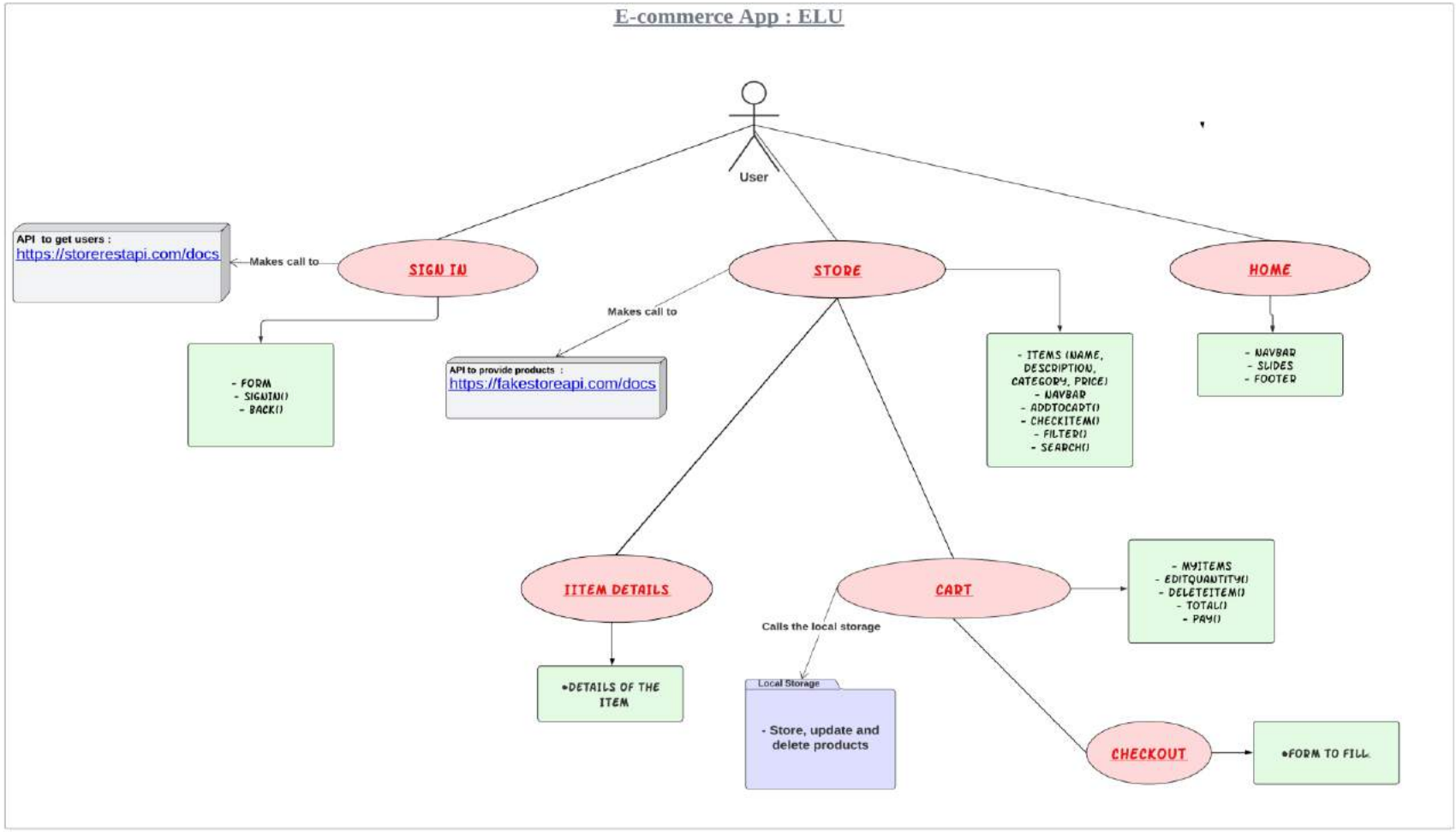
First name : Dorra
Last name : kahla
Matriculation N° : 92108575
Course ID: **DLBCSPJWD01**

The link to my GitHub repository : https://github.com/Do-raa/Store_app.git

Overview

- This application aims to facilitate the process of shopping by using the internet . It provides different articles (clothes, jewelry and electronic devices) that the user can search for them by filtering and obtain more details about them by clicking on the desired card or just add them to their carts . By doing so, they are able to increase or decrease the quantity of each chosen article and remove it from the cart if they want to or simply continue their purchase and go to the check out page by clicking the checkout button . Here, the user needs to sign in if he didn’t sign in yet (please note that only users who already exist in the used APIs are allowed to sign in) and continue the payment’s process . After filling the check out form successfully, the user will get a simple message that confirms the success of the operation .

Diagram

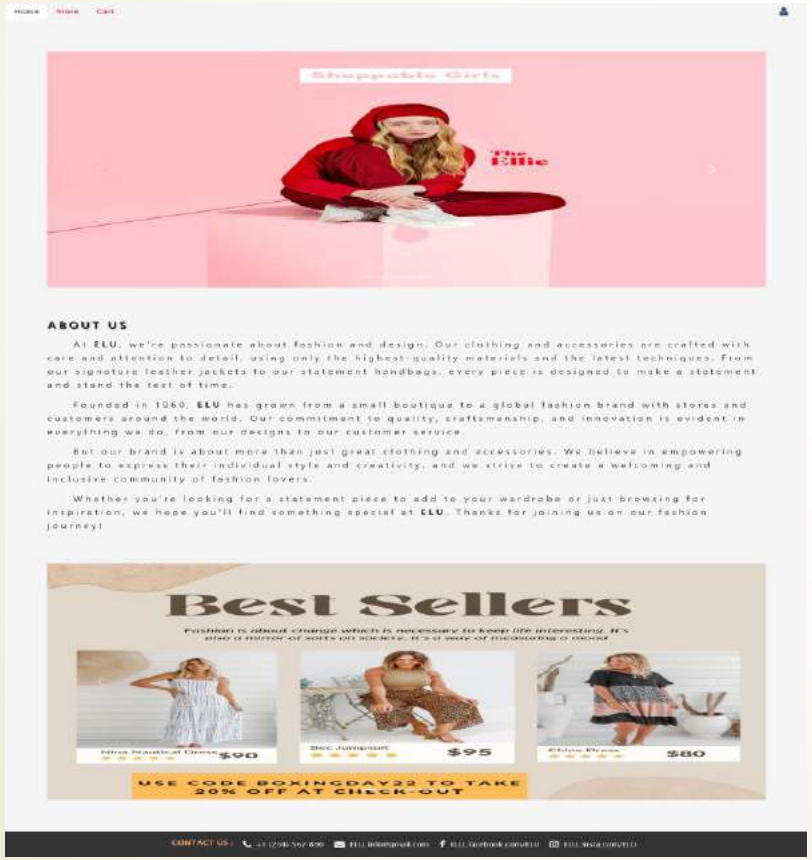

E-commerce App : ELU
User
API to get users :
https://storerestapi.com/docs
Makes call to
SIGN IN
- FORM
- SIGNIN()
- BACK()
STORE
Makes call to
API to provide products :
https://fakestoreapi.com/docs
- ITEMS (NAME, DESCRIPTION, CATEGORY, PRICE)
- NAVBAR
- ADDTOCART()
- CHECKITEM()
- FILTER()
- SEARCH()
HOME
- NAVBAR
- SLIDES
- FOOTER
IITEM DETAILS
•DETAILS OF THE ITEM
CART
Calls the local storage
Local Storage
- Store, update and delete products
- MYITEMS
- EDITQUANTITY()
- DELETEITEM()
- TOTAL()
- PAY()
CHECKOUT
•FORM TO FILL

Technology choices

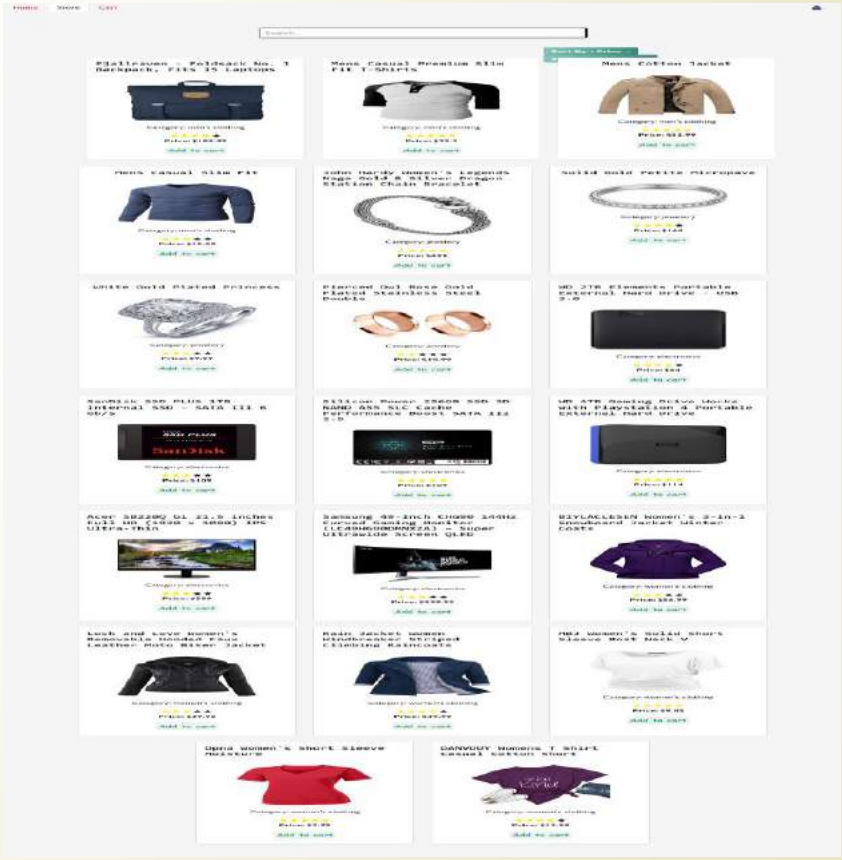
- In one hand, I used CSS, the Bootstrap5 framework and the Font Awesome library to design my application and implement some icons , in addition to HTML5 and JavaScript for structuring my website and making it dynamic . In the other hand, I used the APIs of the Fake Store API website (https://fakestoreapi.com/docs) to get products and the StoreApi website (https://storerestapi.com/docs) to get users .
- For the storage, I chose to use local storage to store the picked products and the signed in user . (Please note that you have to refresh the website in case the updates doesn’t show up to you because we are dealing with the localstorage of the window) .
- VS code was used as an editor and the debugging was performed using the console .
- LUCIDCHART was used to model the architecture of my system .

Screenshots of the website

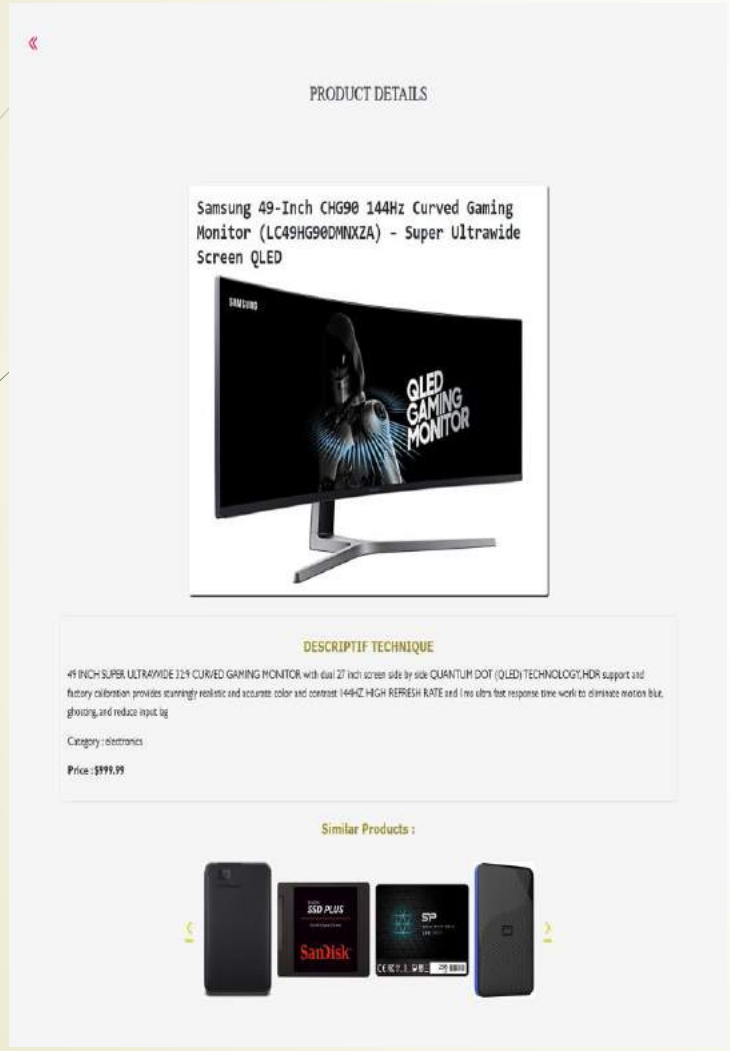
The home page



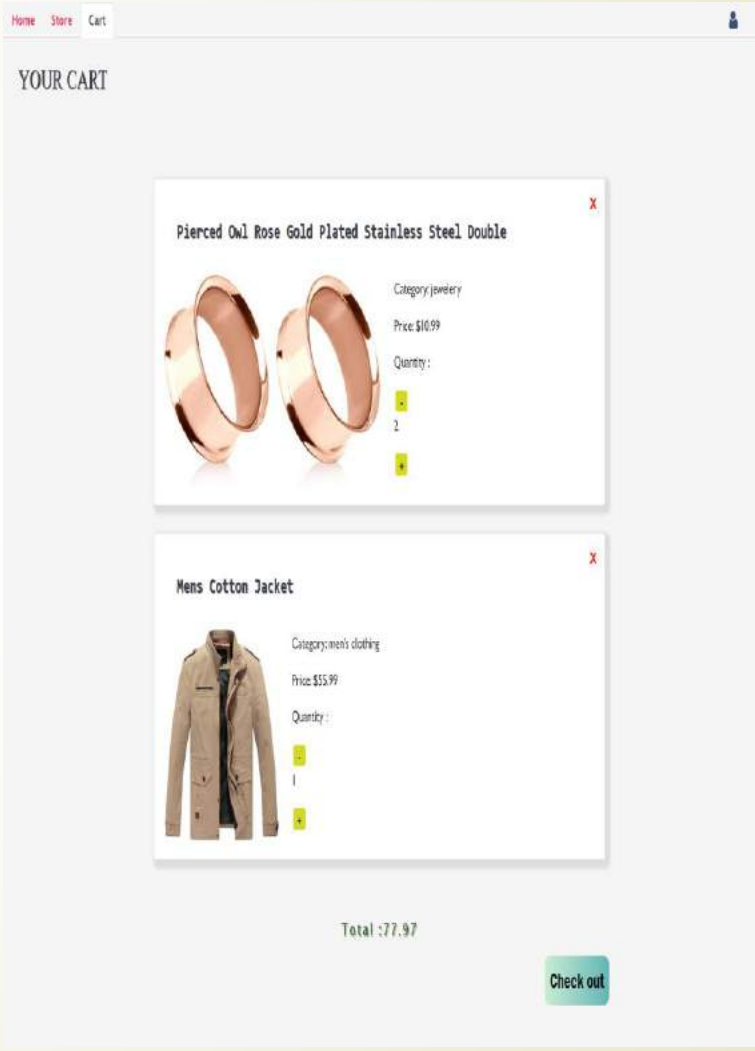
The store page



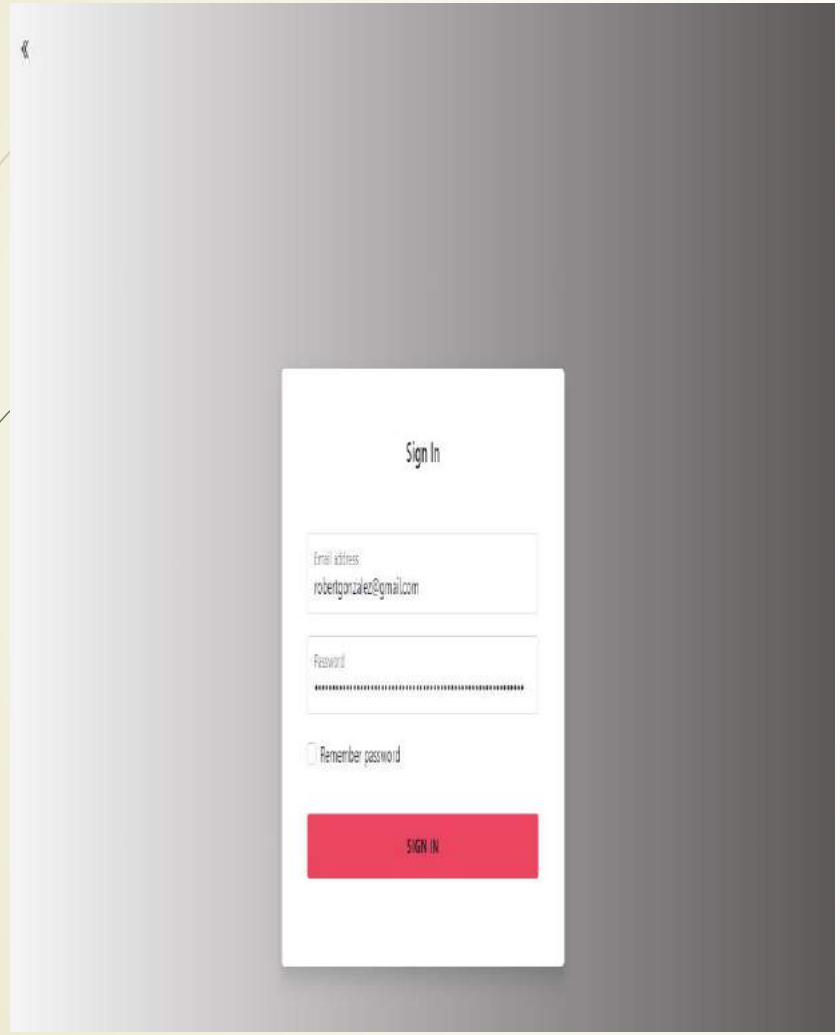
The details page



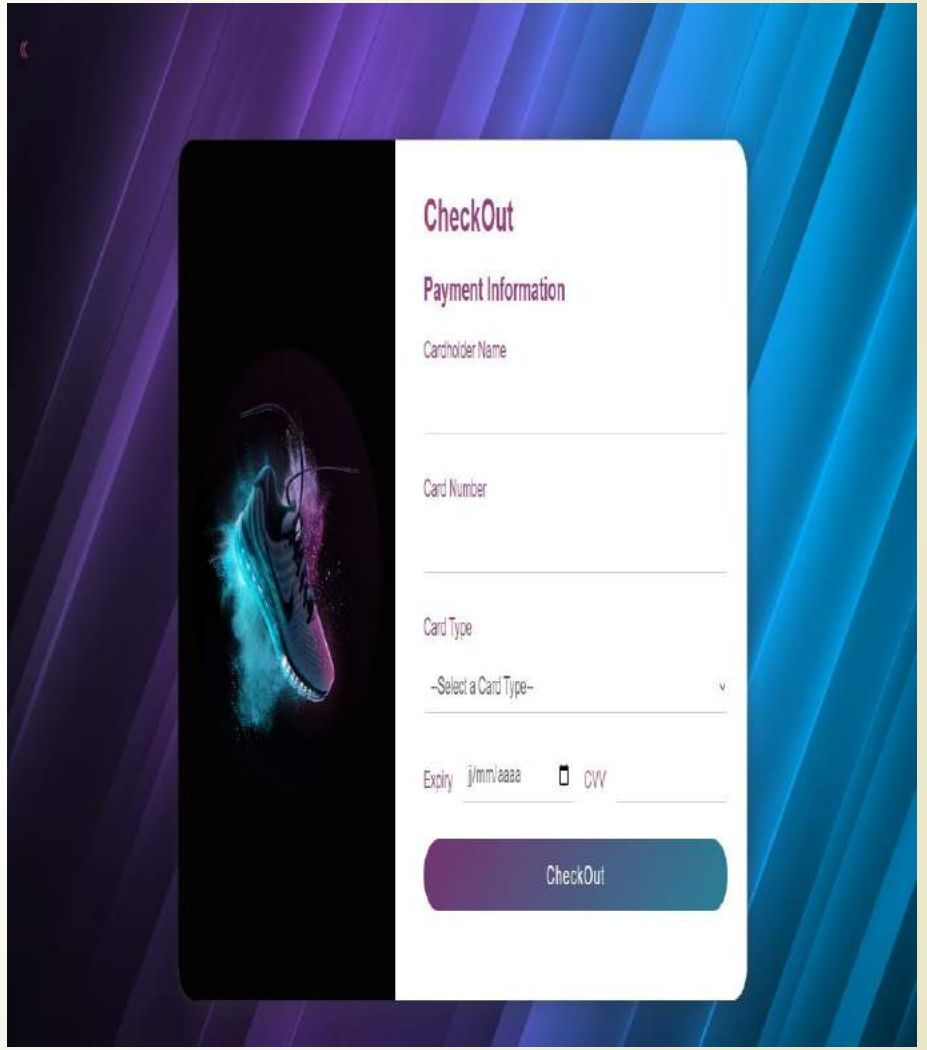
The cart page



The sign in page



The check out page



Updates

- In the conception phase, it was expected that I'll use the API of the Platzi Fake Store API website , but I changed my mind because they weren't really reliable as the images were changing everytime without any reason which was weird actually .
- Also, I remodeling my software based on the C4 model: https://c4model.com/ , taking into account the tutor's advice

The screencast video of the application