

## Header може мати наступний вигляд:

1. Якщо юзер не залогінен:



2. Якщо юзер залогінен:

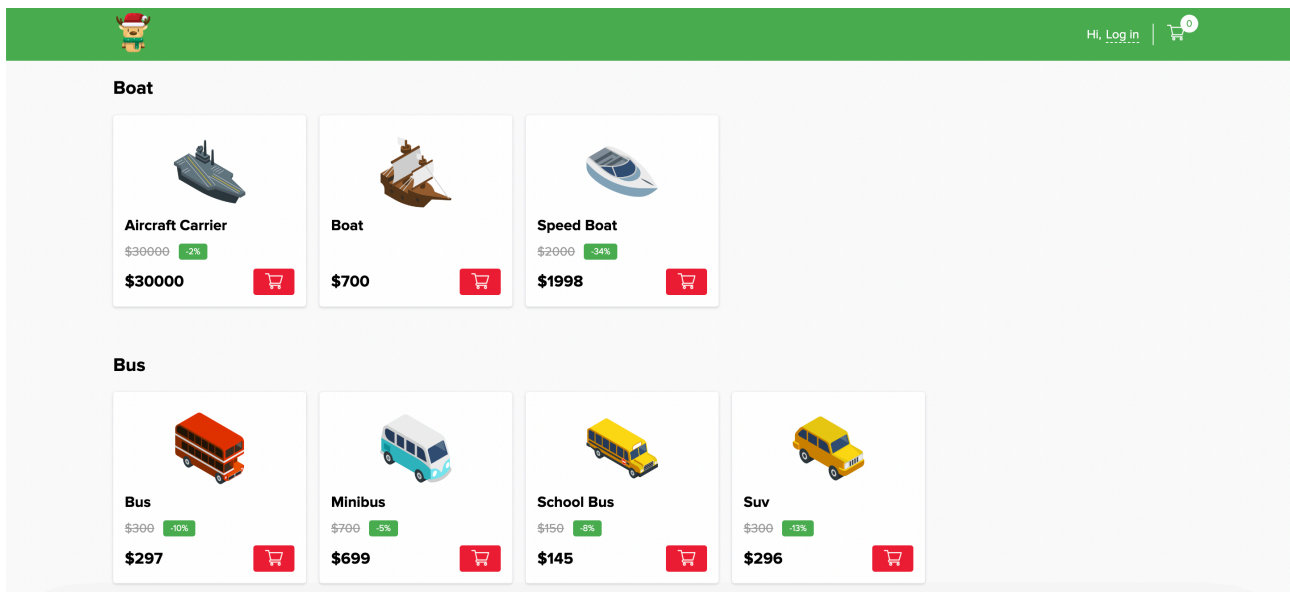


## При натисканні на:

1. Лого: редирект на головну сторінку index.html
2. Sign in: редирект на сторінку логіну/реєстрації login.html.
3. Якщо юзер залогінен, то замість Sign in виводимо його значення name і при натисканні – редирект на сторінку акаунту account.html.
4. Корзина:
  1. Якщо юзер залогінен: редирект на сторінку товарів shoppingCart.html.
  2. Якщо юзер не залогінен: редирект на сторінку логіну/реєстрації login.html.
  3. В білому кружечку відображається кількість товарів, які залогінений користувач додав в корзину.
5. Log out: Кнопка відображається, якщо юзер залогінен
  1. В базі для залогіненого юзера змінюємо значення **status: false**.
  2. Редирект на головну сторінку index.html.

## Головна сторінку сатйу index.html:

1. Якщо юзер не залогінен:



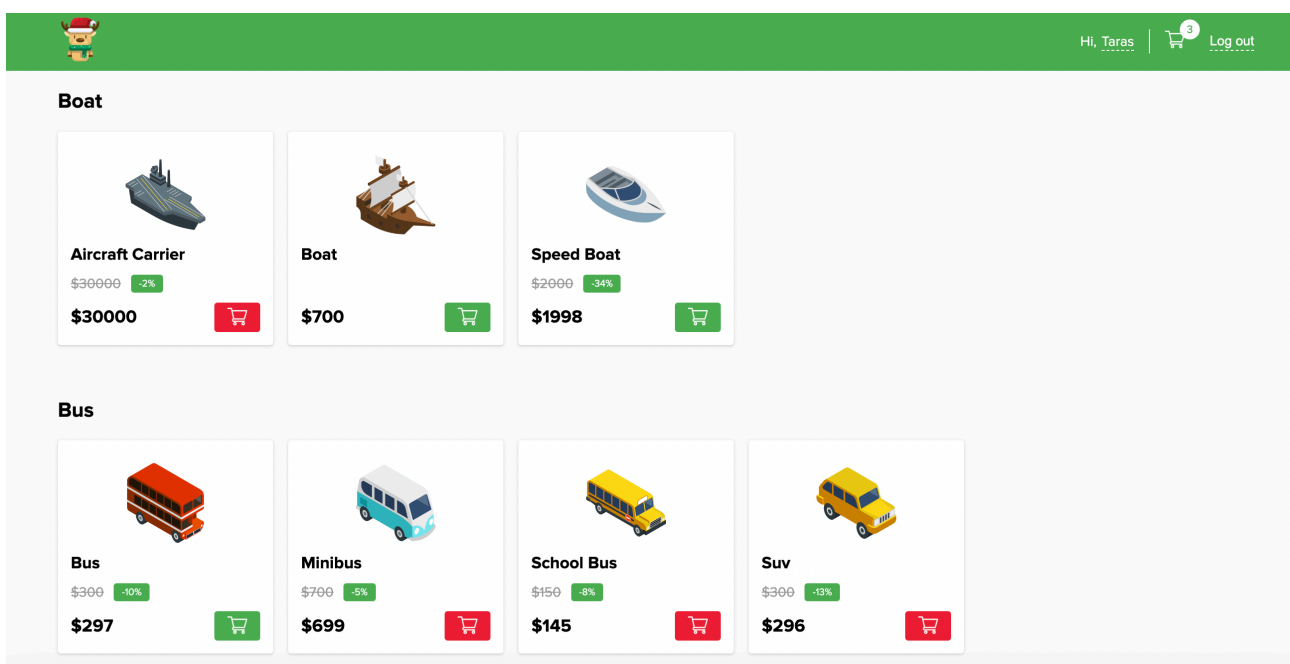
2. Якщо юзер залогінен:

1. В header:

1. Log in (href='login.html') => Taras (href='account.html')
2. Вивід кількості товарів в корзині юзера. href='login.html' => href='shoppingCart.html'
3. Вивід кнопки Log out.

2. В блоці товарів:

1. Товар, який доданий в корзину юзера виводиться з кнопкою зеленого кольору, замість червоного.



## Блоки на головній сторінці:

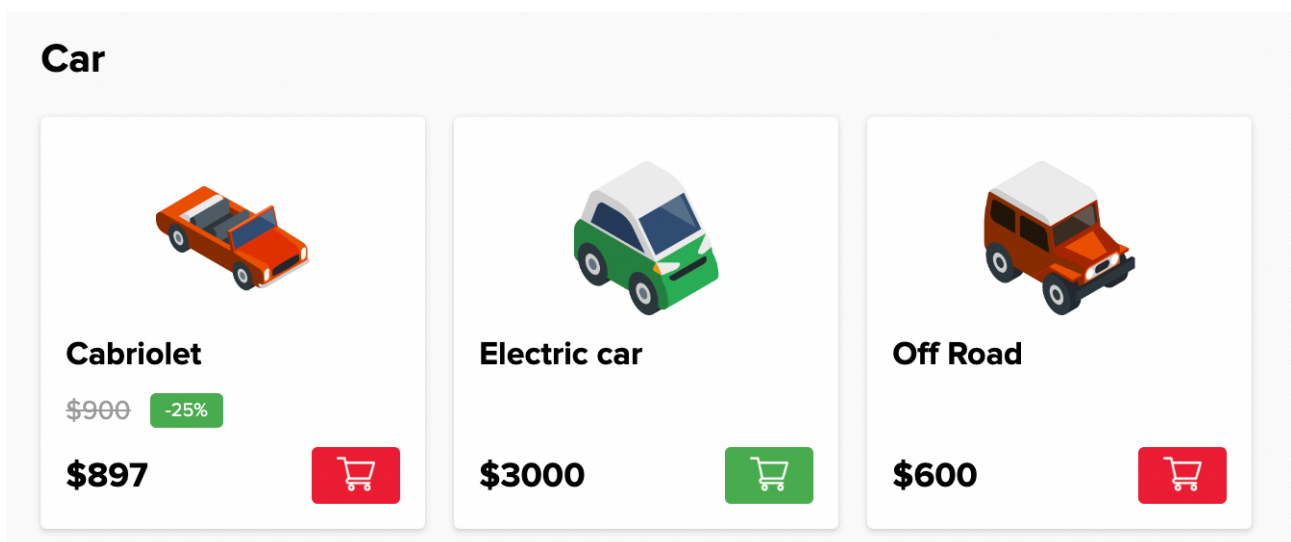
1. Header.
2. Вивід категорій товарів.

## Вивід категорій товарів (index.html)

1. Дані беремо з API: <https://634e9f834af5fdff3a625f84.mockapi.io/products>.  
Приклад масиву з товарами також є в data/products.json.
2. Рендер категорій робимо в блок `div#categoriesContainer`, який знаходиться в `index.html`. Вивід кожної ктегорії має вигляд:

```
<section class="category" data-name="Aircraft">
  <h2>Aircraft</h2>
  <div class="category__container">...</div>
</section>
```
3. В залежності від назви категорії товару – виводимо його у відповідну секцію (Boat, Car, Aircraft, Helicopter, Bus, Bike).

Блок з інформацією щодо товару може мати один із наступних виглядів:



1. Зображення товару.
2. Назва товару (Cabriolet)
3. Якщо у товару є властивість `sale: true`, то виводимо інформацію про знижку.

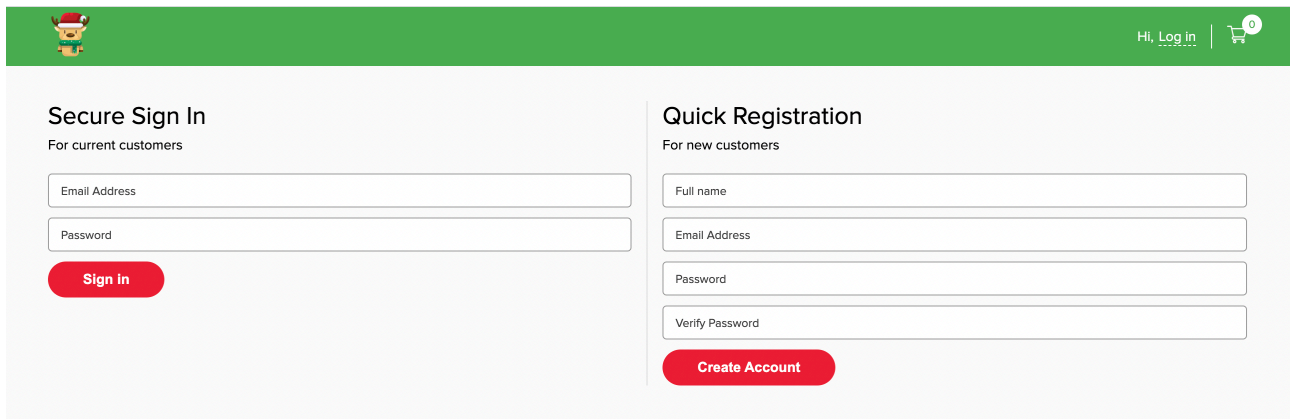
~~\$30000~~ -2%

, де:

1. Сірим кольором виводиться вартість товару `price`.
  2. В зеленому блоці виводиться відсоток знижки `salePercent`.
4. Вартість товару `price`.
1. Якщо у даного товару властивість `sale: true`, то вартість виводимо з урахуванням знижки.

5. Колір кнопки з корзиною показує, чи знаходиться товар в масиві shoppingCart залогіненого юзера.
  1. Якщо товар **в** масиві shoppingCart залогіненого юзера, то колір кнопки – зелений (додаємо до кнопки клас **product\_\_cart—in**).
  2. Якщо товар **НЕ** в масиві shoppingCart залогіненого юзера, то колір кнопки – червоний.
  3. При натисканні на кнопку з корзиною:
    1. Якщо юзер не залогінений, то редирект на сторінку login.html
    2. Якщо юзер залогінений і товар раніше не був доданий в масив shoppingCart, то:
      1. Змінюємо колір кнопки на зелений (додаємо до кнопки клас **product\_\_cart—in**).
      2. Додаємо інформацію про товар в масив shoppingCart юзера у вигляді об'єкту {id: product.id, count: 1}.
      3. В header в білому кружечку з виводом кількості товарів в корзині змінюємо значення.
    3. Якщо юзер залогінений і товар раніше був доданий в масив shoppingCart, то:
      1. Змінюємо колір кнопки на червоний (видаляємо у кнопки клас **product\_\_cart—in**).
      2. Видаляємо об'єкт з інформацією про товар з масиву shoppingCart юзера.
      3. В header в білому кружечку з виводом кількості товарів в корзині змінюємо значення.

## Сторінка логіну/реєстрації login.html:



**Secure Sign In**  
For current customers

Email Address

Password

**Sign in**

**Quick Registration**  
For new customers

Full name

Email Address

Password

Verify Password

**Create Account**

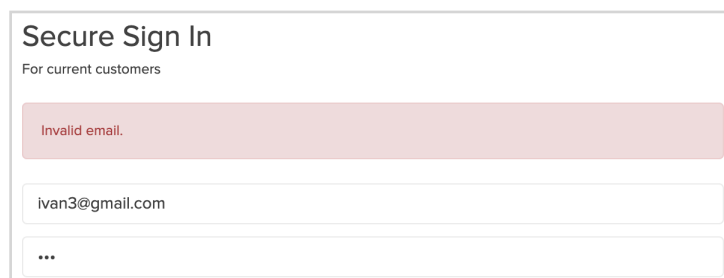
Дані беремо з API: <https://634e9f834af5fdff3a625f84.mockapi.io/users>. Приклад масиву з юзерами також є в data/users.json.

### Форма Sign in:

1. Поле email.
2. Поле password.
3. Кнопка Sign in.

При submit форми:

1. Перевіряємо, чи є в базі юзер з введеним email і password.
2. Якщо:
  1. Юзер з даним email не існує в базі, то показуємо блок .error (додати клас active) з текстом Invalid email.



**Secure Sign In**  
For current customers

Invalid email.

ivan3@gmail.com

...

2. Юзер з даним email існує в базі, але пароль не підходить, то показуємо блок .error (додати клас active) з текстом Invalid password.



**Secure Sign In**  
For current customers

Invalid password.

ivan@gmail.com

.....

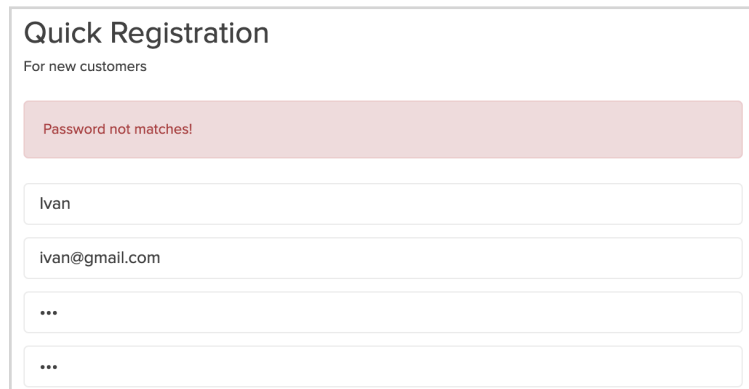
3. Юзер з даним email існує в базі і пароль підходить, то для даного юзера в базі змінюємо властивість **status: true** і редиректимо на головну сторінку index.html.

## Форма Registration:

1. Поле name.
2. Поле email.
3. Поле password.
4. Поле verify password.
5. Кнопка Create account.

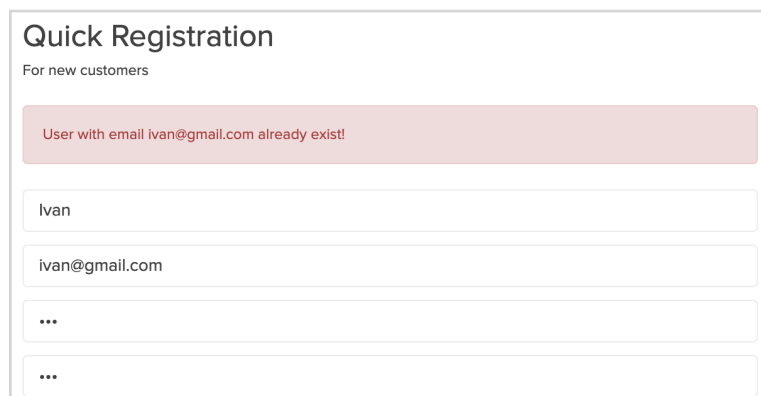
При submit форми:

1. Якщо значення полів password і verify password не співпадає, то показуємо блок .error з текстом Password not matches!



The screenshot shows a web form titled "Quick Registration" with the subtitle "For new customers". A red error message "Password not matches!" is displayed at the top. Below the error, there are four input fields: the first contains "Ivan", the second contains "ivan@gmail.com", and the third and fourth fields contain three dots "...".


2. Перевіряємо, чи є в базі юзер з введеним email.
3. Якщо:
  1. Юзер з даним email вже існує в базі, то показуємо блок з текстом User with email <user email> already exist!




The screenshot shows the same "Quick Registration" form. A red error message "User with email ivan@gmail.com already exist!" is displayed at the top. The input fields below contain the same values as in the previous screenshot: "Ivan", "ivan@gmail.com", and two fields with "...".







2. Юзер з даним email не існує в базі, то додаємо даного юзера в базу з властивістю **status: true** і редиректимо на головну сторінку index.html.
3. Властивості **orders: []** і **shoppingCart: []** додаються автоматично в базі кожному юзеру.

## Сторінка доданих в корзину товарів юзера shoppingCart.html



Hi, Katya |  3 [Log out](#)

### Items in Shopping Cart

Item Description	Price	Sale	Quantity	Total	Action
 Speed Boat	\$2000	- 34%	<input type="text" value="3"/>	\$5994	
 Minibus	\$700	- 5%	<input type="text" value="1"/>	\$699	
 Off Road	\$600	-	<input type="text" value="2"/>	\$1200	

### My Order Summary

Order Total\$7900

Complete Order

1. Таблиця «Items in Shopping Cart» з переліком товарів в корзині.
2. Блок «My Order Summary» з загальною сумою за всі товари в корзині і кнопкою для оформлення замовлення.


### Таблиця «Items in Shopping Cart»


1. Для залогіненого юзера з масиву shoppingCart виводим дані в таблицю.
2. При зміні кількості одиниць товару:
  1. Перераховуємо для даного товару ціну (з урахуванням знижки).
  2. Перераховуємо загальну вартість всіх товарів в корзині – блок «My Order Summary».
3. При натисканні на корзину (видалення):
  1. Видаляємо з бази у залогіненого юзера з масиву shoppingCart об'єкт з інформацією по даному товару {id: '1', count: 2}.
  2. Видаляємо з таблиці рядок з товаром.
  3. В header в білому кружечку змінюємо кількість товарів в shoppingCart.

### Блок «My Order Summary»




1. Для залогіненого юзера на основі даних з масиву shoppingCart виводимо загальну вартість всіх товарів, з урахуванням знижки і кількості одиниць, в поле Order Total.
2. При submit форми #orderSummary:
  1. В базі для залогіненого юзера всі об'єкти з масиву shoppingCart переносимо в масив orders.
  2. В базі для залогіненого юзера з масиву shoppingCart видаляємо всі об'єкти.
  3. Редирект на сторінку account.html.

## Сторінка юзера account.html



Hi, Katya | <sup>3</sup> [Log out](#)

### Ordered Items

	Item Description	Price	Sale	Quantity	Total
	Speed Boat	\$2000	- 34%	4	\$7992
	Minibus	\$700	- 5%	1	\$699
	Aircraft Carrier	\$30000	- 2%	4	\$120000

### My Info

Name: Taras

Email: taras.grygorovych@gmail.com

Delete account

1. Таблиця «Ordered Items» з переліком придбаних товарів.
2. Блок «My Info» з виводом name, email юзера і кнопкою для видалення акаунту.

### Таблиця «Ordered Items»

1. Для залогіненого юзера з масиву orders виводим дані в таблицю.

### Блок «My Info»

1. Для залогіненого юзера виводимо значення властивостей name, email.
2. При натисканні на кнопку Delete account:
  1. В базі для залогіненого юзера змінюємо значення **status: false**.
  2. Редирект на головну сторінку index.html.