

### ТЗ проекта:

#### Шапка сайта может иметь следующий вид:

1. Если пользователь не залогинен:



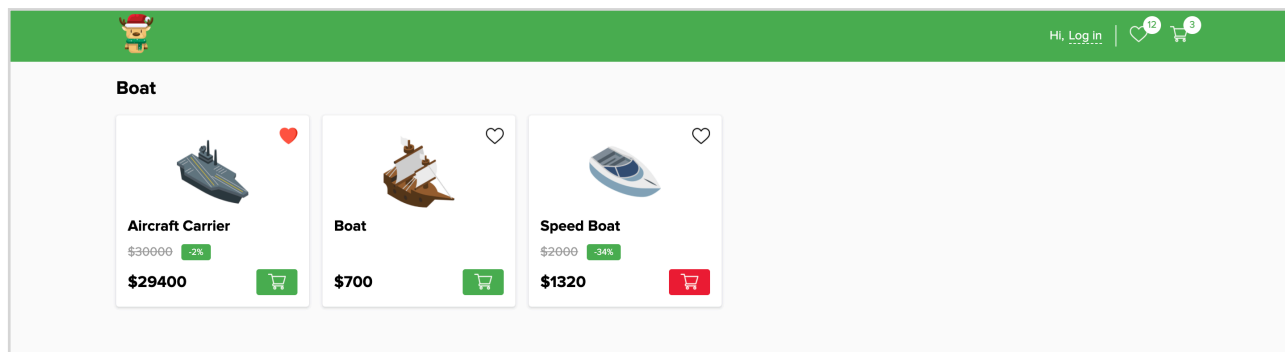
2. Если пользователь залогинен:



#### При нажатии на:

1. Лого: пользователя редиректит на главную страницу index.html
2. Sign in: пользователя редиректит на страницу логина/регистрации login.html.
3. Если пользователь залогинен, то вместо Sign in выводится его поле name и при нажатии пользователя редиректит на страницу аккаунта account.html
4. Сердечко:
  1. Если пользователь залогинен: пользователя редиректит на страницу любимых товаров favourites.html
  2. Если пользователь не залогинен: пользователя редиректит на страницу логина/регистрации login.html
  3. В белом кружочке рядом с иконкой сердечка отображается количество товаров, которое залогиненный пользователь добавил в Избранное.
5. Корзину:
  1. Если пользователь залогинен: пользователя редиректит на страницу заказов shoppingCart.html
  2. Если пользователь не залогинен: пользователя редиректит на страницу логина/регистрации login.html
  3. В белом кружочке рядом с иконкой корзины отображается количество товаров, которое залогиненный пользователь добавил в корзину.
6. Log out:
  1. Кнопка появляется если пользователь залогинен.
  2. Для данного пользователя в localStorage меняем значение свойства status на false.
  3. Пользователя редиректит на главную страницу index.html.

## Главная страница сайта index.html:

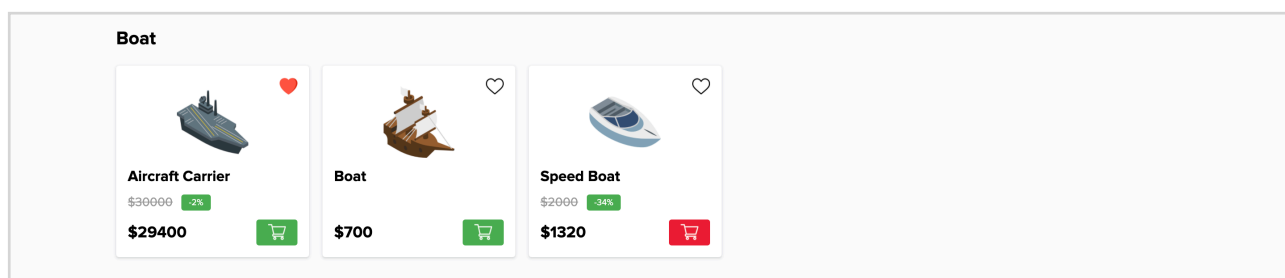


### Блоки на главной странице:

1. Шапка сайта
2. Вывод категорий продуктов

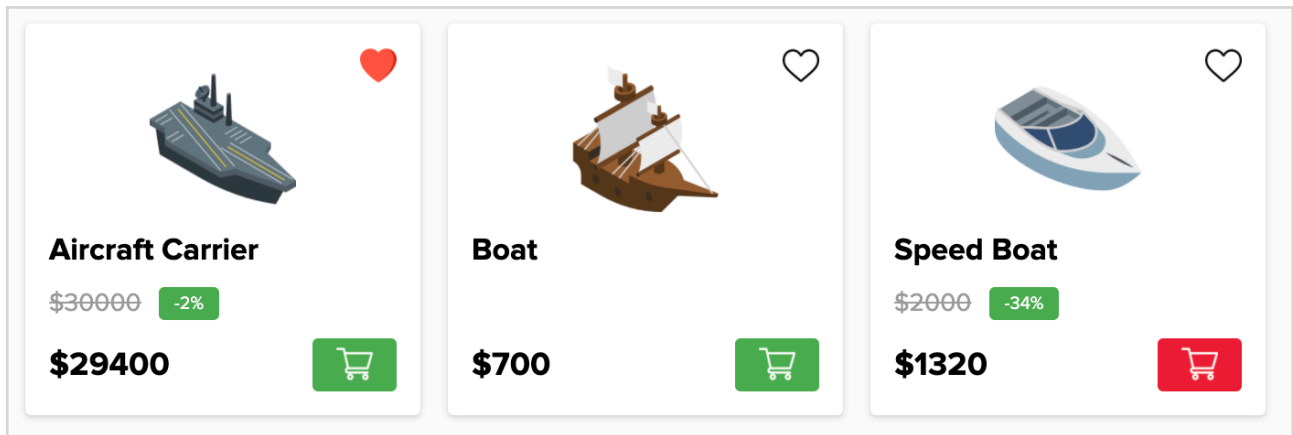
### Вывод категорий товаров (index.html)

Блок категории имеет следующий вид:



1. Данные берем с массива PRODUCTS в файле js/script.js и записываем в `localStorage.setItem('products', JSON.stringify(PRODUCTS))`. Вывод категорий производим в блок `<div class="container" id="categoriesContainer"></div>`, который находится на странице index.html.
2. В зависимости от названия категорий, в которых находится товар, выводим на страницу необходимое количество категорий (Boat, Car, Aircraft, Helicopter, Bus, Bike). Если товар находится в нескольких категориях, например Car и Bus, то выводим этого товар сразу в двух категориях.
3. Заглавие блока Boat – название категории.

Блок с информацией по товару может иметь один из следующих видов:



1. Вывод иконки сердечка, которое показывается, товар находится в массиве favourites или нет.
  1. Если товар в favourites, то картинка сердечка `images/product__favourite—true.png`
  2. Если товар не в favourites, то картинка сердечка `images/product__favourite.png`
  3. При нажатии на сердечко:
    1. Если пользователь не залогинен, то редиректим его на страницу `login.html`
    2. Если пользователь залогинен и товар ранее не был добавлен в favourites, то:
      1. Меняем картинку сердечка на `images/product__favourite—true.png`
      2. Добавляем id товара в массив favourites в localStorage для данного юзера.
      3. В шапке сайта в блоке с выводом количества избранных товаров меняем значение.
    3. Если пользователь залогинен и товар ранее был добавлен в favourites, то:
      1. Меняем картинку сердечка на `images/product__favourite.png`
      2. Удаляем id товара с массива favourites в localStorage для данного юзера.
      3. В шапке сайта в блоке с выводом количества избранных товаров меняем значение.
2. Вывод изображения товара
3. Вывод названия товара (Aircraft Carrier)
4. Если у товара свойство `sale: true`, то выводим информацию о скидке.

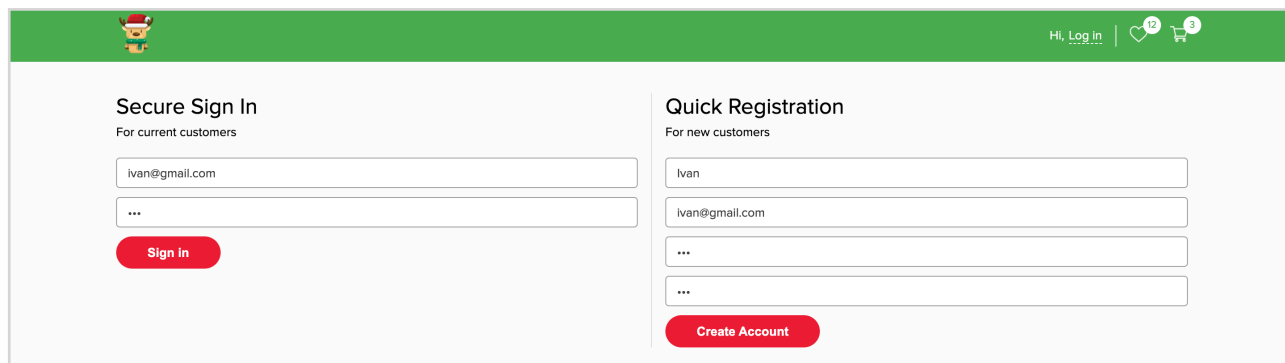
\$30000 -2%

, где:

1. Серым цветом выводится стоимость товара `price`.
  2. В зеленом блоке выводится процент скидки `salePercent`.
5. Вывод стоимости товара `price`.
    1. Если у данного товара свойство `sale: true`, то стоимость выводится с учетом скидки.
  6. Вывод иконки корзины:
    1. Если пользователь не залогинен, то код кнопки будет таким: `<button class="product__cart"></button>. При нажатии редиректим пользователя на страницу login.html

2. Если пользователь залогинен и товар ранее не был добавлен в массив shoppingCart, то:
  1. Код кнопки: <button class="product\_\_cart"></button>.
  2. При нажатии:
    1. Добавляем объект товара в массив shoppingCart в localStorage для данного юзера. **Вид объекта товара: {id: 1, count: 1}, где id – id товара, count – количество товара.**
    2. В шапке сайта в блоке с выводом количества добавленных в корзину товаров меняем значение.
3. Если пользователь залогинен и товар ранее был добавлен в массив shoppingCart, то:
  1. Код кнопки: <button class="product\_\_cart product\_\_cart--in"></button>.
  2. При нажатии:
    1. Удаляем объект товара с массива shoppingCart в localStorage для данного юзера.
    2. В шапке сайта в блоке с выводом количества добавленных в корзину товаров меняем значение.
4. Если пользователь залогинен и товар ранее был добавлен в массив orders, то:
  1. Код кнопки: <button class="product\_\_cart product\_\_cart--ordered"></button>.
  2. При нажатии ничего не происходит.

## Страница логина/регистрации login.html:



При заходе на страницу в `localStorage` ищем свойство `users`. Если такого свойства там нет, то создаем его и добавляем в него массив `USERS` с файла `js/script.js`.

### Форма Sign in:

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

При сабмите формы:

1. В `localStorage` в массиве `USERS` проверяем, есть ли пользователь с введенным email и password.
2. Если:
  1. Пользователь с данным email не существует в базе, то показываем блок с текстом Invalid email.



2. Пользователь с данным email существует в базе, но пароль не совпадает, то показываем блок с текстом Invalid password.



3. Пользователь с данным email существует в базе и пароль совпадает, то для данного юзера в `localStorage` в массиве `USERS` меняем свойство `status`: `true` и редиректим на страницу `account.html`.

## Форма Registration:

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

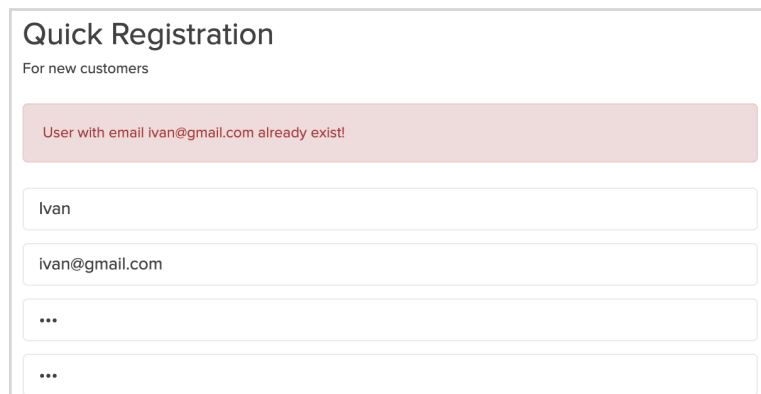
При сабмите формы:

1. Если значения полей password и verify password не совпадают, то показываем блок с текстом Password not matches!



The image shows a 'Quick Registration' form for new customers. It has a title 'Quick Registration' and a subtitle 'For new customers'. Below the subtitle is a red error message box that says 'Password not matches!'. The form contains four input fields: the first contains 'Ivan', the second contains 'ivan@gmail.com', and the third and fourth are empty and masked with three dots '...'. The form is styled with a light gray border and a white background.

2. В localStorage в массиве USERS проверяем, есть ли пользователь с введенным email.
3. Если:
  1. Пользователь с данным email существует в базе, то показываем блок с текстом User with email <user email> already exist!





The image shows a 'Quick Registration' form for new customers. It has a title 'Quick Registration' and a subtitle 'For new customers'. Below the subtitle is a red error message box that says 'User with email ivan@gmail.com already exist!'. The form contains four input fields: the first contains 'Ivan', the second contains 'ivan@gmail.com', and the third and fourth are empty and masked with three dots '...'. The form is styled with a light gray border and a white background.

2. Пользователь с данным email не существует в базе, то добавляем юзера в localStorage в массив USERS со значением свойства status: true и редиректим на страницу account.html.

## Страница аккаунта пользователя account.html:

The screenshot shows a user account page with a green header. The header contains a user avatar, the text "Hi, Ivan", a heart icon with "12", a shopping cart icon with "3", and a "Log out" link. The main content area is divided into two sections: "Ordered Items" and "My Info".

**Ordered Items**

Item Description	Quantity	Price	Sale	Total
 Cabriolet	3	\$900	- 25%	\$2025
 Commercial Plane	2	\$1000	-	\$2000

**My Info**

Name:

Email:

[Save changes](#) [Delete account](#)

1. Блок Ordered Items с информацией о завершенных заказах.
2. Блок My info с личной информацией юзера.

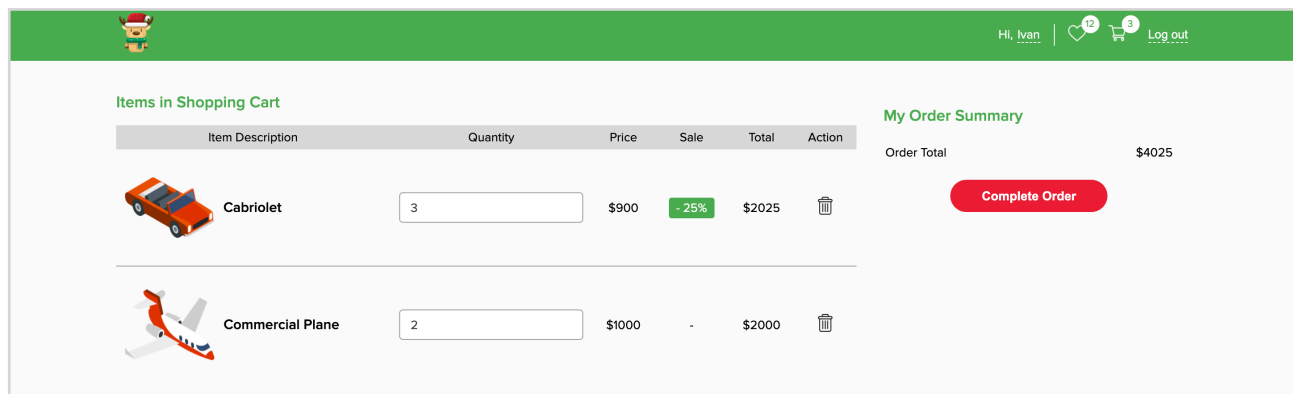
### Блок My orders с информацией о завершенных заказах

1. С localStorage с массива USERS для данного пользователя с массива orders выводим товары в таблицу id="orderTable".

### Блок My info с личной информацией юзера

1. Поле name. С localStorage с массива USERS выводим данные свойства name данного пользователя в input, что бы пользователь мог изменить значение.
2. Поле email. С localStorage с массива USERS выводим данные свойства email данного пользователя.
3. Кнопка Save changes:
  1. При сабмите формы My info:
    1. В localStorage с массива USERS для данного пользователя меняем значение свойства name.
4. Кнопка Delete account:
  1. Удаляем с localStorage с массива USERS данного пользователя;
  2. Пользователя редиректит на главную страницу index.html.

## Страница корзины пользователя shoppingCart.html:



1. Блок Items in Shopping Cart с информацией о не завершенных заказах пользователя.
2. Блок My Order Summary с информацией об общей сумме заказа пользователя.

### Блок Items in Shopping Cart с информацией о не завершенных заказах


1. С localStorage с массива USERS для данного пользователя с массива shoppingCart выводим данные в таблицу id="orderTable".
2. При изменении количества товара (Quantity):
  1. Пересчитываем поле Total для данного товара.
  2. Пересчитываем поле Order total в блоке My Order Summary.
  3. Меняем в localStorage в массиве USERS для данного пользователя в массиве shoppingCart свойство count в объекте данного товара.
3. При удалении товара (иконка корзины):
  1. Удаляем строку с информацией о данном товаре с таблицы.
  2. Пересчитываем поле Order total в блоке My Order Summary.
  3. Удаляем в localStorage в массиве USERS для данного пользователя в массиве shoppingCart весь объект данного товара.
  4. В шапке сайта в блоке с выводом количества добавленных в корзину товаров меняем значение.

### Блок My Order Summary с информацией об общей сумме заказа


1. Поле Order Total – общая сумма за все товары, добавленные в корзину.
2. Кнопка Complete Order. При нажатии на кнопку:
  1. В localStorage в массиве USERS для данного пользователя с массива shoppingCart удаляем все объекты и добавляем их в массив orders.
  2. В localStorage в массиве USERS для данного пользователя с массива favourites удаляем все id, которые УЖЕ НАХОДЯТСЯ в массиве orders. То есть, если пользователь уже купил данный товар, то он убирается с массива favourites.
  3. Редиректим пользователя на страницу account.html.




## Страница Избранное пользователя favourites.html



Hi, Ivan









12



3

Log out

Favourite Items

Item Description	Price	Sale	Total	Action
<div>Cabriolet</div>	\$900	- 25%	\$675	<div></div>
<div>Commercial Plane</div>	\$1000	-	\$1000	<div></div>

1. Блок Favourite Items с информацией о не завершенных заказах пользователя.

### Блок Favourite Items с информацией о не завершенных заказах

1. С localStorage с массива USERS для данного пользователя с массива favourites выводим данные в таблицу id="orderTable".
2. При нажатии на корзину:
  1. Если товар ранее не был добавлен в корзину:
    1. Добавляем кнопке класс **product\_\_cart—in** (фон кнопки будет зеленым).
    2. Добавляем в localStorage в массив USERS для данного пользователя в массив shoppingCart объект данного товара.
    3. В шапке сайта меняем количество добавленных в корзину товаров.
  2. Если товар ранее был добавлен в корзину:
    1. Удаляем кнопке класс **product\_\_cart—in** (фон кнопки будет красным).
    2. Удаляем в localStorage в массиве USERS для данного пользователя в массиве shoppingCart весь объект данного товара.
    3. В шапке сайта меняем количество добавленных в корзину товаров.
3. При нажатии на сердечко:
  1. Если товар ранее не был добавлен в корзину:
    1. Удаляем с таблицы строку товара.
    2. Удаляем в localStorage в массиве USERS для данного пользователя в массиве favourites объект данного товара.
    3. В шапке сайта меняем количество избранных товаров.