

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения вычислительной техники и автоматизированных
систем

Лабораторная работа №2
по дисциплине: Технологии web-программирования
тема: «**Клиентское программирование. Angular12**»

Выполнил: ст. группы ВТ-42
Бобылев М.С
Проверил: Картамышев С.В.

Белгород 2021 г.

Цель работы: познакомиться с основами разработки на языке JavaScript. Познакомится с основами разработки frontend приложения, в частности с использованием клиентских фреймворков (Angular). Научиться переносить вёрстку в отдельные компоненты.

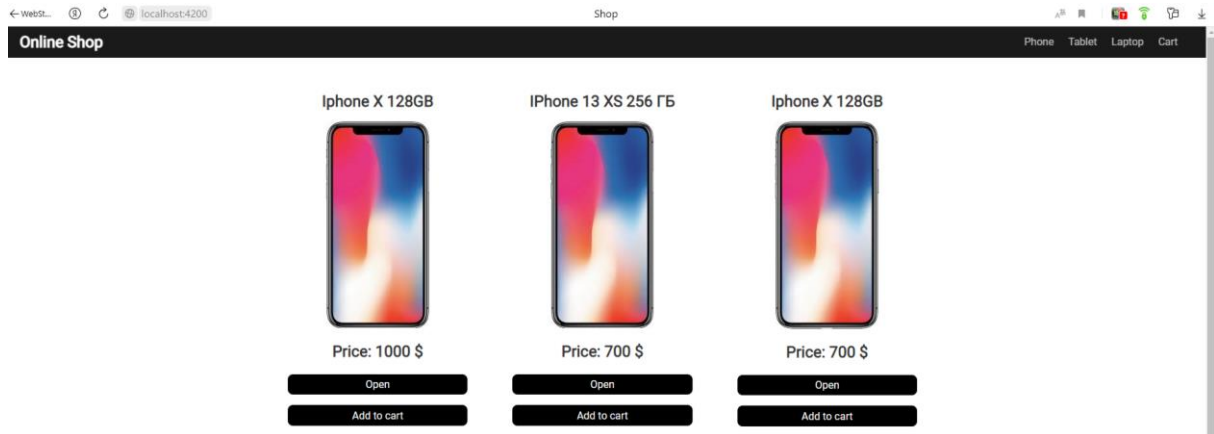
Задание к лабораторной работе:

1. Изучить основы разработки на языке JavaScript.
2. Изучить основы разработки frontend-приложения.
3. Развернуть базовое приложение на фреймворке Angular.
4. Добавить необходимые компоненты и перенести в них вёрстку, сделанную в прошлой лабораторной работе.
5. Продемонстрировать работу web-приложения.

Выполнение

В ходе выполнения работы были изучены основы разработки frontend-приложений на языке JavaScript, развернуто базовое приложение на фреймворке Angular, в него перенесены страницы из 1ой лабораторной работы, разделены на компоненты.

Первая страница, которую видит пользователь – страница входа.



Если у пользователя еще нет аккаунта, он может перейти на страницу с регистрацией

App ready

Online Shop

Login

Login as administrator

Email

Администратор

Enter valid Email

Password

Login

Основная страница со списком продуктов

Online Shop

DashboardAdd productOrdersLogout

Dashboard of products

Find a product...

ID	Name	Price	Date of creation	Open	Delete
1	Iphone X 128GB	1000	10/6/21, 4:51 PM	Open	Delete
2	IPhone 13 XS 256 GB	700	10/7/21, 4:55 PM	Open	Delete
3	Iphone X 128GB	700	10/6/21, 4:51 PM	Open	Delete
4	MacBook Pro 128GB	1100	10/6/21, 4:52 PM	Open	Delete
5	Ipad Pro 128GB	800	10/6/21, 4:53 PM	Open	Delete

Страница добавления продукта

Add product

Type

Name

Photo:

B I U G " " H₁ H₂ x₁ x¹ ¶ Normal : Normal : A Sans Serif :

Insert text here ...

Information:

B I U G " " H₁ H₂ x₁ x¹ ¶ Normal : Normal : A Sans Serif :

Insert text here ...

Price

Add

Реализовано модальное окно с добавлением задачи

Страница для доставки

Online Shop							Dashboard	Add product	Orders	Logout
Table of orders:										
Date	Phone	Name	Address	Products	Price	Status				
Oct 7, 2021, 4:08:41 PM	4235235	Максим		Iphone X 128GB	1000\$ Cash	Complited				
Oct 7, 2021, 4:43:47 PM	12334567	Анастасия		Iphone X 128GB Iphone 13 XS Ipad Pro 128GB MacBook Pro 128GB	3600\$ Card	Complited				
Oct 9, 2021, 9:52:35 AM	2141242145	Максим	Вокальная 34, кв 5	MacBook Pro 128GB	1100\$ Card	Complited				


Страница информации о товаре

Online Shop

PhoneTabletLaptopCart

Back

Iphone X 128GB



Price: 700 \$

Add to cart

Information:

The main

A type smartphone

Operating system Apple iOS

Operating system version iOS 13

Screen size 5.8"

Screen resolution 1125x2436

RAM 3 GB

Flash memory 128 GB

Integrated camera 12MP + 12MP

Number of main cameras 2

Number of matrix points 12 megapixels

SIM Card Number 1

SIM Card Format nano-sim, eSIM

Maximum video resolution 3840x2160 (60 fps)

CPU

CPU Apple A11 Bionic

CPU clock speed 2 390 MHz

Number of Cores 6 (2 + 4)

CPU architecture ARMv8-A (2x Monsoon + 4x Mistral)

CPU capacity 64 bits

Process technology 10 nm

Пример исходного кода

Product Component

```
<div *ngIf="product$ | async as product; else loading">
  <div class="product">
    <div class="header right">
      <button routerLink="/" class="btn btn-dark">Back</button>
    </div>

    <div class="flex">
      <div class="item text-center">
        <h2>{{ product.title }}</h2>

        <div class="flex-center">
          <quill-view-html [content]="product.photo"></quill-view-
html>

          </div>

          <div>
            <h2>Price: {{product.price}} $</h2>

            <button class="btn btn-dark" [routerLink]="['/cart']"
(click)="addProduct(product)">Add to cart</button>
          </div>
        </div>

        <div class="item">
          <h2>Information:</h2>

          <div>
            <quill-view-html [content]="product.info"></quill-view-
html>

            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

  <ng-template #loading>
    <p class="text-center">Loading...</p>
  </ng-template>
```

Main Page Component

```
<div *ngIf="products$ | async as products" class="dashboard">
  <app-product *ngFor="let product of products | sorting: productService.type
" class="box" [product]="product">
  </app-product>
</div>
```

Card Page Component

```
<div class="container" *ngIf="cartProducts.length > 0; else empty">
  <h2 class="text-center">Your order:</h2>

  <table>
    <tbody>
      <tr *ngFor="let cartProduct of cartProducts; let i = index">
        <td>{{ cartProduct.type }}</td>
        <td>{{ cartProduct.title }}</td>
        <td>
          <button class="btn btn-danger">
```

```

(click)="delete (cartProduct)">Delete</button>
      </td>
      <td>{{ cartProduct.price }} $</td>
    </tr>
  </tbody>
</table>
<hr>
<b class="right">Total: {{totalPrice}} $</b>

<h2 class="text-center">Delivery:</h2>

<div *ngIf="added" class="text-center text-danger">{{ added }}</div>

<form [formGroup]="form" (ngSubmit)="submit()">

  <div class="form-control">
    <label for="name">Name</label>
    <input id="name" type="text" formControlName="name">
    <div *ngIf="form.get('name').touched && form.get('name').invalid"
class="validation">
      <small *ngIf="form.get('name').errors.required">Enter your
name</small>
    </div>
  </div>

  <div class="form-control">
    <label for="phone">Phone number</label>
    <input id="phone" type="text" formControlName="phone">
    <div *ngIf="form.get('phone').touched &&
form.get('phone').invalid" class="validation">
      <small *ngIf="form.get('phone').errors.required">Enter your
phone number</small>
    </div>
  </div>

  <div class="form-control">
    <label for="address">Address</label>
    <input id="address" type="text" formControlName="address">
    <div *ngIf="form.get('address').touched &&
form.get('address').invalid" class="validation">
      <small *ngIf="form.get('address').errors.required">Enter your
address</small>
    </div>
  </div>

  <div class="form-control">
    <label for="payment">Payment method:</label>
    <select formControlName="payment">
      <option value="Cash">Cash</option>
      <option value="Card">Card</option>
    </select>
  </div>

  <button type="submit" class="btn btn-primary btn-block"
[disabled]="form.invalid || submitted">Add</button>

</form>
</div>

<ng-template #empty>
  <h2 class="text-center">Cart is empty, select product</h2>
</ng-template>

```


Login Page

```
<form [formGroup]="form" (ngSubmit)="submit()">

  <h2>Login as administrator</h2>

  <div class="form-control">
    <label for="email">Email</label>
    <input id="email" type="email" formControlName="email">
    <div *ngIf="form.get('email').touched && form.get('email').invalid"
class="validation">
      <small *ngIf="form.get('email').errors.required">Enter
Email</small>
      <small *ngIf="form.get('email').errors.email">Enter valid
Email</small>
    </div>
  </div>

  <div class="form-control">
    <label for="password">Password</label>
    <input id="password" type="password" formControlName="password">
    <div *ngIf="form.get('password').touched &&
form.get('password').invalid" class="validation">
      <small *ngIf="form.get('password').errors.required">Enter
Password</small>
      <small *ngIf="form.get('password').errors.minlength">
        Passwor must be at least
{{form.get('password').errors.minlength.requiredLength}} characters.
        Now he {{form.get('password').errors.minlength.actualLength}}
      </small>
    </div>
  </div>

  <button type="submit" class="btn btn-primary btn-block"
[disabled]="form.invalid || submitted">Login</button>

</form>
```

AddPage

```
<form [formGroup]="form" (ngSubmit)="submit()">

  <h2 class="text-center">Add product</h2>

  <div class="form-control">
    <label for="type">Type</label>
    <select formControlName="type">
      <option value="Phone">Phone</option>
      <option value="Tablet">Tablet</option>
      <option value="Laptop">Laptop</option>
    </select>
    <div *ngIf="form.get('type').touched && form.get('type').invalid"
class="validation">
      <small *ngIf="form.get('type').errors.required">Chose product
type</small>
    </div>
  </div>

  <div class="form-control">
    <label for="title">Name</label>


```

```
<input id="title" type="title" formControlName="title">
  <div *ngIf="form.get('title').touched && form.get('title').invalid"
class="validation">
    <small *ngIf="form.get('title').errors.required">Enter product
name</small>
  </div>
</div>
<div class="form-control">
  <label for="type">Photo:</label>
  <quill-editor formControlName="photo"></quill-editor>

</div>
<div class="form-control">
  <label for="type">Information:</label>
  <quill-editor formControlName="info"></quill-editor>

</div>
<div class="form-control">
  <label for="price">Price</label>
  <input id="price" type="price" formControlName="price">
  <div *ngIf="form.get('price').touched && form.get('price').invalid"
class="validation">
    <small *ngIf="form.get('price').errors.required">Enter product
price</small>
  </div>
</div>

  <button type="submit" class="btn btn-primary btn-block"
[disabled]="form.invalid || submitted">Add</button>

</form>
```