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 №6

тема: «Работа с HTTP запросами»

дисциплина:

Технологии WEB-программирования

Выполнила:

ст. группы ПВ-41

Забелина Е.А.

Проверил: Картамышев С. В.

Белгород 2022 г.

**Цель**: изучить принципы работы отправки Ajax запросов. Изучить работу отправки синхронных и асинхронных запросов в Vue JS. Получить навыки работы с промисами.

**Задание к лабораторной работе:**

1. Изучить возможности Vue js для отправки http запросов.

2. Выбрать подходящую библиотеку для работы с запросами.

3. Реализовать взаимодействие фронтенда с REST API, спроектированном

в прошлой лабораторной работе.

4. Продемонстрировать работу взаимодействия фронтенд приложения с

REST API.

**Код взаимодействия фронтенд приложения с REST API**

**Api.php**

<?php  
  
use App\Http\Controllers\UserController;  
use Illuminate\Http\Request;  
use Illuminate\Support\Facades\Route;  
use \App\Http\Controllers\ProductController;  
  
/\*  
|--------------------------------------------------------------------------  
| API Routes  
|--------------------------------------------------------------------------  
|  
| Here is where you can register API routes for your application. These  
| routes are loaded by the RouteServiceProvider within a group which  
| is assigned the "api" middleware group. Enjoy building your API!  
|  
\*/  
  
Route::middleware('auth:sanctum')->get('/user', function (Request $request) {  
 return $request->user();  
});  
  
Route::prefix('products')->group(function () {  
 Route::get('/', [ProductController::class,'list']);  
});  
  
Route::prefix('user')->group(function(){  
 Route::post('/', [UserController::class, 'authorization']);  
});  
  
Route::prefix('main\_products')->group(function () {  
 Route::get('/', [ProductController::class,'mainProducts']);  
});  
  
Route::prefix('products1')->group(function () {  
 Route::get('/', [ProductController::class,'products1']);  
});  
  
Route::prefix('products2')->group(function () {  
 Route::get('/', [ProductController::class,'products2']);  
});  
  
Route::prefix('products3')->group(function () {  
 Route::get('/', [ProductController::class,'products3']);  
});  
  
Route::prefix('cart')->group(function(){  
 Route::get('/{userId}', [\App\Http\Controllers\CartController::class, 'info']);  
 Route::post('/', [\App\Http\Controllers\CartController::class, 'addToCart']);  
});

**Login**

authorization(){  
 //тут добавление в бд корзины пользователя сделать  
 const params = {  
 name: this.username, password: this.password  
 }  
 this.$http.post('/user', params)  
 .then(response => this.checkCorrect(response.data.userId))  
 },  
 checkCorrect(data){  
 if(data !== -1)  
 {  
 this.userId = data  
 this.correct = true  
 this.$router.push('/')  
 this.info = "Success"  
 ***localStorage***.userId = this.userId;  
 }  
 else  
 {  
 this.info = "Wrong data"  
 }  
 ***console***.log(this.correct)  
 }  
},  
mounted(){  
 if (***localStorage***.userId) {  
 this.userId = ***localStorage***.userId;  
 }  
}

**Cart**

data() {  
 return {  
 cart: []  
 }  
 },  
 created(){  
 this.$http.get('/cart/' + ***localStorage***.userId)  
 .then(response => this.cart = response.data)  
 }  
}

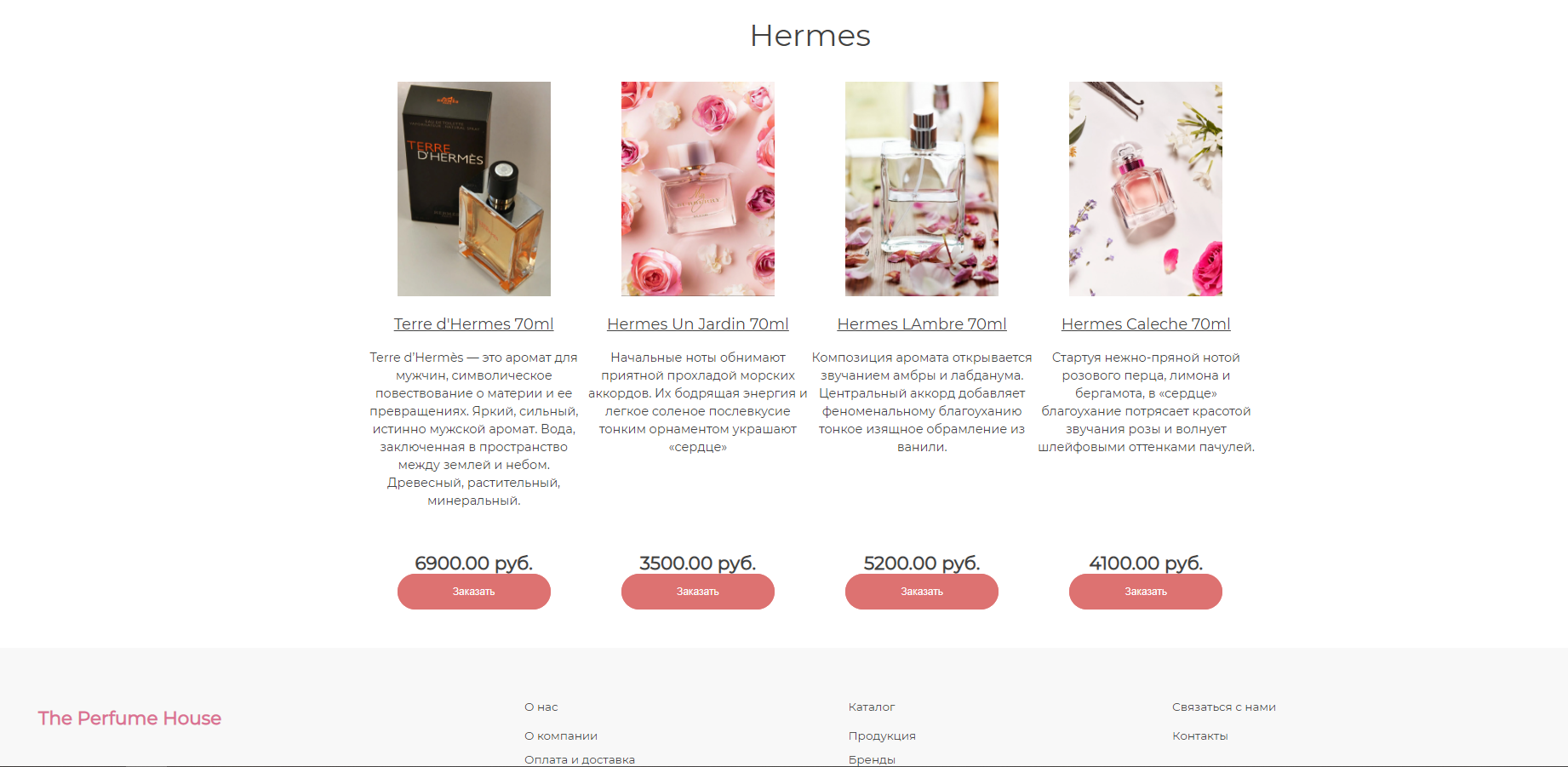
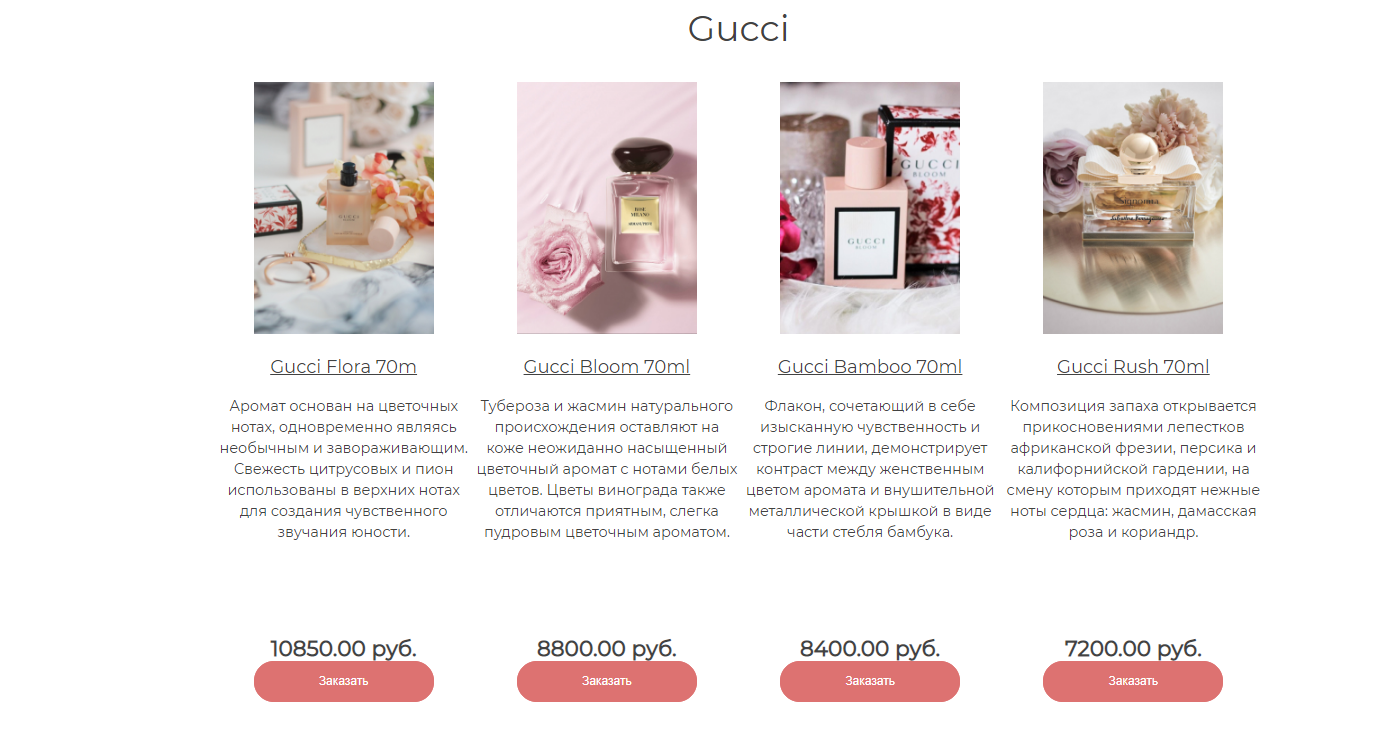
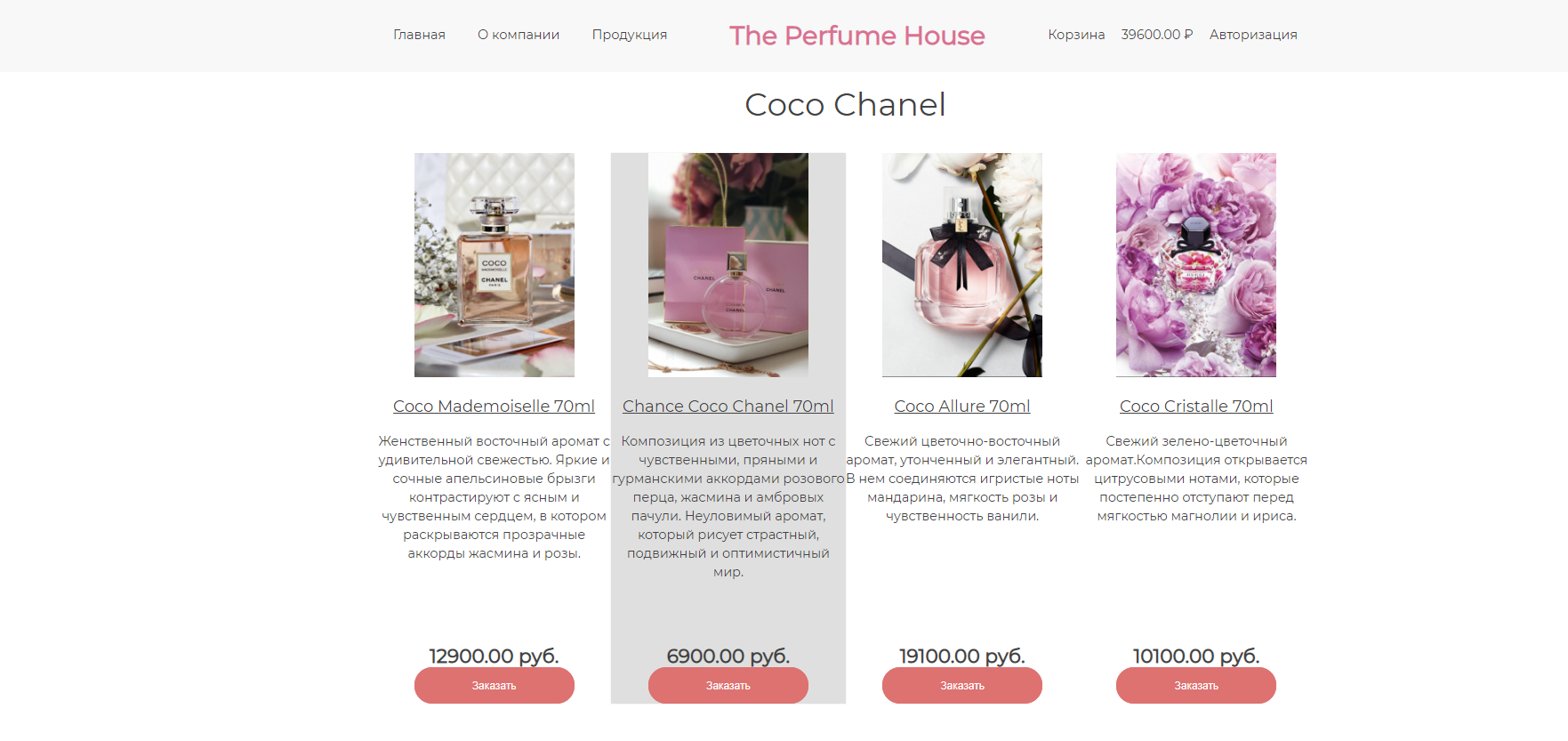
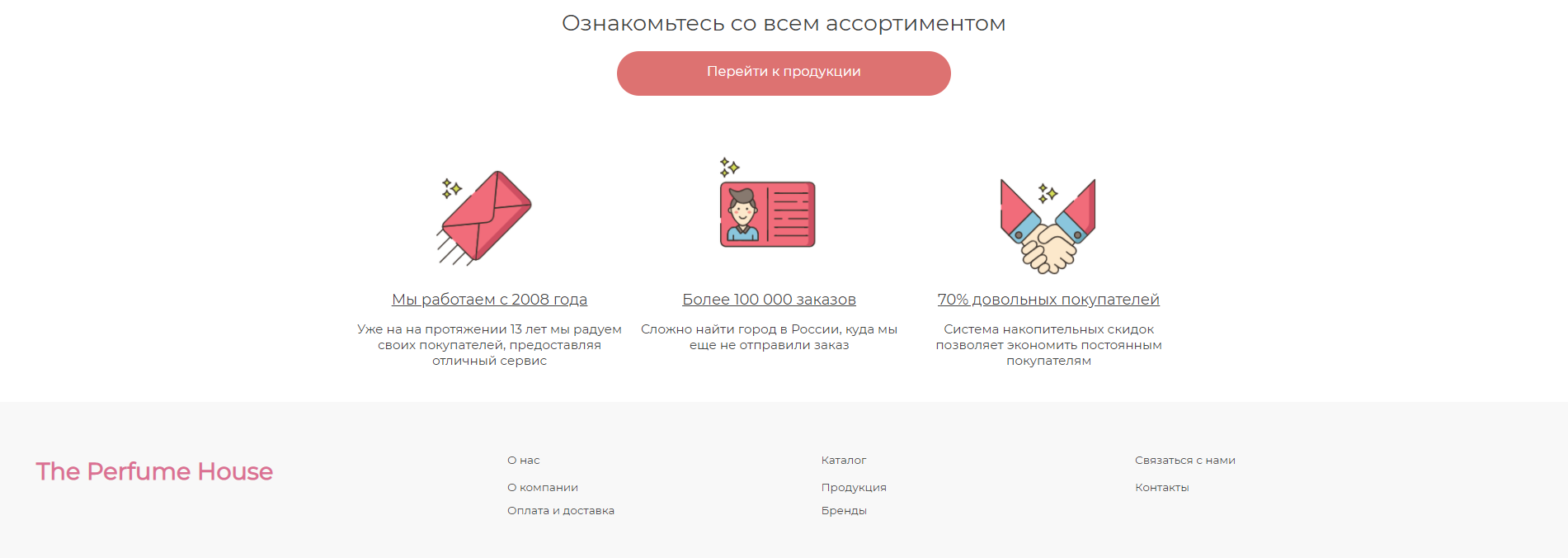
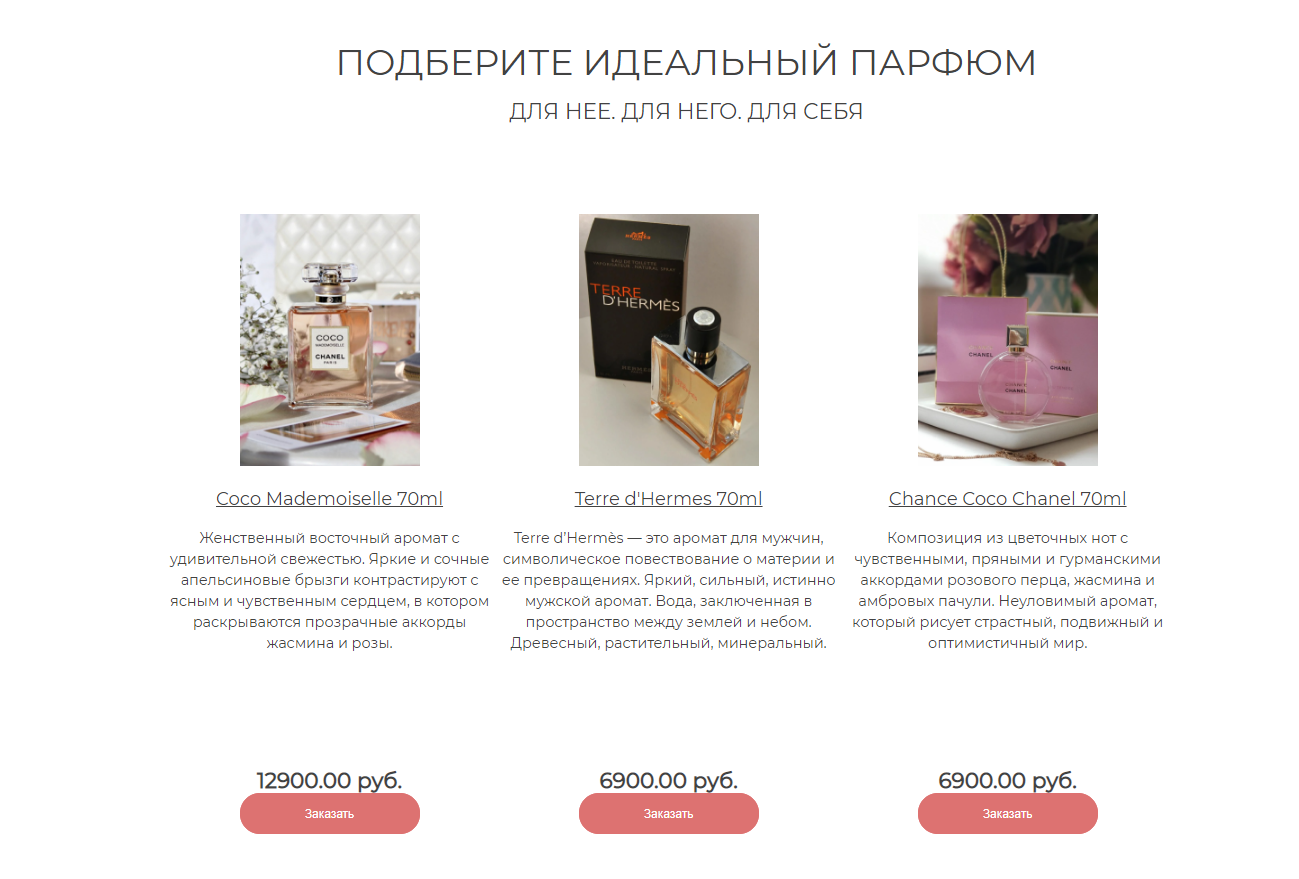
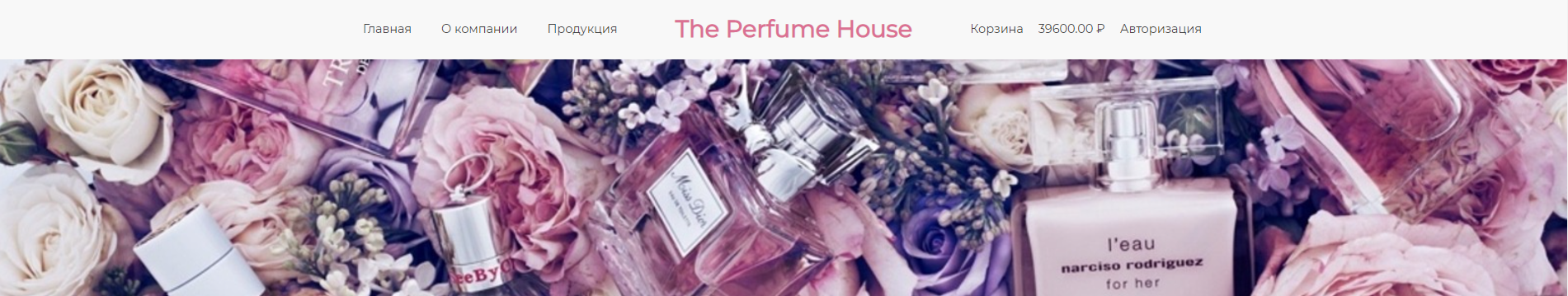
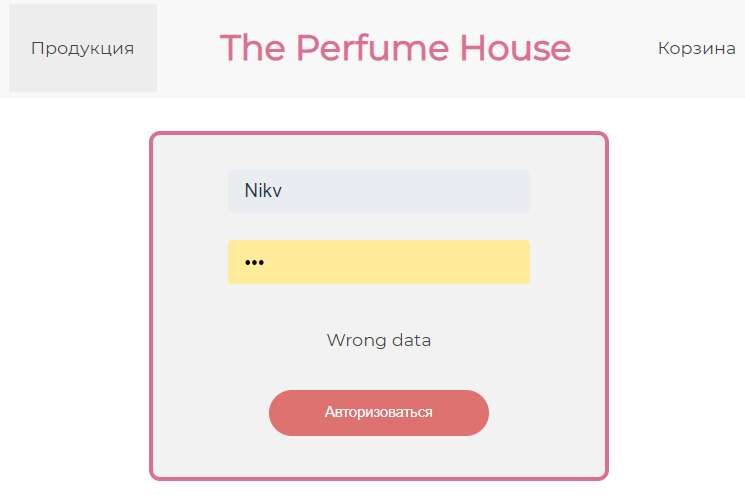
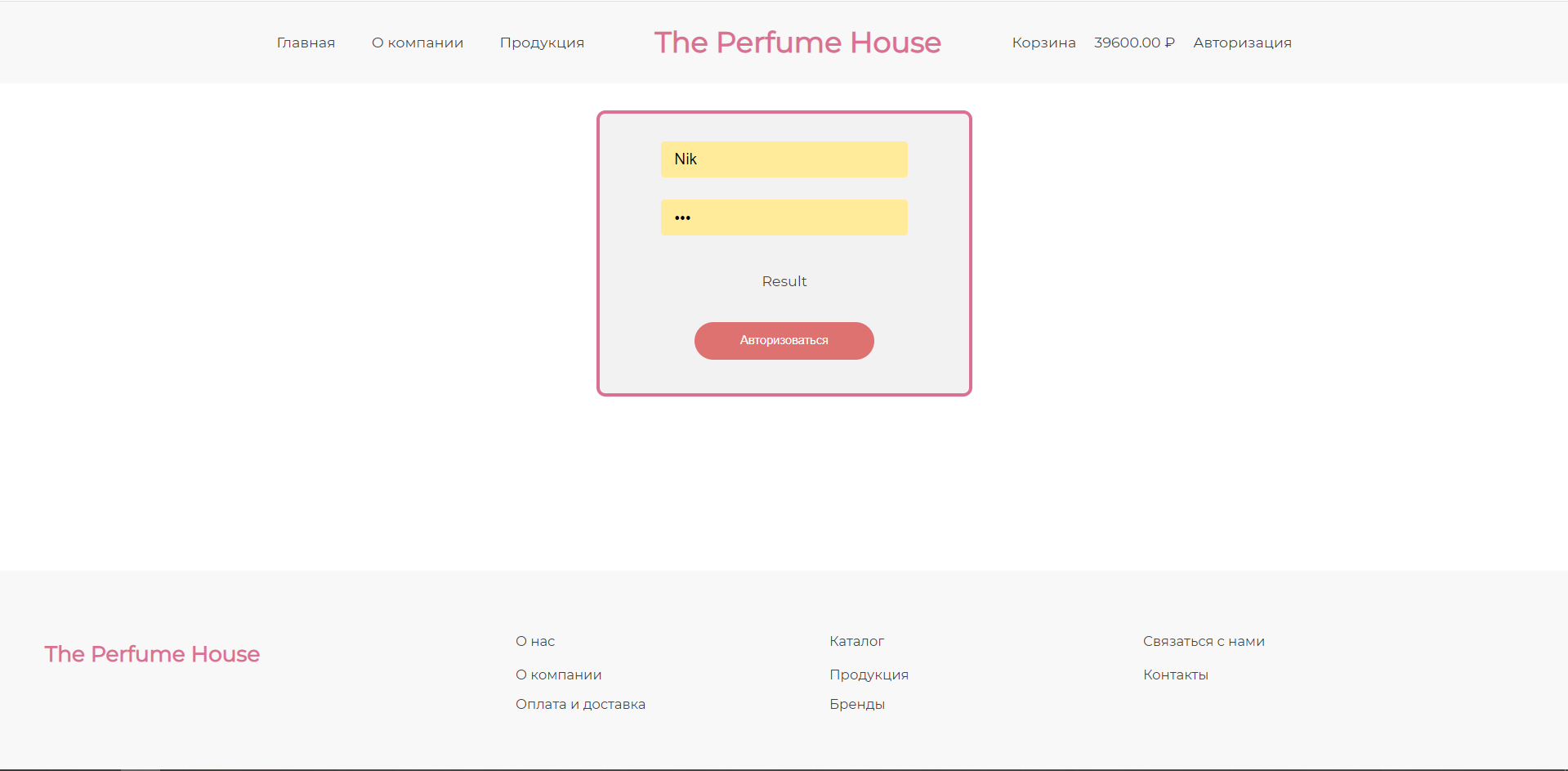
**ProductPage**

data() {  
 return {  
 products1: [],  
 products2: [],  
 products3: []  
 }  
 },  
  
 created() {  
 this.$http.get('/products1')  
 .then(response => this.products1 = response.data)  
 this.$http.get('/products2')  
 .then(response => this.products2 = response.data)  
 this.$http.get('/products3')  
 .then(response => this.products3 = response.data)  
 }  
}

**Home**

created() {  
 this.$http.get('/main\_products')  
 .then(response => this.products = response.data)  
}

**Результаты работы приложения**



**GITHUB:** [**https://github.com/SerpensHell/Web**](https://github.com/SerpensHell/Web%20)