**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**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  
дисциплина: Технологии Web-программирования  
тема: «Работа с HTTP запросами»

Выполнила: ст. группы ПВ-41  
Немшилова А.Г.  
Проверил: Картамышев С.В.

Белгород 2020

**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.

**Выполнение работы:**

Было реализовано следующее взаимодействие фронтенда с REST API:

Получение продукта по его url (Product.vue):

created() {  
 const url = this.$route.params.url  
  
 this.$http.get('/product/url', {params: {url: url}})  
 .then((response) => {***console***.log(response.data)  
 this.product = response.data})  
},

Получение всех продуктов, их категорий и начинок (ProductList.vue):

created() {  
 this.$http.get('/product/all')  
 .then((response) => this.items = response.data)  
  
 this.$http.get('/category/all')  
 .then((response) => this.filter.categories = response.data)  
  
 this.$http.get('/filling/all')  
 .then((response) => this.filter.fillings = response.data)  
},

Фильтрация по определенным категориям продуктов (ProductList.vue):

checkCategory(){  
 var category;  
 this.filter.categories.forEach(function(item, i) {  
 var x = ***document***.getElementById(item.name);  
 if (x.checked) {  
 category = item.id;  
 }  
 });  
  
 this.$http.get('/product/idcategory', {params: {idCategory: category}})  
 .then((response) => {***console***.log(response.data)  
 this.items = [...response.data]})  
  
 var filling;  
 this.filter.fillings.forEach(function(item, i) {  
 var x = ***document***.getElementById(item.name);  
 if (x.checked) {  
 filling = item.id;  
 }  
 });  
  
 this.$http.get('/product/idfilling', {params: {idFilling: filling}})  
 .then((response) => {***console***.log(response.data)  
 this.items = [...response.data]})  
  
 }  
}

Получение продуктов, которые находятся в «Хитах продуктов» на главной странице (Home.vue):

created() {  
 this.$http.get('/product/hitproducts')  
 .then((response) => {  
 ***console***.log(response.data)  
 this.items = [...response.data]  
 })  
}

Для этого в контроллере продукта был добавлен метод:

public function actionHitproducts() {  
 return Product::findAll(['hitProduct' => *1*]);  
}

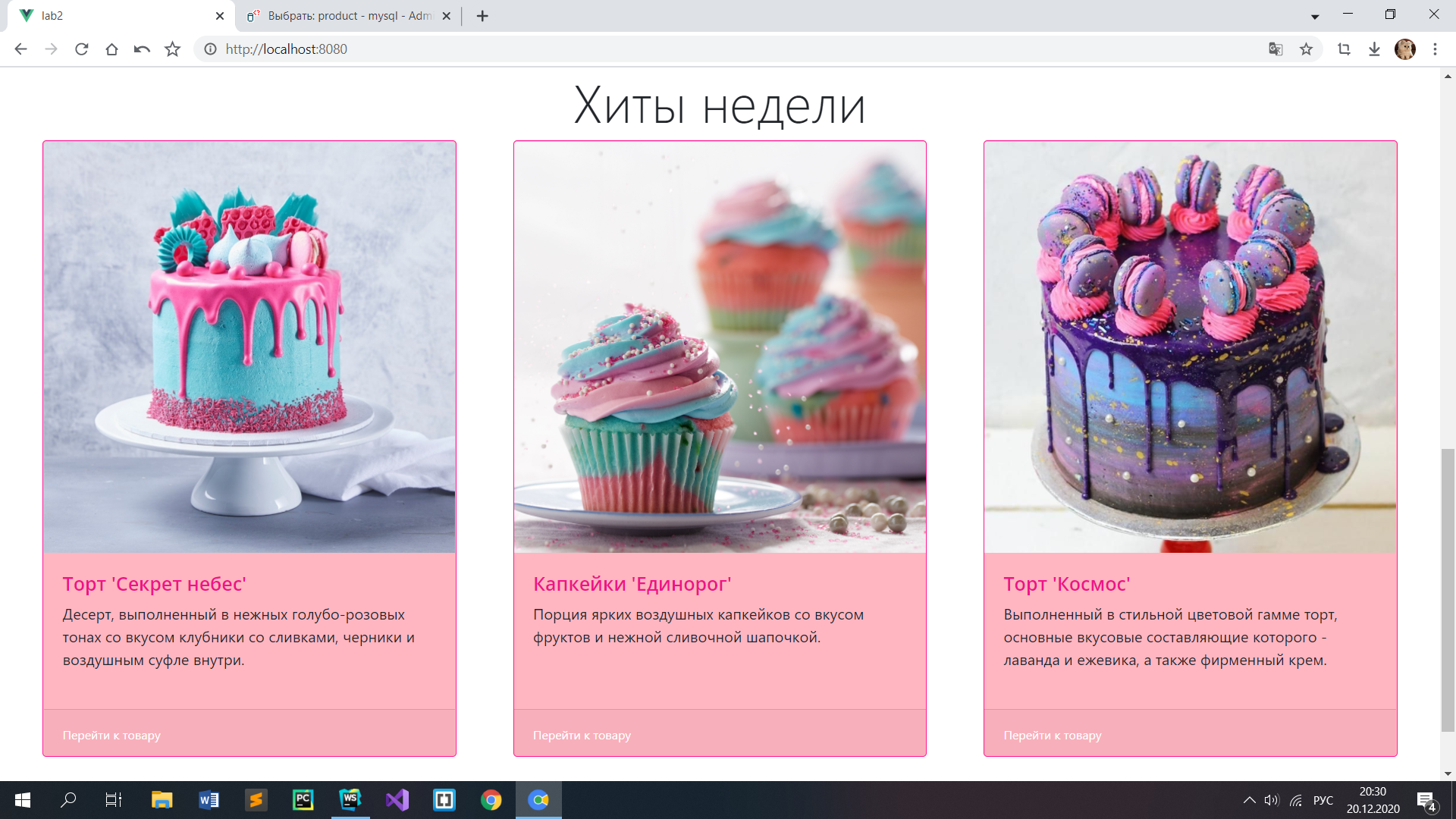
Авторизация пользователя:

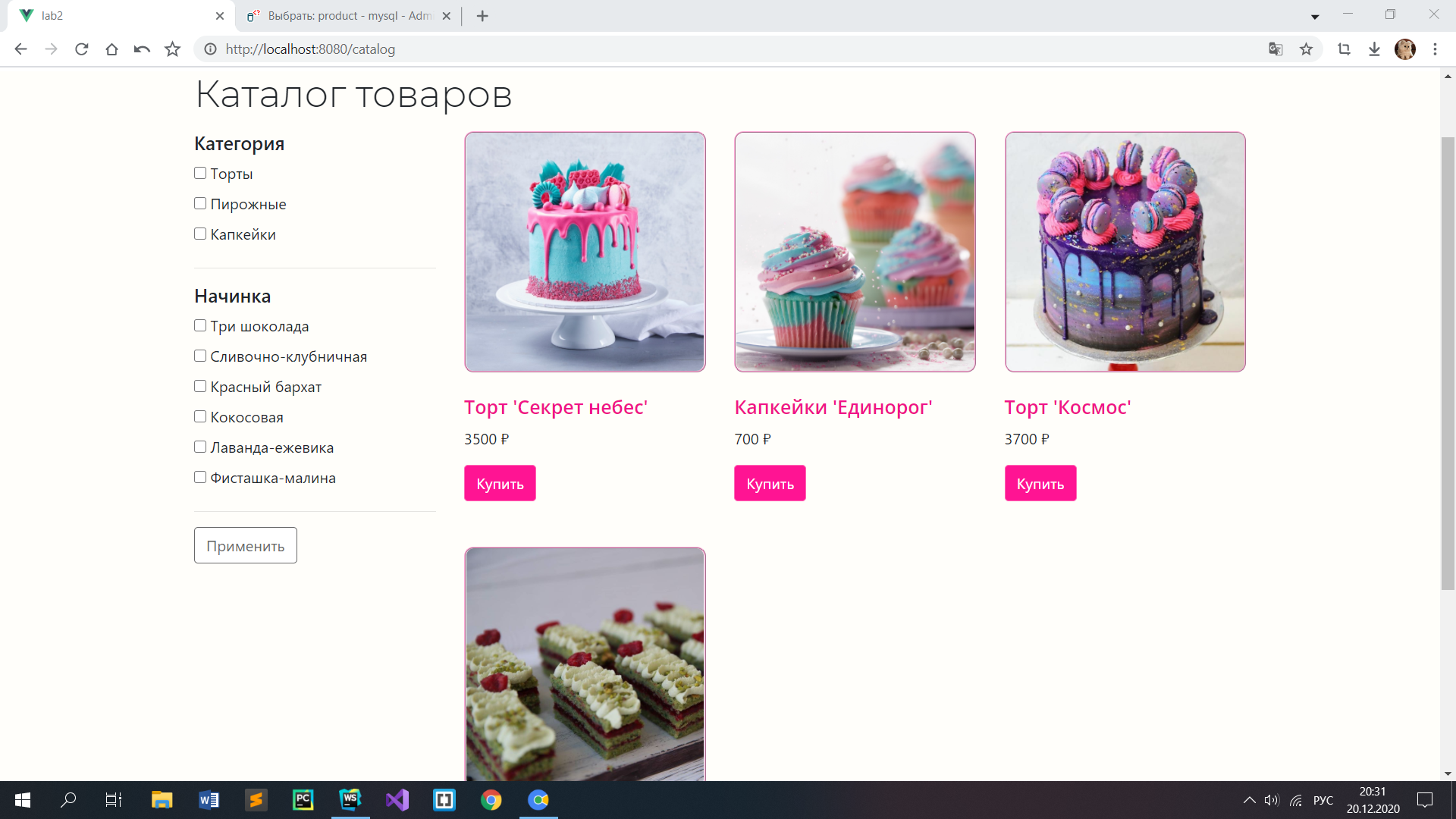
handleSubmit: function (e) {  
 e.preventDefault()  
  
 const submissionData = {  
 username: this.username,  
 password: this.password,  
 }  
  
 this.$http.post('/user/login', submissionData)  
 .then((response) => {  
 ***localStorage***.setItem('userData', ***JSON***.stringify(response.data))  
 if (response.data.name != undefined){  
 ***console***.log(response.data.name)  
 ***router***.push({ name: 'Profile' })  
 }  
 else{  
 alert("Неправильно введен логин или пароль")  
 }  
 })  
 .catch((error) => alert(error.response.data.statusText))  
}

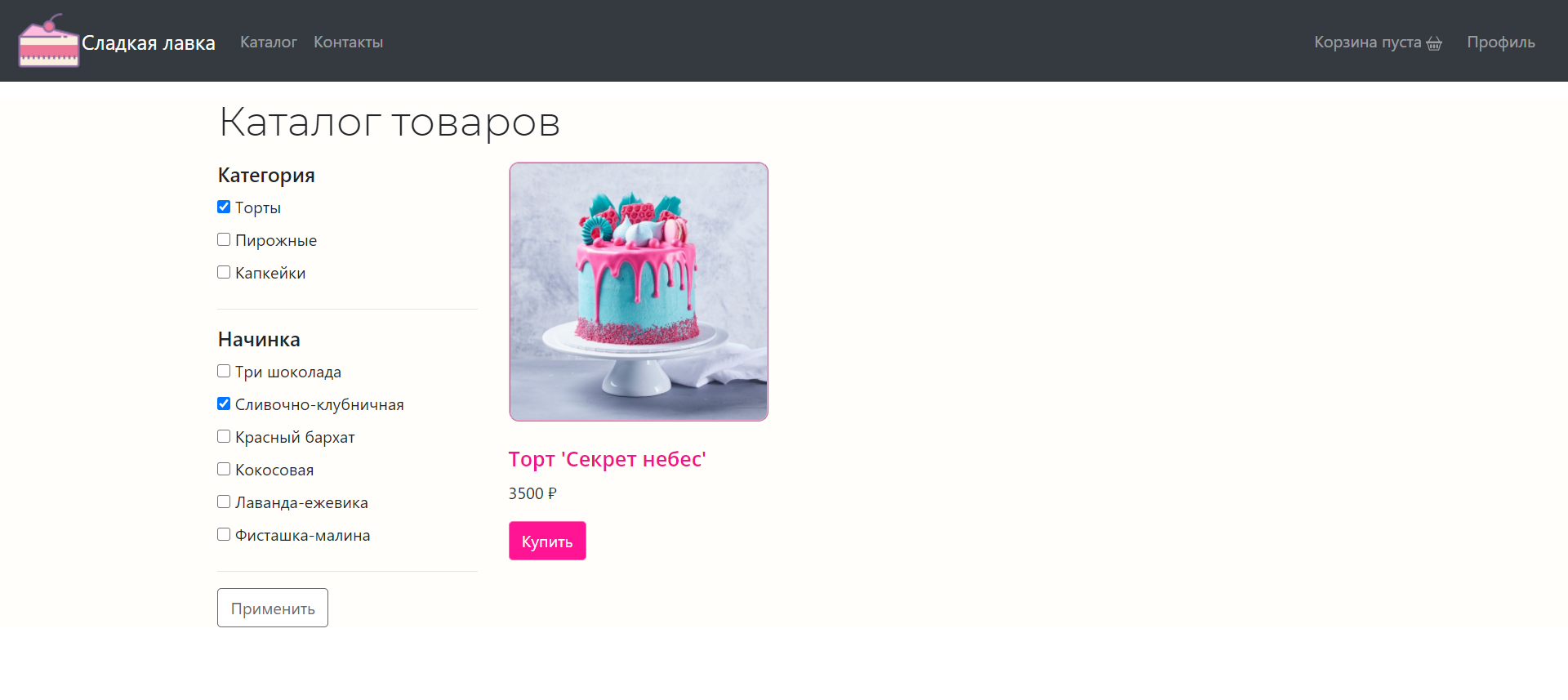
Регистрация пользователя:

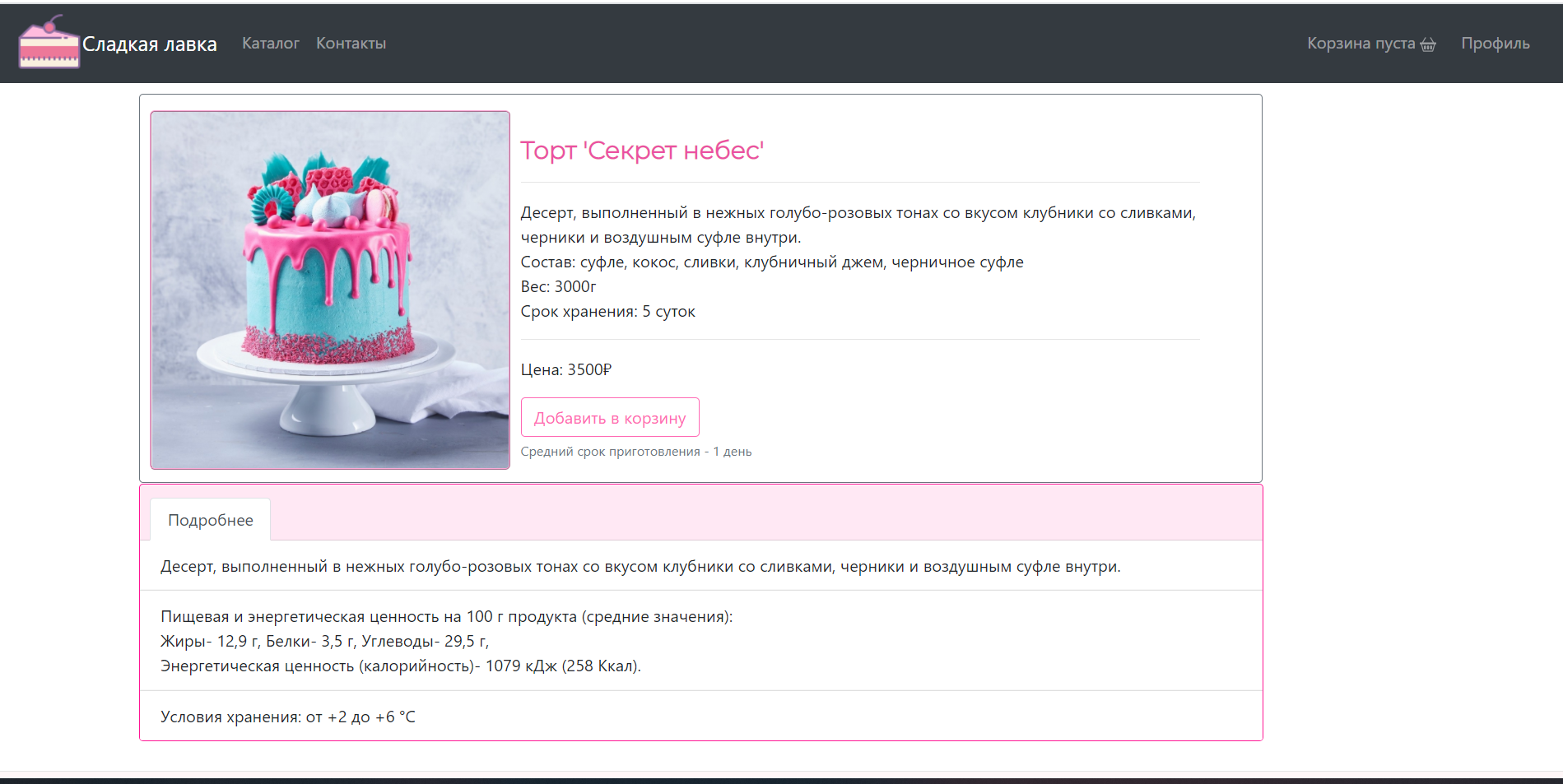
handleSubmit: function (e) {  
 e.preventDefault()  
 if (this.password !== this.repeatPassword) {  
 alert('Пароли не совпадают')  
 return  
 }  
 const submissionData = {  
 username: this.username,  
 password: this.password,  
 }  
 this.$http.post('/user/register', submissionData)  
 .then((response) => {  
 ***localStorage***.setItem('userData', ***JSON***.stringify(response.data))  
 ***router***.push({ name: 'Profile' })  
 })  
 .catch((error) => alert(error.response.data.statusText))  
}

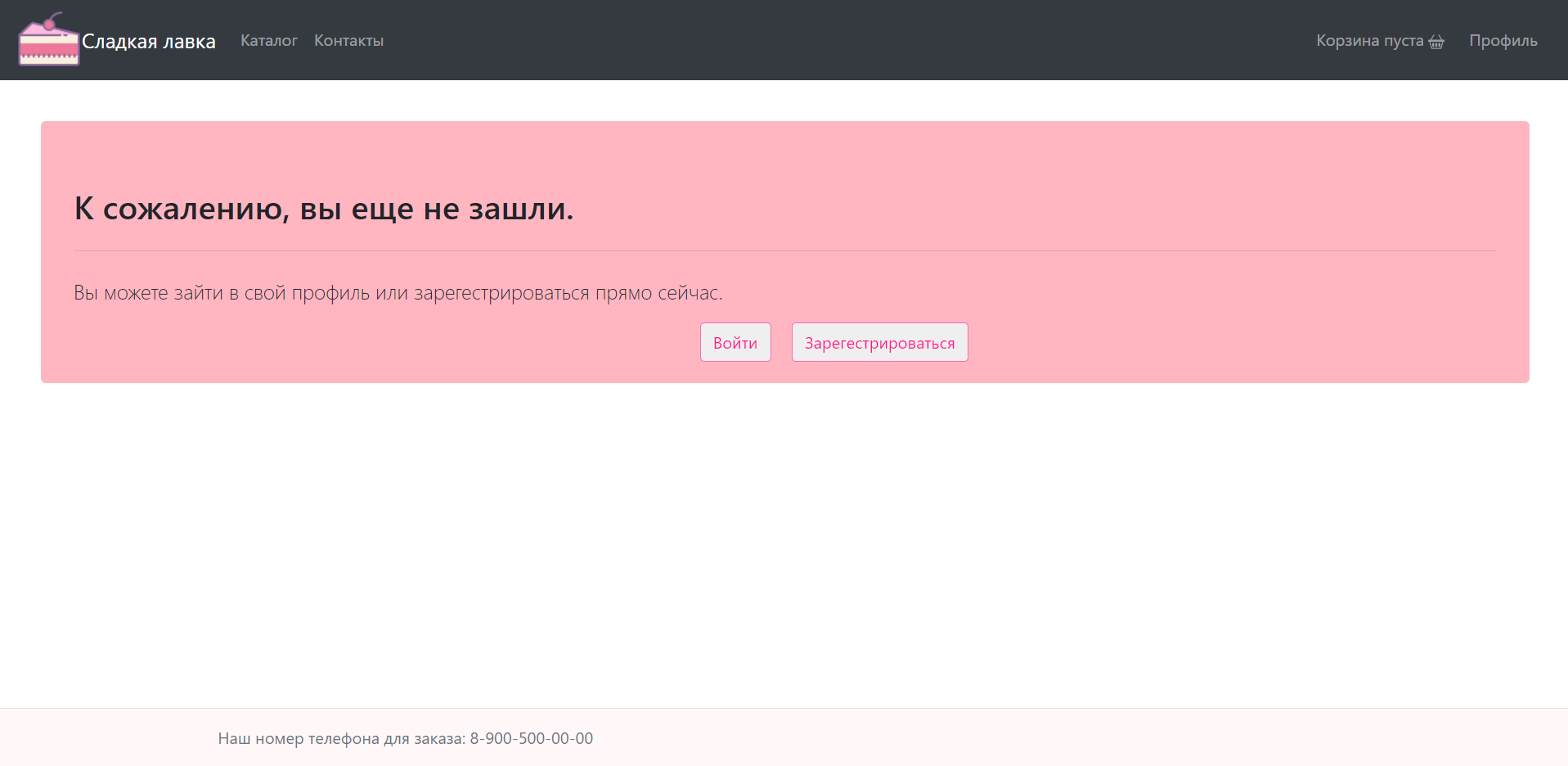
Полученный сайт:

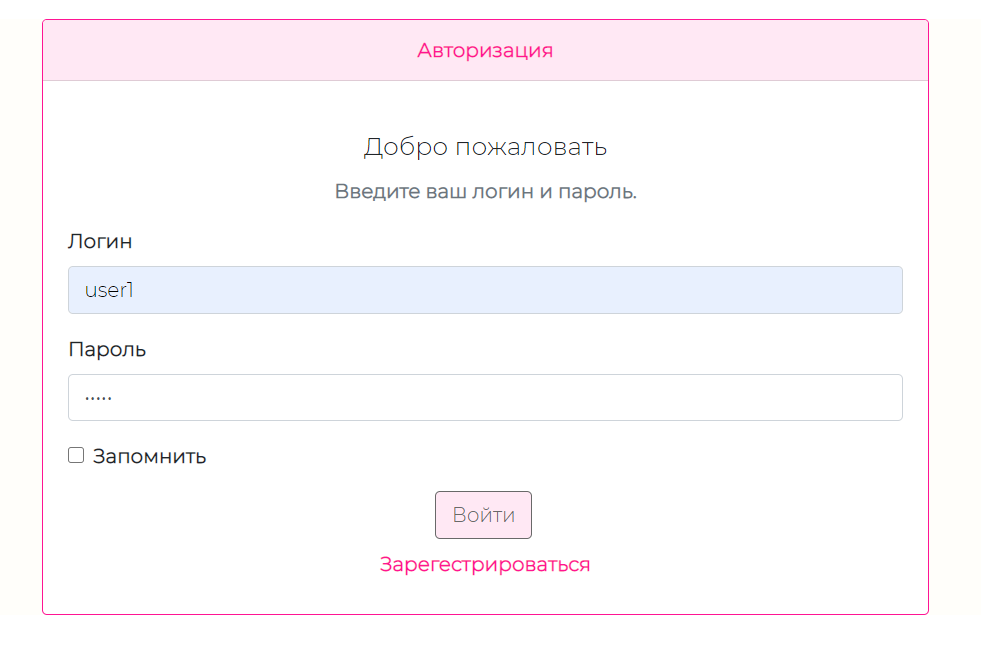


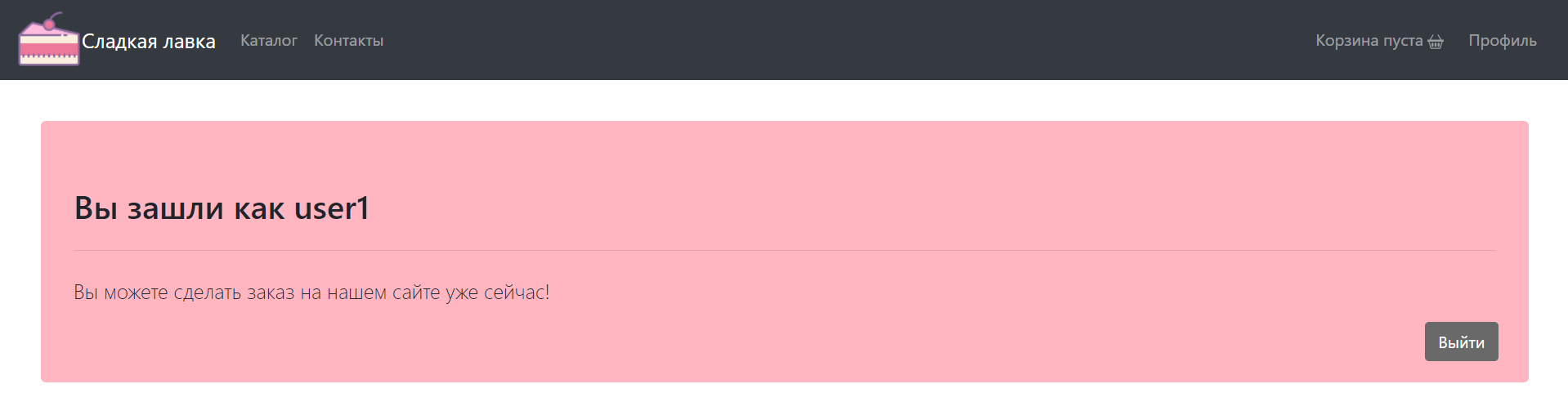


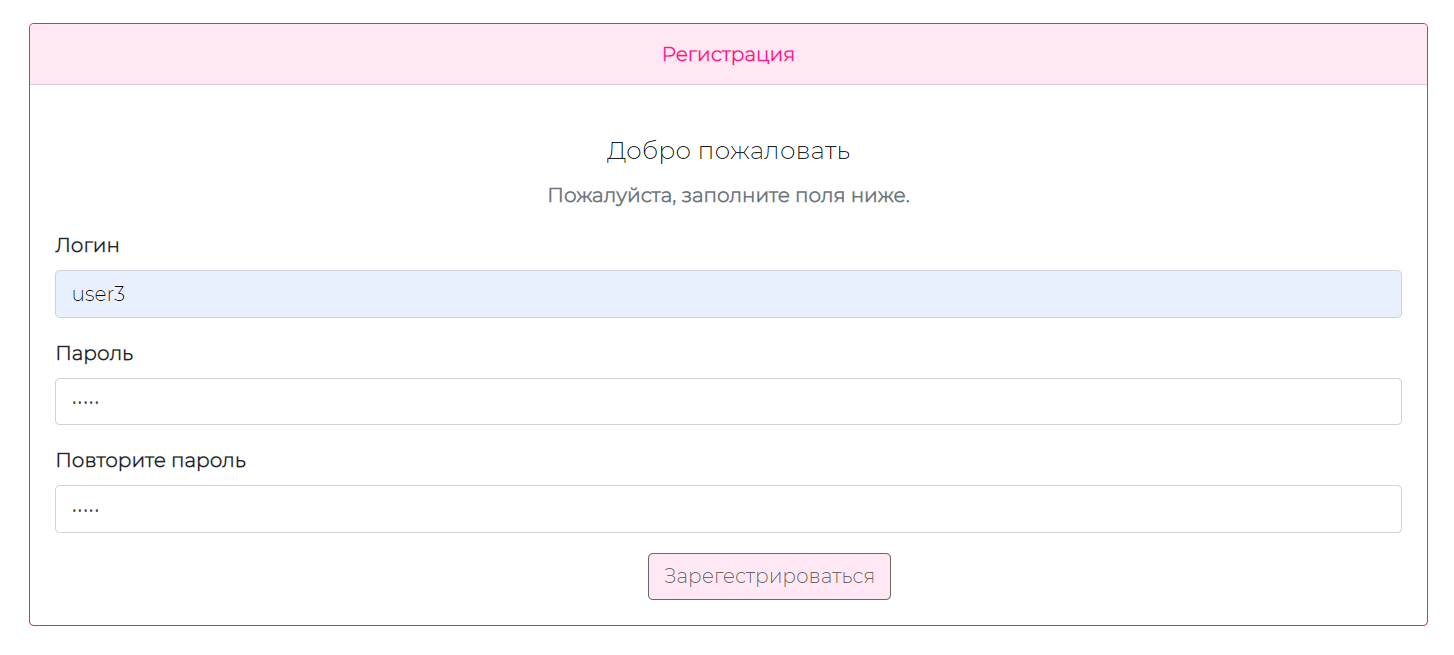












**Вывод:** в ходе выполнения лабораторной работы изучили принципы работы отправки Ajax запросов. Изучили работу отправки синхронных и асинхронных запросов в Vue JS. Получили навыки работы с промисами.