**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  
Ярыгина И. Ю.  
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:**

Авторизация пользователя:

signIn() {  
 const formData = {  
 username: this.email,  
 password: this.password,  
 grant\_type: 'password'  
 };  
 this.$http.post('/oauth/token', ***querystring***.stringify(formData), {  
 headers: {  
 'authorization': 'Basic dWk6bXJJVG1ndnpSWE9a',  
 "Access-Control-Allow-Origin": "\*",  
 'Content-Type': 'application/x-www-form-urlencoded',  
 "cache-control": "no-cache"  
 // 'Accept-Encoding': 'br'  
 }, baseURL: 'http://localhost:8080/',  
 })  
 .then((response) =>  
 {***console***.log(response.data)  
 ***localStorage***.setItem('access\_token', ***JSON***.stringify(response.data.access\_token))  
 ***localStorage***.setItem('user\_id', ***JSON***.stringify(response.data.user\_id))  
 this.$router.push({ name: 'Profile' })  
 })  
 .catch((error) => ***console***.log(error))  
},

Получение авторизованного пользователя:

created() {  
 let userId = ***localStorage***.getItem('user\_id')  
 userId = userId.substring(1, userId.length - 1)  
 ***console***.log(userId)  
 this.$http.get(`/v1/user/${userId}`)  
 .then((response) => {  
 ***console***.log(response.data)  
 this.user = {...response.data}  
 })  
}

Регистрация:

handleSubmit: function (e) {  
 e.preventDefault()  
 if (this.inputPassword !== this.inputPasswordRepeat) {  
 alert('Пароли не совпадают')  
 return  
 }  
 const result = {  
 firstName: this.formName,  
 secondName: this.formSecondName,  
 patronymic: '',  
 authority: 'USER',  
 username: this.inputEmail,  
 password: this.inputPassword,  
 }  
 ***console***.log(result)  
 this.$http.post('/register', result)  
 .then(() => {  
 ***router***.push({name: 'SignIn'})  
 })  
 .catch((error) => alert(error.response.data.statusText))  
},

Получение всех фильмов:

created() {  
 this.$http.get('/v1/film')  
 .then((response) => {  
 ***console***.log(response.data)  
 this.items = [...response.data]  
 })  
}

Получение фильма по id:

created() {  
 let id = this.$route.params.id;  
 this.$http.get(`/v1/film/${id}`)  
 .then((response) => {  
 ***console***.log(response.data)  
 this.film = response.data  
 })  
},

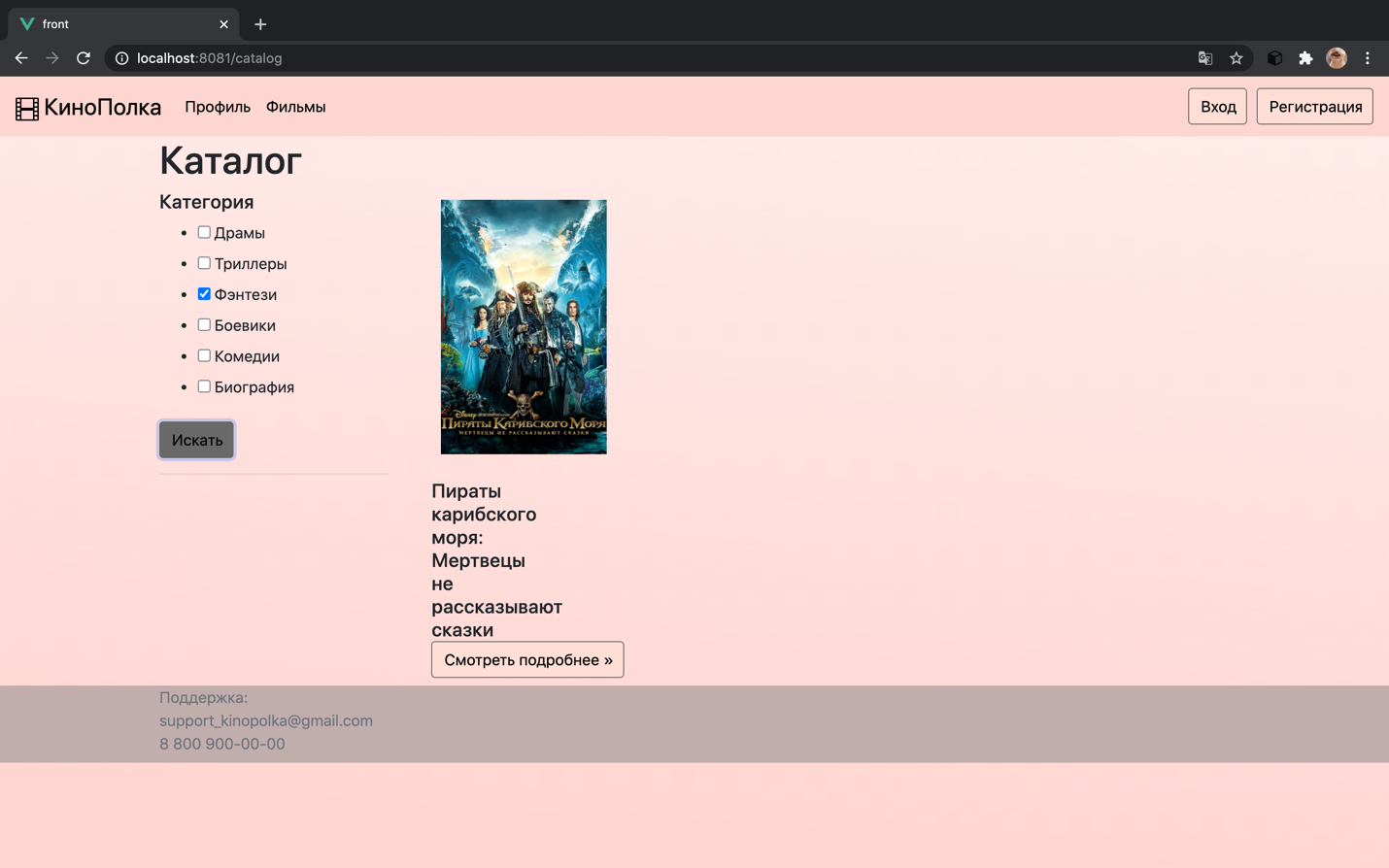
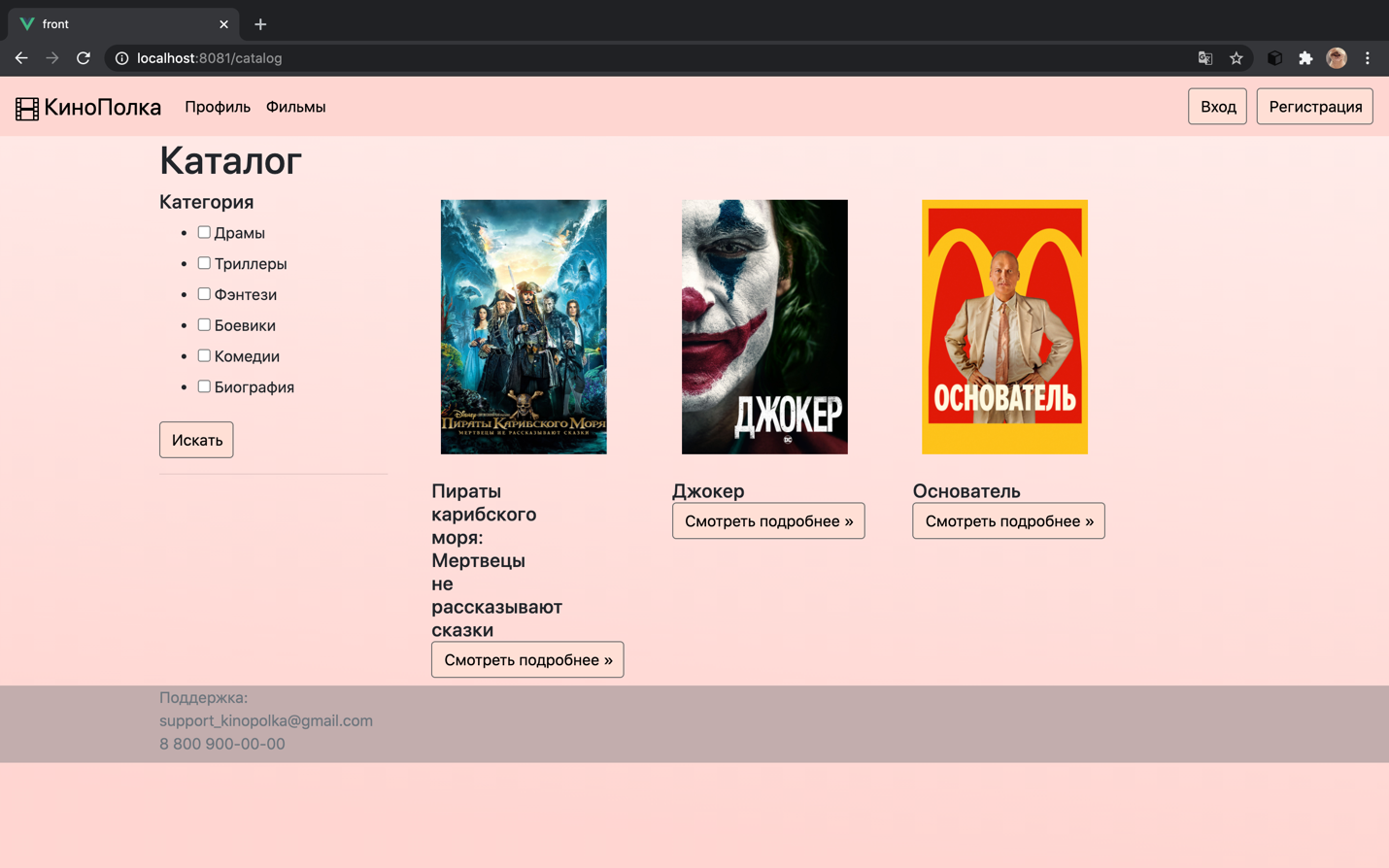
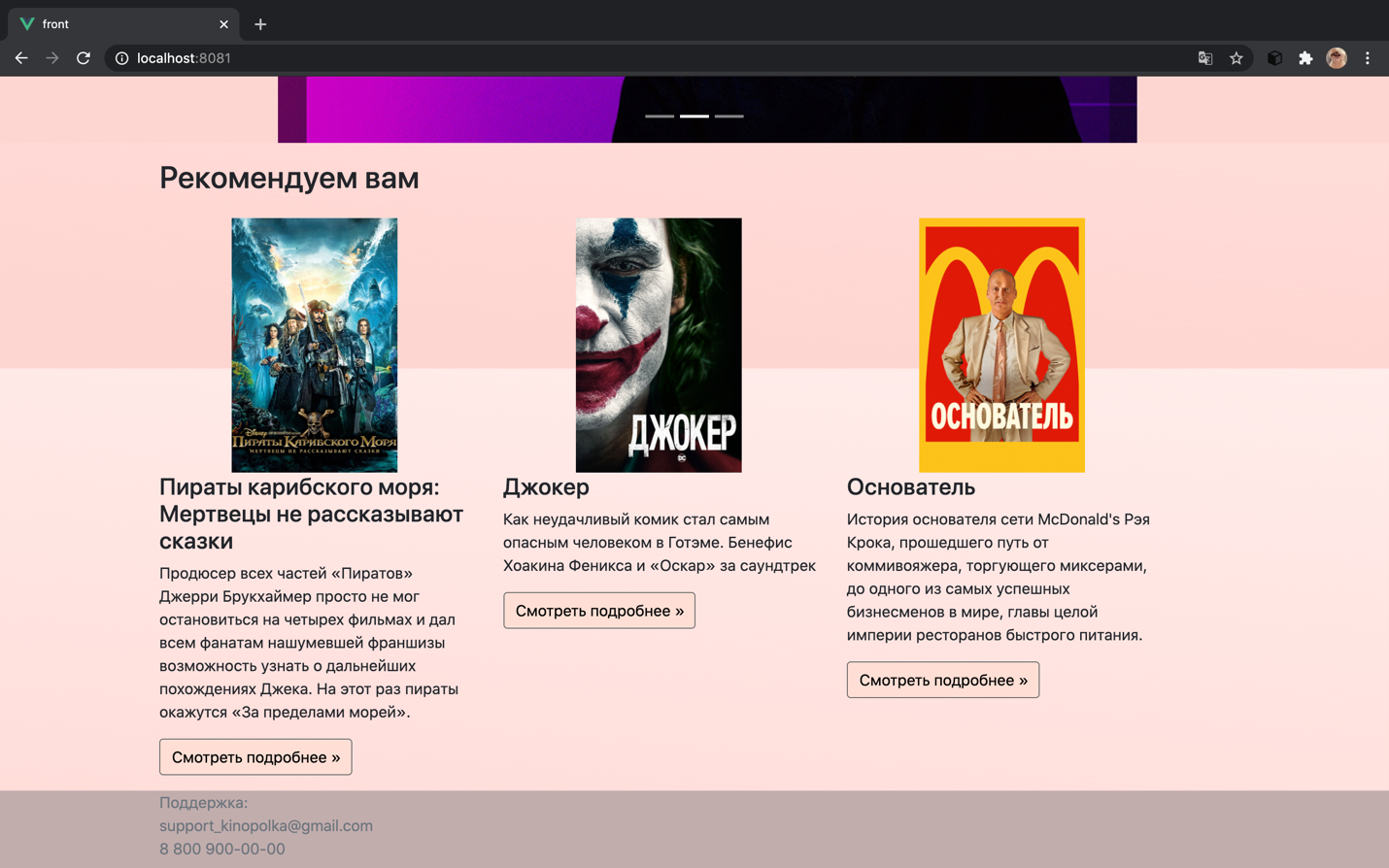
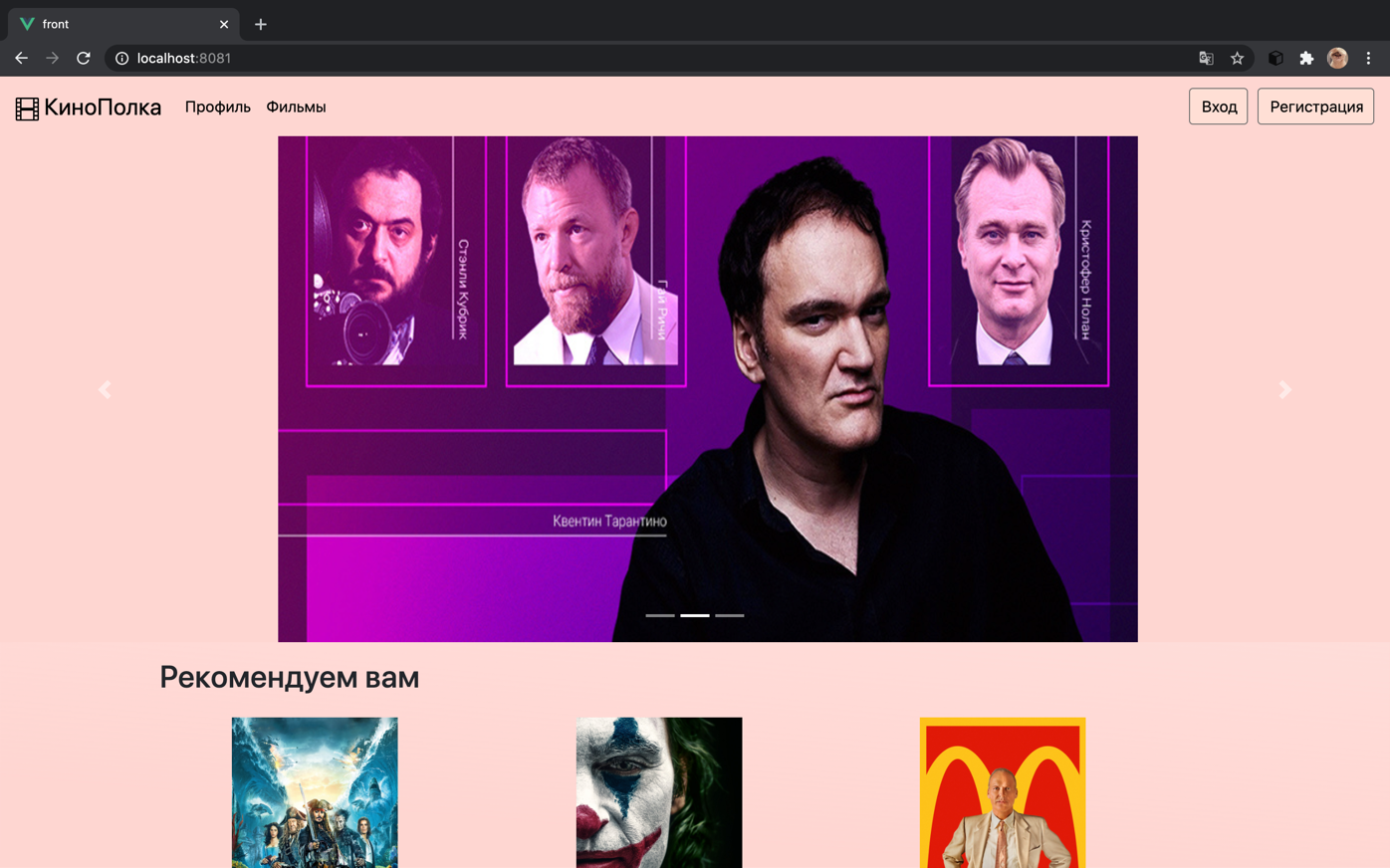
Получение списка жанров:

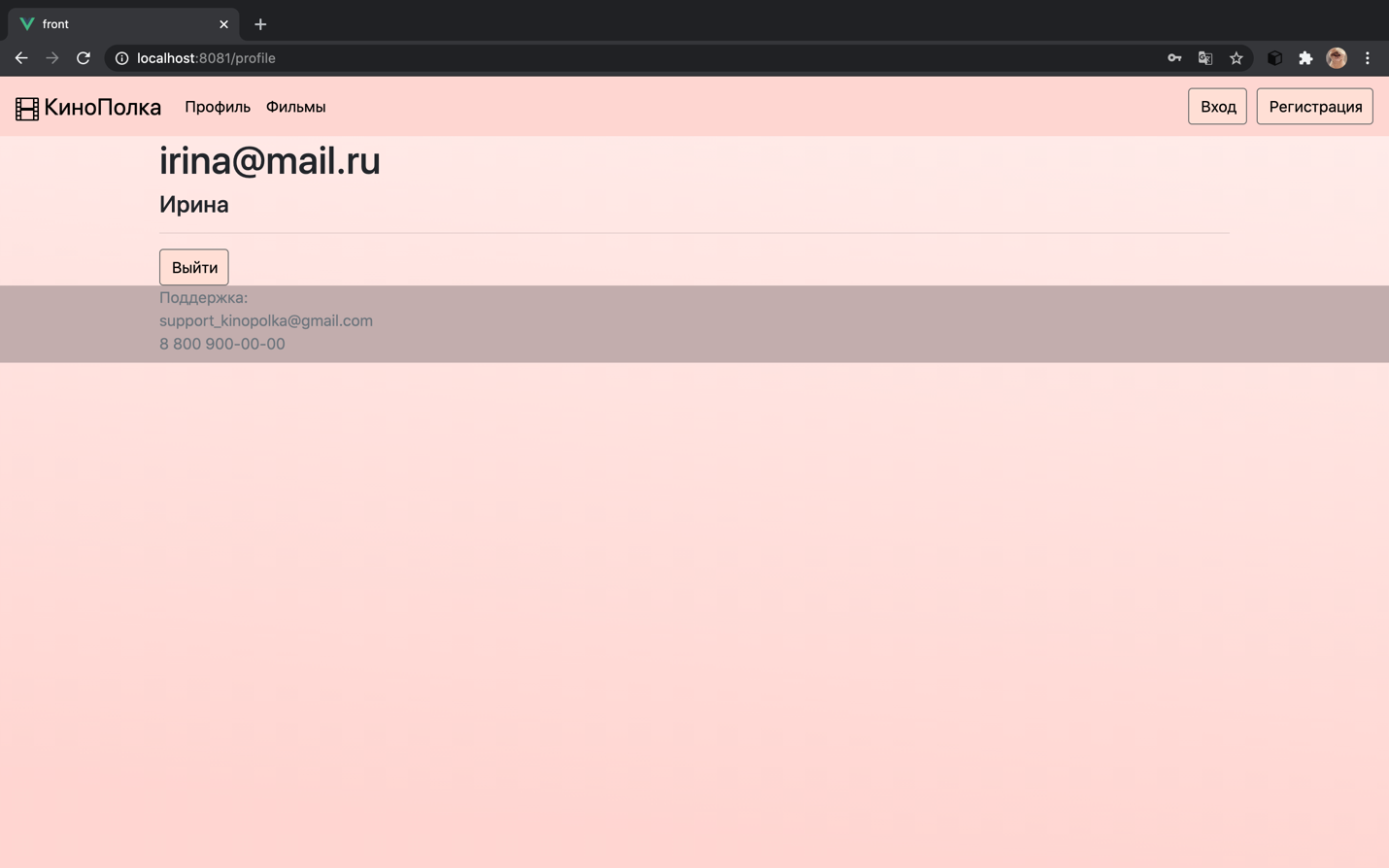
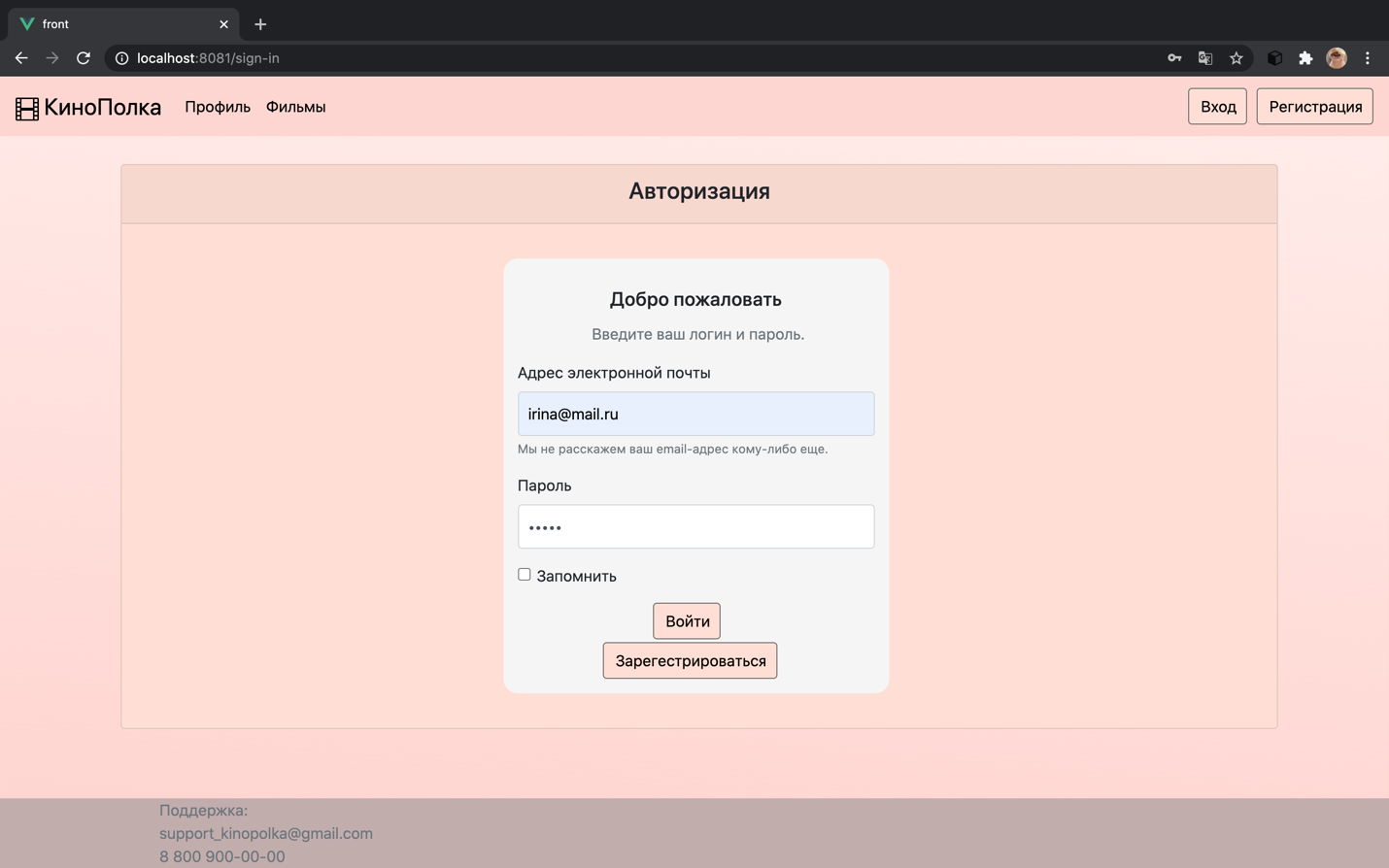
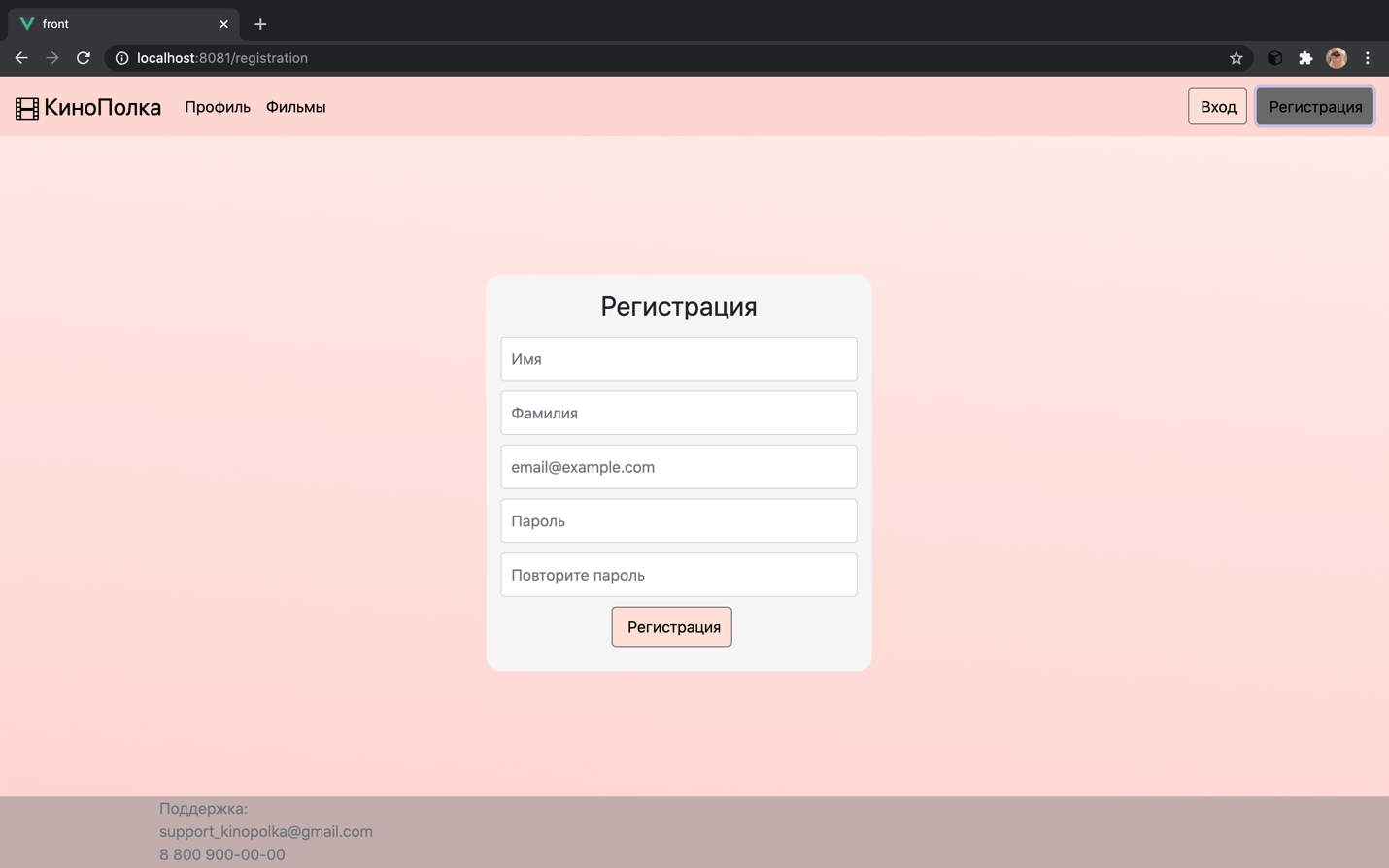
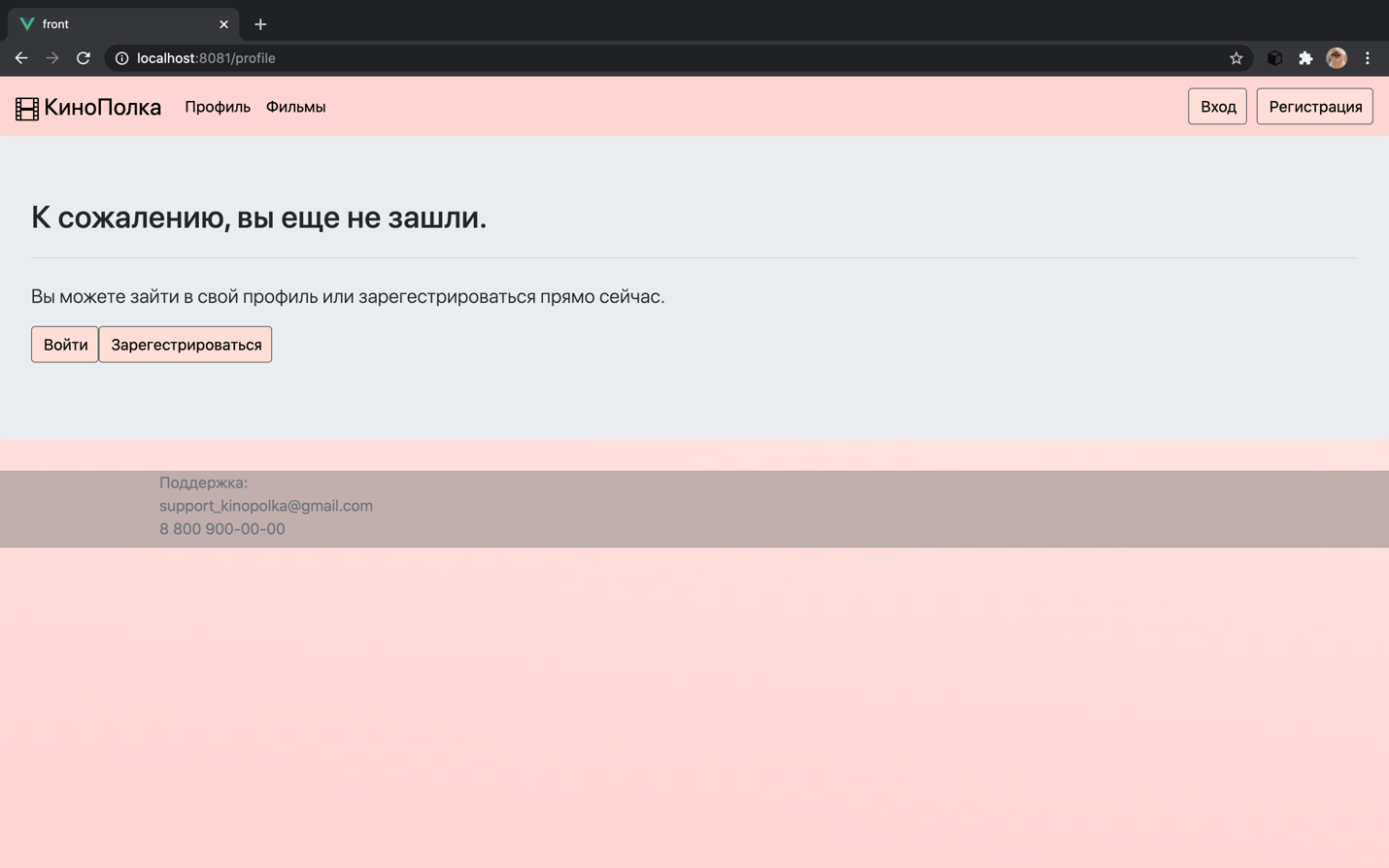
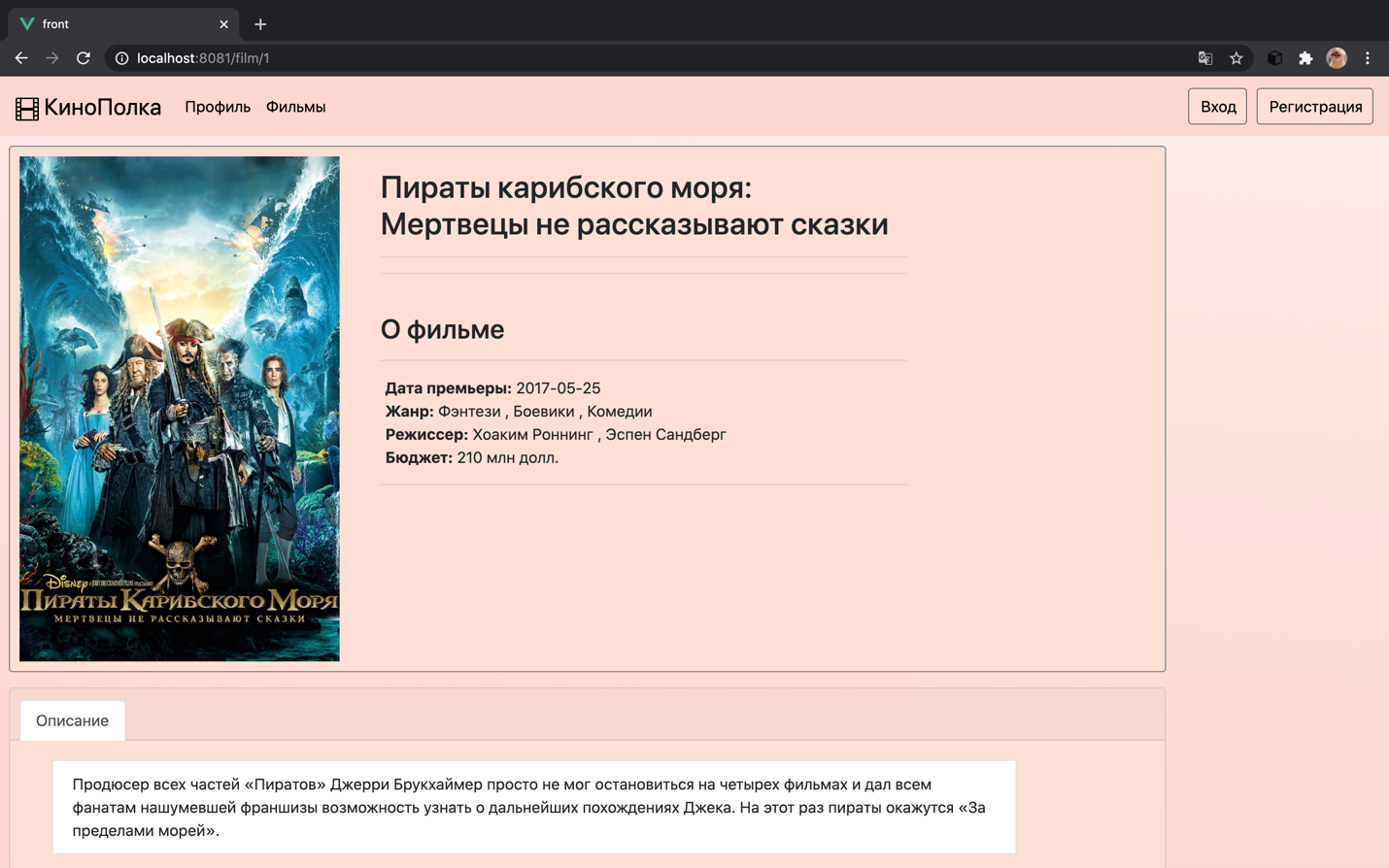
created() {  
 this.$http.get('v1/film')  
 .then((response) => this.items = response.data)  
 this.$http.get('v1/genre')  
 .then((response) => this.filter.categories = response.data)  
},

Получение фильмов по определенным жанрам:

checkCategory() {  
 let category = [];  
 this.filter.categories.forEach(function (item) {  
 let x = ***document***.getElementById(item.name);  
 if (x.checked) {  
 category.push(item.id);  
 }  
 });  
 ***console***.log(category)  
 this.$http.get(`/v1/film/filter/${category}`)  
 .then((response) => {  
 ***console***.log(response.data)  
 this.items = [...response.data]  
 })  
}

Общий вид получившегося сайта:





**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.