Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Уфимский университет науки и технологий»

Факультет информатики и робототехники

Кафедра вычислительной математики и кибернетики

Отчет Лабораторной работе №2

«Создание интерактивных элементов пользовательского интерфейса с помощью DOM, JavaScript и jQuery»

Выполнили:

студенты группы МО-321 Шарипов А.Г., Дементьев Д.В.

Проверил:

Юдинцев Б.С.

Уфа, 2023

**Задание**:

1. Реализовать анимацию главного навигационного меню по клику: сделать выделение пункта главного навигационного меню (nav), при переходе на соответствующую ему страницу, при этом наименование раздела должно меняться в заголовке страницы.
2. Реализовать выделение пункта главного навигационного меню (nav) при наведении указателя мыши. Вынести JavaScript-код для навигационного меню в отдельный js-файл.
3. Интерактивная форма: создать форму из 3-5 полей и сделать так, чтобы при нажатии на кнопку ввода появлялось модальное окно с подтверждением введенных данных.
4. Анимация текстовых блоков:

* добавить заголовки (например, тег в текстовый контент;
* реализовать анимацию таким образом, чтобы при загрузке страницы были видны только заголовки, а при клике мышкой на заголовок соответствующий ему нижестоящий блок текста появлялся;
* при повторном клике на этот же заголовок текст должен скрываться, а фон заголовка меняться.

1. Анимация изображений: реализовать изменение стиля изображения при наведении курсора (например, изменение размера, рамок или прозрачности).

**Ход работы:**

1. Реализовали анимацию главного навигационного меню по клику с помощью следующего кода:

$(document).ready(function () {

// Получаем элемент <nav> и сохраняем его в переменную navmenu

var navmenu = document.getElementsByClassName("first-level")[0];

// Получаем все элементы <li> внутри элемента <nav> и сохраняем их в переменную navlist

var navlist = navmenu.getElementsByTagName("li");

// Получаем значение id элемента <main> и сохраняем его в переменную main\_id

var main\_id = location.pathname.split('/')[7].split('.')[0];

//alert(location.pathname.split('/')[7].split('.')[0]);

// Создаем переменную menuelem, в которую будет сохранен элемент <li>, который будет активным на текущей странице

var menuelem;

switch (main\_id) {

case "index":

menuelem = navlist[0];

break;

case "about":

menuelem = navlist[1];

break;

case "unity":

menuelem = navlist[2];

break;

case "android":

menuelem = navlist[3];

break;

case "blog":

menuelem = navlist[4];

break;

case "contact":

menuelem = navlist[5];

break;

}

// Добавляем класс "active" к элементу, сохраненному в переменной menuelem, чтобы он был подсвечен на странице

menuelem.classList.add("active");

});

Рис.1(Анимация текстовых блоков и подсветка соответствующей страницы в навигационном меню)

1. Реализовали выделение пункта главного навигационного меню (nav) при наведении указателя мыши с помощью следующего кода:

nav.desktop-nav ul.first-level li a::after {

content: "";

position: absolute;

top: 50px;

left: 0px;

width: 50%;

height: 1px;

background-color: #60606e;

opacity: 0;

}

При наведении указателя мыши на ссылку начинает действовать код:  
nav.desktop-nav ul.first-level > li:hover > a::after{

top: 35px;

opacity: 1;

}

Рис.2(Выделение пункта главного навигационного меню при наведении указателя мыши)

1. Создали форму с валидацией. При нажатии на кнопку ввода появляется модальное окно с подтверждение введенных данных(Рис.6). Если ввод неверный в модальном окне выводится ошибка. (Рис.4, Рис.5). Реализация:

    function validateAllForm() {

    var name = document.getElementById("name").value;

    var email = document.getElementById("name").value;

    var object = document.getElementById("name").value;

    var message = document.getElementById("name").value;

    // Если хотя бы одно поле пустое, то выводим сообщение об ошибке

    if (!name || !email || !object || !message) alert("Заполните все поля.");

    else{

        let fname = document.getElementById("name").value;

        var alerttext = `${fname}, вы отправили запрос!`;

        alert(alerttext);}

    }

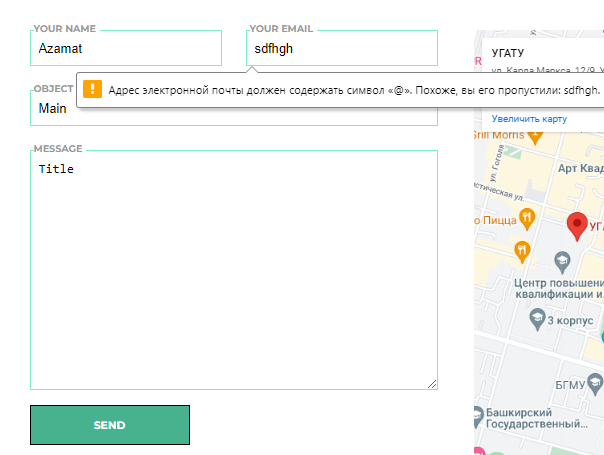


Рис.4(Форма с валидацией. Ошибка заполнения email)

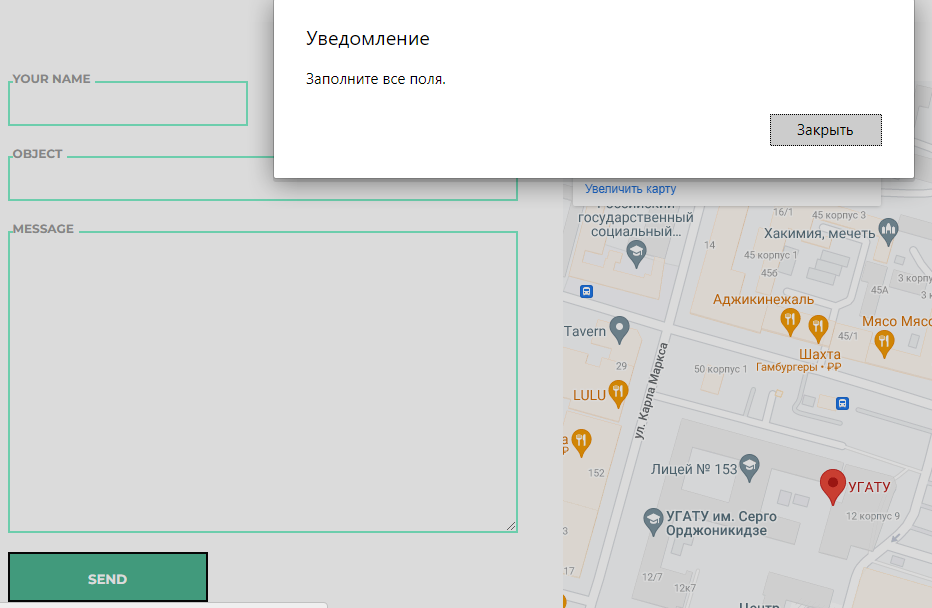


Рис.5(Форма с валидацией. Заполнены не все поля)

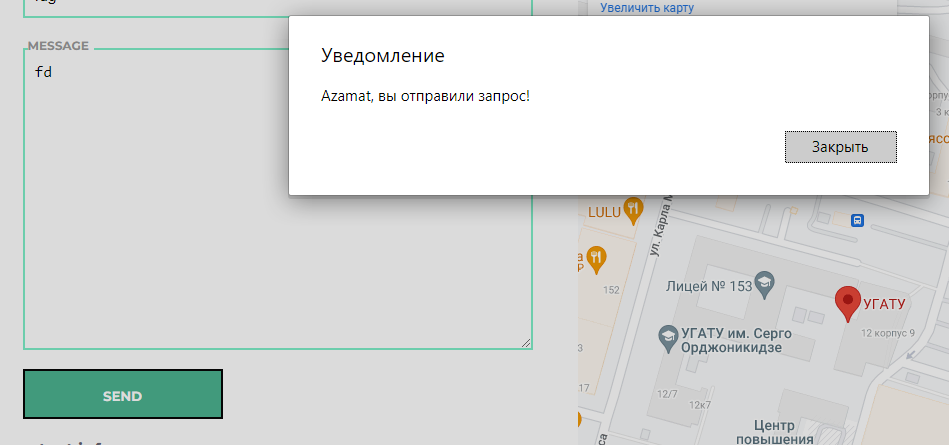


Рис.6(Форма с валидацией. Уведомление о том, что запрос отправлен)

1. Анимация текстовых блоков реализована с помощью следующего кода:

 $(document).ready(function () {

        $(".hidebox .hidable").hide();

        $(".hidebox").css("background-color", "#7beec7");

    });

    $(".hidebox").click(function () {

        $(".hidebox .hidable").hide("slow");

        $(*this*).css("background-color", "#7beec7");

    });

    $(".hidebox").dblclick(function () {

        $(".hidebox .hidable").show("slow");

        $(*this*).css("background-color", "#ffffff");

    });

Открытый блок(рис.7 )

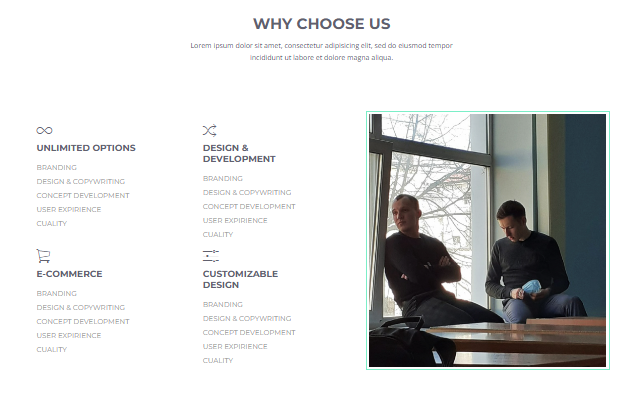


Рис.7(Открытый блок)

Закрытый блок(рис.8)



Рис.8(Закрытый блок)

1. Анимация изображений: реализовали изменение стиля изображения при наведении курсора с помощью следующего кода:

 $(".increase-on-hover").hover(

        function () {

            $(*this*).animate({

                width: "75%",

                height: "75%",

                borderRadius: "2%"

            }, "slow");

        }, function () {

            $(*this*).animate({

                width: "100%",

                height: "100%",

                borderRadius: "10%"

            }, "slow");

        }

    );

Состояние фотографии до наведения мыши(рис.9 ):



Рис.9(Состояние фотографии до наведения мыши)

Состояние фотографии после наведения мыши(рис. 10):



Рис.10(Состояние фотографии после наведения мыши)

**Вывод:**

В ходе работы были рассмотрены различные методы работы с DOM-элементами, написания и привязки обработчиков событий. Важным аспектом работы было использование jQuery для более удобной работы с DOM, особенно при выполнении анимаций и изменения стилей.