**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 9

по дисциплине «Современные технологии FRONT-END разработки»

на тему: «Взаимодействие с API браузера»

Выполнил: студент гр. ИТИ-41   
 Давидовский Д.Е.  
 Принял: преподаватель

Титова Л.К.

Гомель 2022

**Цель работы:** научиться использовать методы взаимодействия с локальным хранилищем браузера

**Задание.**

* 1. В приложении из ЛР№8 добавить меню для отображения основного контента из работы №3.
  2. Реализовать в уже имеющемся приложении функционал согласно варианту 7

Используя браузерное хранилище устройства (например, *Locale Storage*), разработать TODO-лист для различных назначений (например, для организации списка задач), реализовав интерфейс для CRUD-операций (создание новой записи, отображения записи/записей, его/их редактирования и удаления). Список TODO-листа должен быть актуальный даже после перезагрузки страницы и закрытия браузера с веб-приложением (то есть его состояние и содержимое при следующем открытии должно быть такое, каким было при закрытии страницы).

Система сохранения данных в local storage реализуется при помощи отдельной функции в Vue.js. Данные функции показана на рисунке 1.1.



Рисунок 1.1 – Функции CRUD для TODO листа

При перезагрузке страницы в фреймворке Vue.js вызывается отдельный метод, который проверяет localStorage используемого устройства и восстанавливает оттуда данные. Данная функция показана на рисунке 1.2.

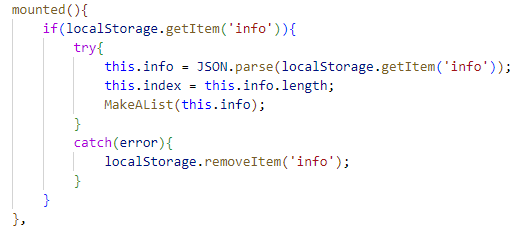


Рисунок 1.2 – Загрузка данных из LocalStorage

При успешном выполнении операции добавления новой записи, сохранения записи после редактирования или удаления записи, заставлять мобильное устройство совершить отклик в виде двух краткосрочных выбиро-сигналов (вибрацией устройства можно управлять с помощью *Vibration API*).

Система вибрации при добавлении реализована при помощи команды, показанной на рисунке 1.3.



Рисунок 1.3 – Метод вибрации устройства

На рисунке 1.4. показано веб приложение с добавленными элементами TODO листа.

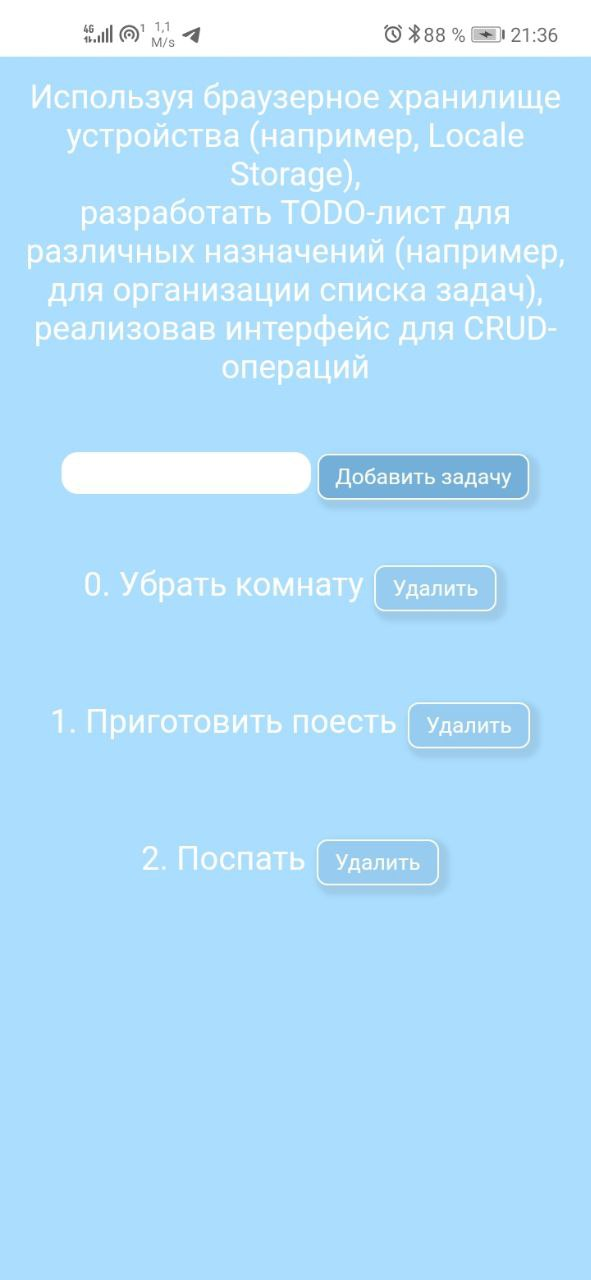


Рисунок 1.4 – Веб-приложение

Листинг страницы показан в приложении А. Также в приложении даны используемые таблицы стилей и *html* код.

**Вывод:** В ходе лабораторной работы были изучена работа с фреймворком *Vue*, а также реализовано веб приложение, позволяющее преобразовывать данные. Также разработана адаптивная верстка для телефонов. Разработан TODO лист, реализующий CRUD операции над собой и хранящий информацию в LocalStorage устройства.

**ПРИЛОЖЕНИЕ А**

(обязательное)

Листинг программы

Lab7.js

*class* *Info*{

*constructor*(*index*,*name*){

*this*.*name* = *name*;

*this*.*index* = *index*;

    }

}

*var* *inputInfo* = "*RG*";

*var* *outputInfo* = "*RR*";

*var* *app* = *new* *Vue*({

*el*: '#*app*',

*data*: {

*input* :0,

*inputValue*:'',

*outputValue*:'',

*inputS*:'',

*outputS*:'',

*info*:*[]*,

*Description*:'',

*index*:0,

    },

*mounted*(){

*if*(*localStorage*.*getItem*('*info*')){

*try*{

*this*.*info* = *JSON*.*parse*(*localStorage*.*getItem*('*info*'));

*this*.*index* = *this*.*info*.*length*;

*MakeAList*(*this*.*info*);

            }

*catch*(*error*){

*localStorage*.*removeItem*('*info*');

            }

        }

    },

*methods*: {

*ConvertInfo*: *function*(*value*,*firstIndex* = 0,*lastIndex* = 0){

*var* *result* = *Convert*(*value*,*firstIndex*,*lastIndex*);

*try*{

*this*.*inputS* = *this*.$*refs*.*inputSize*.*options[this*.$*refs*.*inputSize*.*selectedIndex]*.*text*;

*this*.*outputS* = *this*.$*refs*.*outputSize*.*options[this*.$*refs*.*outputSize*.*selectedIndex]*.*text*;

            }

*catch*{

            }

*return* *result*;

        },

*AddInfo*(){

*if*(!*this*.*Description*){

*return*;

            }

*if*(*this*.*info*.*length*==0){

*this*.*index* = 0;

            }

*var* *info* = *new* *Info*(*this*.*index*,*this*.*Description*);

*this*.*info*.*push*(*info*);

*this*.*Description*='';

*this*.*index*++;

*this*.*SaveInfo*();

*MakeAList*(*this*.*info*);

        },

*SaveInfo*(){

*const* *parsed* = *JSON*.*stringify*(*this*.*info*);

*localStorage*.*setItem*('*info*',*parsed*);

        },

*RedactInfo*: *function*(*id*){

*var* *index* = *parseInt*(*id*);

*var* *info* = *prompt*("Введите новое значение");

*if*(*info*!=*null*){

*var* *info* = *new* *Info*(*index*,*info*);

*this*.*info[index]* = *info*;

*this*.*SaveInfo*();

*MakeAList*(*this*.*info*);

            }

        },

*DeleteInfo*: *function*(*id*){

*var* *index* = *parseInt*(*id*);

*this*.*info*.*splice*(*index*,1);

*for*(*i* = 0 ;*i*<*this*.*info*.*length*;*i*++){

*this*.*info[i]*.*index* =*i*;

            }

*this*.*SaveInfo*();

*MakeAList*(*this*.*info*);

        }

    }

})

*function* *RedactInfo*(*id*){*app*.*RedactInfo*(*id*)}

*function* *DeleteInfo*(*id*){*app*.*DeleteInfo*(*id*)}

*function* *MakeAList*(*massInfo*){

*var* *infoObject* = *document*.*getElementById*('*InfoAbout*');

*infoObject*.*innerHTML* = "";

*massInfo*.*forEach*(*element* => {

*var* *InfoElement* = *document*.*createElement*("*div*");

*var* *InfoAbout* = *document*.*createElement*("*p*");

*var* *deleteButton* = *document*.*createElement*("*button*");

*var* *currentID* = *element*.*index*;

*InfoAbout*.*innerText* = *element*.*index* + ". " + *element*.*name*;

*InfoAbout*.*setAttribute*("*onClick*","*RedactInfo*(*\*""+*currentID*+"*\*")");

*InfoAbout*.*setAttribute*("*class*","*ToDoElement*");

*deleteButton*.*setAttribute*("*onClick*","*DeleteInfo*(*\*""+*currentID*+"*\*")");

*deleteButton*.*setAttribute*("*class*","*DeleteButton*");

*deleteButton*.*innerText* = "Удалить";

*InfoElement*.*setAttribute*("*class*","*infoElement*");

*InfoElement*.*appendChild*(*InfoAbout*);

*InfoElement*.*appendChild*(*deleteButton*);

*infoObject*.*appendChild*(*InfoElement*);

    });

*window*.*navigator*.*vibrate*(*[*150,150*]*);

}

*function* *Convert*(*value*,*firstIndex*, *lastIndex*){

*var* *amount* = *parseInt*(*firstIndex*)-*parseInt*(*lastIndex*);

*value* = *parseFloat*(*value*);

    //*console*.*log*("*Value*: "+*value*);

    //*console*.*log*("*FirstIndex*: "+*firstIndex*);

    //*console*.*log*("*LastIndex*: "+*lastIndex*);

*amount* = *parseInt*(*amount*);

    //*console*.*log*("*Amount*: "+*amount*);

*if*(*amount*>0){

*for*(*var* *i* = 0; *i*<*amount*;*i*++){

*value* = *value*\*1024;

            }

            //*console*.*log*("*Returning* *value*:"+*value*);

*return* *value*;

        }

*else* *if*(*amount*<0){

*amount*\*=-1;

*for*(*var* *i* = 0; *i*<*amount*;*i*++){

*value* = *value*/1024;

            }

            //*console*.*log*("*Returning* *value*:"+*value*);

*return* *value*;

        }

*else*{

            //*console*.*log*("*Returning* *value*:"+*value*);

*return* *value*;

        }

*console*.*log*("*No* *switch*");

}

Lab7.css

\* {

    margin:0;

    padding:0;

    font-family: 'Open Sans', sans-serif;

}

\*:focus{

    outline: none;

}

.TaskInfo{

    color:white;

    margin: 25pt 0 0 0;

    font-size: 15pt;

}

#RotationInfo{

    display:none;

}

.header-horizontal{

    position: relative;

    height: 55px;

    background-color: #97ccf0;

    box-shadow: 0 0 10px rgba(39, 39, 39, 0.5);

}

.Text a{

    display: inline-block;

    width: 100%;

    text-align: center;

    font-family: fantasy;

    font-weight: 700;

    color: white;

    font-size: 1.5em;

    padding-top: 8px;

    font-family: 'Nanum Gothic', sans-serif;

    text-decoration: none;

}

.Text a:visited{

    text-decoration: none;

    color: white;

}

.header-horizontal ul li{

    color: white;

}

main{

    background-color: #aaddff;

    width: 100%;

    height: 5000px;

    display:flex;

}

.button{

    margin-top: 15pt;

    border: 1px solid;

    padding: 10px 40px;

    border-radius: 0.5em;

    background-color: #97ccf0;

    box-shadow: 3px 3px 3px 3px rgba(39, 39, 39, 0.1);

    color: white;

}

.button:hover{

    background-color: #74afda;

}

.inputText{

  margin-top: 15pt;

  border: 1px solid;

  border-radius: 0.5em;

  font-size: 20px;

  width: 250px;

}

.inputAnagram{

  margin-top: 15pt;

  border: 1px solid;

  border-radius: 0.5em;

  font-size: 20px;

}

.app{

    width: 100%;

    text-align: center;

    color: white;

}

.app input{

    border-color: white;

    overflow: none;

    color: #74afda

}

.app div{

    margin: 15pt 0;

}

.app p{

    margin: 15pt 0;

    font-size: 15pt;

}

.app select{

    padding: 5pt 10pt;

    font-size: 15pt;

    border-radius: 15pt;

    border-color: white;

    background-color: #97ccf0;

    color: white;

}

.app select:hover{

    background-color: #74afda;

}

.ToDoElement{

    display: inline-block;

}

.app button{

    margin-top: 15pt;

    border: 1px solid;

    padding: 5px 10px;

    border-radius: 0.5em;

    background-color: #97ccf0;

    box-shadow: 3px 3px 3px 3px rgba(39, 39, 39, 0.1);

    color: white;

}

.app button:hover{

    background-color: #74afda;

}

.app p:hover{

    color:rgb(186, 186, 194);

}

.DeleteButton

{

    margin: 0 5pt;

}

/\* Portrait \*/

@media all

  and (max-width: 720px)

  and (max-height: 1560px)

  and (-webkit-device-pixel-ratio: 2)

  and (orientation: portrait)

{

    .inputText{

        margin-top: 15pt;

        border: 1px solid;

        border-radius: 0.5em;

        font-size: 20px;

        width: 150px;

      }

}

/\* Landscape \*/

@media all

  and (max-width: 1560px)

  and (max-height: 720px)

  and (-webkit-device-pixel-ratio: 2)

  and (orientation: landscape)

{

.inputText{

    margin-top: 15pt;

    border: 1px solid;

    border-radius: 0.5em;

    font-size: 20px;

    width: 150px;

  }

  #app{

    display: none;

  }

  #RotationInfo{

    display: block;

    width: 100%;

    text-align: center;

    color: white;

    padding: 15pt 0;

  }

}

@media (prefers-color-scheme: dark){

    .header-horizontal{

        background-color: #a6a6a7;

        box-shadow: 0 0 10px rgba(39, 39, 39, 0.5);

    }

    #RotationInfo{

        color: rgb(186, 186, 194);

      }

    .app select{

        border-color: rgb(186, 186, 194);;

        background-color: #5f5f66;

        color: rgb(186, 186, 194);;

    }

    .app button{

        background-color: #5f5f66;

        color: rgb(186, 186, 194);

    }

    .app select:hover{

        background-color: #393b3d;

    }

    .app button:hover{

        background-color: #393b3d;

    }

    .app{

        color: rgb(186, 186, 194);;

    }

    .app input{

        border-color: rgb(186, 186, 194);;

        color: #3a586e

    }

    .header-horizontal{

        background-color: #313841;

        box-shadow: 0 0 10px rgba(39, 39, 39, 0.5);

    }

    .Text a{

        color: rgb(186, 186, 194);;

    }

    .Text a:visited{

        color: rgb(186, 186, 194);;

    }

    .header-horizontal ul li{

        color:rgb(186, 186, 194);;

    }

    main{

        background-color: #23262b;

    }

    .inputText{

        background-color: #c3c3c4;

    }

}

Lab7.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Давидовский Даниил</title>

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link href="styleLab7.css" rel="stylesheet" type="text/css">

    <title>Document</title>

</head>

<body>

    <header>

        <nav class="header-horizontal">

            <ul>

                <li class="Text"><a href="D:\\Учеба\\FrontEnd\\Project\\index.html">Лабораторные работы</a></li>

            </ul>

        </nav>

    </header>

    <main>

        <div id="app" class="app">

            <p class="TaskInfo">7   Перевод объёма информации. Предоставлять пользователю указать две величины объёма информации:<br>

                 исходную и ту, в которую надо конвертировать<br> (например, байты в килобайты, гигабайты в мегабайты, терабайты в гигабайты).</p>

            <div>

                <input class="inputText" type="text" id="inputInfo" v-model="input">

                <p>{{input}} {{inputS}} равняется {{ConvertInfo(input,inputValue,outputValue)}} {{outputS}}</p>

            </div>

            <div>

                <select name="" id="inputSelect" v-model="inputValue" ref="inputSize">

                    <option value="1" selected>Байт</option>

                    <option value="2">Килобайт</option>

                    <option value="3">Мегабайт</option>

                    <option value="4">Гигабайт</option>

                    <option value="5">Терабайт</option>

                    <option value="6">Петабайт</option>

                    <option value="7">Эксабайт</option>

                </select>

                <select name="" id="outputSelect" v-model="outputValue" ref="outputSize">

                    <option value="1" selected>Байт</option>

                    <option value="2">Килобайт</option>

                    <option value="3">Мегабайт</option>

                    <option value="4">Гигабайт</option>

                    <option value="5">Терабайт</option>

                    <option value="6">Петабайт</option>

                    <option value="7">Эксабайт</option>

                </select>

            </div>

            <p class="TaskInfo">Используя браузерное хранилище устройства (например, Locale Storage),<br>

                 разработать TODO-лист для различных назначений (например, для организации списка задач),<br>

                  реализовав интерфейс для CRUD-операций</p>

            <div class="ToDoList">

                <div class="InputForm">

                    <input v-model="Description" class="inputText">

                    <button @click="AddInfo">Добавить задачу</button>

                </div>

                <div id="InfoAbout">

                </div>

            </div>

        </div>

        <div id="RotationInfo">

            <h1>Невозможно отобразить в ландшафтном режиме</h1>

        </div>

    </main>

</body>

<script src="https://cdn.jsdelivr.net/npm/vue@2/dist/vue.js"></script>

<script src="Lab7.js"></script>

</html>