**МУЛЬТИМЕДИЙНЫЙ ИНТЕРФЕЙС**

**ДЛЯ**

**ДИАЛОГА С БАЗОЙ ЗНАНИЙ**

РАЗРАБОТКА ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

*Диалоговый пользовательский интерфейс* с Базой знаний на Web-страницах информационной системы представляет собой использование интерактивных всплывающих диалоговых окон по запросам пользователя на Web-страницах. Для создания интерактивных сайтов чаще всего используется библиотека JavaScript **jQuery**.

Библиотека **jQuery** реализует архитектуру на основе концепций, положенных в основу языка разметки **HTML** и языка каскадных таблиц стилей **CSS,** со следующими базовыми возможностями:

* может обращаться к любому элементу объектной модели Web-документа **DOM** и предлагает механизм селекторов;
* может работать с событиями;
* осуществляет различные визуальные эффекты с изменением внешнего вида страницы с использованием **CSS**;
* имеет различные JavaScript**-плагины**, предназначенные для создания элементов пользовательских интерфейсов, например, диалоговых окон.

**СТРУКТУРА ДИАЛОГОВОЙ WEB-СТРАНИЦЫ**

Для реализации удобного диалога с Базой знаний, содержащей мультимедийную информацию, на диалоговой Web-странице необходимо разработать интерфейс с диалоговыми окнами:

1. В **папку сайта** добавить файлы:

* библиотеку jQuery – файл **jquery.js,**
* файл разработанного в ЛР по Теме 12 программного кодана языке JavaScript для обработки диалога, например, с именем **dialog.js** с функцией формирования диалогового окна, например, с именем **dialog\_window();**
* файлвнешней таблицы разработанных для диалогаCSS-стилей, сохраненного, например, с именем **style.css**.

1. В **HTML**-разметку диалоговой Web-страницы добавить следующие элементы:

* строку с тегом **<script>** для подключения файла библиотеки **jQuery jquery.js;**
* строку с тегом **<script>** для подключения **аудио сервиса Yandex**;
* строку с тегом **<script>** для подключения внешнего файла с программным кодом JavaScript **dialog.js,** в который добавить код, обеспечивающий интерфейс на основе диалогового окна;
* строку с тегом **<link>** для подключения внешней таблицы CSS-стилей из файла **style.css;**
* в теге **<body>** загрузить оператором **onLoad()** созданную дополнительно **функцию организации интерфейса для диалогового окна** для Web-страницы, например, с именем **dialog\_window().**

Таким образом, общая структура Web-страницы с мультимедийным диалогом может выглядеть следующим образом:

**<html lang="ru" xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<title>Диалог с базой знаний</title>**

**<meta charset="utf-8" />**

**<script src="jquery.js"></script>**

**<script src="dialog.js"></script>**

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

**</head>**

**<body onLoad="dialog\_window()">**

**…**

**</body>**

**</html>**

**СОЗДАНИЕ ИНТЕРФЕЙСА ДИАЛОГОВОГО ОКНА**

Для создания соответствующего интерфейса мультимедийного диалога в программный код файла **dialog.js** семантического генератора ответов из Базы знаний необходимо добавить код для создания диалогового окна и программных модулей для функционирования его интерфейса – флаговую **переменную** для активизации диалога и три функции: 1) функцию **формирования** диалогового окна; 2) функцию **выдвижения-скрытия** диалогового окна; 3) функцию **вывода ответа** на заданный вопрос в диалоговом окне.

Вводим в код файла **dialog.js** переменную, используемую для активизации диалога:

**var dialogOn = false;**

Далее создаем для диалогового окна три следующие функции.

**ФУНКЦИЯ ФОРМИРОВАНИЯ ДИАЛОГОВОГО ОКНА:**

**function dialog\_window(){**

Добавляем на страницу диалоговое окно как новый раздел *div* с идентификатором *dialog*, задаем ему CSS-стиль ***dialog***, располагающий этот раздел справа от основного раздела страницы, и затем надвигаем его левое поле на страницу на 25px:

**document.body.innerHTML+=**

**"<div id='dialog' class='dialog'>"**

Здесь используется внешний CSS-стиль ***dialog***:

***.dialog{***

***position:fixed;***

***top:10%;***

***left:100%;***

***width:800px;***

***height:800px;***

***margin-left:-30px;***

***margin-top:-25px;***

***padding:10px;***

***border:#000 2px solid;***

***border-radius:10px;***

***background:#fff;***

***color:#f00;***

***z-index:1500;***

***font-size:12pt;***

***}***

Не закрывая этот раздел, добавляем в него новый раздел со строкой *«Нажми, чтобы спросить!»*, задаем ему CSS-стиль ***label***, который уже будет вложен в стиль ***dialog*** родительского окна и команду на выполнение функции ***openDialog()***выдвижения диалогового окна щелчком мыши:

**+"<div class='label' onclick='openDialog()'>Нажми, чтобы спросить!</div>"**

Здесь используется внешний CSS-стиль ***.dialog. label***:

***.dialog .label{***

***transform: rotate(270deg);***

***width:430px;***

***height:20px;***

***text-align:center;***

***color:#f00;***

***overflow:hidden;***

***display:inline-block;***

***margin-left:-26%;***

***margin-top:50%;***

***cursor:pointer;***

***}***

Добавляем также в раздел диалогового окна еще один новый раздел для заголовка *со словом «История»* и задаем ему CSS-стиль ***header****:*

**+"<div class='header'>История:</div>"**

Здесь используется внешний CSS-стиль, вложенный в стиль всего диалогового окна ***.dialog header***:

***.dialog .header{***

***text-align:center;***

***margin-top:-430px;***

***font-size:20px;***

***}***

Далее добавляем в раздел диалогового окна еще один новый раздел,задаем емуему CSS-стиль ***history***и задаем ему идентификатор *history***:**

**+"<div class='history' id='history'></div>"**

Здесь используется внешний CSS-стиль ***.dialog .history***:

***.dialog .history{***

***height: 730px;***

***overflow-x:hidden;***

***overflow-y:scroll;***

***}***

Наконец, добавляем в диалоговое окно последний новый раздел с CSS-стилем ***question***и вложенным в негополем формы ***input*** с идентификатором ***Qdialog*** и с текстом-приглашением ***'Введите вопрос'***,исчезающим при вводе вопроса, а также в этот раздел в строке ниже включим кнопку с названием *«****Спросить****»* с командой для запуска функции***ask(),*** которая обрабатывает заданный вопрос и выводит в раздел ***history*** полученный в результате обработки ответ:

**+"<div class='question'><input id='Qdialog' placeholder='Введите вопрос'/> <br><button onclick='ask(\"Qdialog\")'>Спросить</button></div>"**

Здесь используется следующие CSS-стили:

***.dialog .history .question{***

***margin-left:40px;***

***background:#99f;***

***margin:5px;***

***width:100%;***

***border-radius: 5px;***

***padding:5px;***

***}***

***.dialog button{***

***margin:2px 20px;***

***width:95%;***

***border:1px #666 solid;***

***border-radius: 5px;***

***font-size:20px;***

***}***

***input{***

***font-size:20px;***

***width:100%;***

***border-radius:10px;***

***padding:2px 10px;***

***}***

После добавления этого раздела нужно закрыть общий раздел диалогового окна, закончив тем самым процесс его формирования на Web-странице:

**+"</div>";**

Наконец, осталось закрыть тело функции **dialog\_window():**

**}**

**ФУНКЦИЯ ПЛАВНОГО ПОКАЗА-СКРЫТИЯ ДИАЛОГОВОГО ОКНА:**

**function openDialog(){**

**if(dialogOn){** //анимация закрытия диалогового окна

**$("#dialog").animate({"margin-left":"-30px"},1000,function(){});**

**dialogOn=false;**

**}**

**else{** //анимация открытия диалогового окна

**$("#dialog").animate({"margin-left":"-1250px"},1000,function() {});**

**dialogOn=true;**

**clearInterval(timer);**

**}**

**}**

**ФУНКЦИЯ ОБРАБОТКИ ВОПРОСА В ДИАЛОГОВОМ ОКНЕ:**

**function ask(questionInput){**

*//переменная для считывания содержания окна ввода вопроса****:***

**var question=document.getElementById(questionInput).value;**

*//активизация диалога*

**dialogOn=true;**

*//создаем переменную и сохраняем в ней тег <div>*

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

//задаем класс оформления созданного блока

**newDiv.className='question';**

//заполняем созданный блок *текстом вопроса*

**newDiv.innerHTML=question;**

//вставляем созданный блок в блок <history> и закрываем его

**document.getElementById("history").appendChild(newDiv);**

**+"</div>";**

//создаем еще один блок <div>

**newDiv=document.createElement("div");**

//задаем класс оформления созданного блока

**newDiv.className='answer';**

//наполняем созданный блок ответом, полученным от функции getAnswer()

**newDiv.innerHTML=getAnswer(question);**

//ОЗВУЧИВАНИЕ ПОЛУЧЕННОГО ОТВЕТА

//добавляем в ответ тег аудио, ссылающийся на звук от синтезатора Yandex

**newDiv.innerHTML+="<audio controls='true' autoplay='true'**

//обращение к голосовому сервису Yandex

**src='http://tts.voicetech.yandex.net/**

//в котором указывается формат звука и язык озвучиваемого текста **generate?format=wav&lang=ru-RU&**

//а также ключ доступа к SpeechKit Cloud API

**key=4a4d3a13-d206-45fc-b8fb-e5a562c9f587&**

//озвучиваемый текст, который берется из сгенерированного ответа

**text="+newDiv.innerText+"'></audio>";**

//вставка зввукового блока в блок <history> и закрываем его

**document.getElementById("history").appendChild(newDiv);**

**+"</div>";**

//запуск звука

**if(newDiv.lastChild.tagName=="AUDIO"){newDiv.lastChild.play();}**

//КОНЕЦ ОЗВУЧИВАНИЯ ОТВЕТА

//прокрутка в окне истории ответа в самый низ

**document.getElementById("history").scrollTop =**

**document.getElementById("history").scrollHeight;**

//очистка текстового поля для ввода нового вопроса

**document.getElementById(questionInput).value="";**

**}**

В этой функции используются следующие внешние CSS-стили:

***.dialog input, .dialog button, .dialog .history{***

***margin:2px 25px;***

***width:85%;***

***border:1px #666 solid;***

***border-radius: 5px;***

***font-size:20px;***

***}***

***.dialog .history .question {***

***margin:5px;***

***width:100%;***

***border-radius: 5px;***

***padding:5px;***

***margin-left:40px;***

***background:#99f;***

***}***

Для оформления в ответе изображений, аудио и видео используются следующие стили:

***.dialog .history .answer{***

***margin:5px;***

***width:100%;***

***border-radius: 5px;***

***padding:5px;***

***margin-right:40px;***

***background:#9ff;***

***}***

***.dialog .history .answer img, .dialog .history .answer audio, .dialog .history .answer video {***

***cursor: pointer;***

***display: block;***

***max-width: 650px;***

***}***