Министерство сельского хозяйства Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Пермский государственный аграрно-технологический университет имени академика Д.Н. Прянишникова»

Кафедра Информационных технологий и программной инженерии

КУРСОВОЙ ПРОЕКТ

по дисциплине: Программная инженерия

на тему: Разработка продукционной экспертной системы.

Выполнил: студент 3-го курса очного

отделения специальности 09.03.03 Программная информатика

шифр ПИб-436-2019

Юдин Сергей Александрович

Проверил: доцент кафедры ИТиПИ,

к.т.н., доцент Беляков Андрей Юрьевич

Оглавление

[Постановка задачи 3](#_Toc93853011)

[Разработка структуры для хранения данных 4](#_Toc93853012)

[Программная реализация 5](#_Toc93853013)

[Заключение 7](#_Toc93853014)

[Список литературы 8](#_Toc93853015)

[Приложение 9](#_Toc93853016)

постановка задачи

Прежде всего, поставим задачу, для решения которой будет разрабатываться экспертная система. Подходящей задачей, вытекающая из следующей ситуации: человек в первый раз видит игру и ему хочется узнать какие танки присутствуют в данной игре. Для решения задачи необходимо разработать экспертную систему, показывающая какие танки присутствуют в данной игре

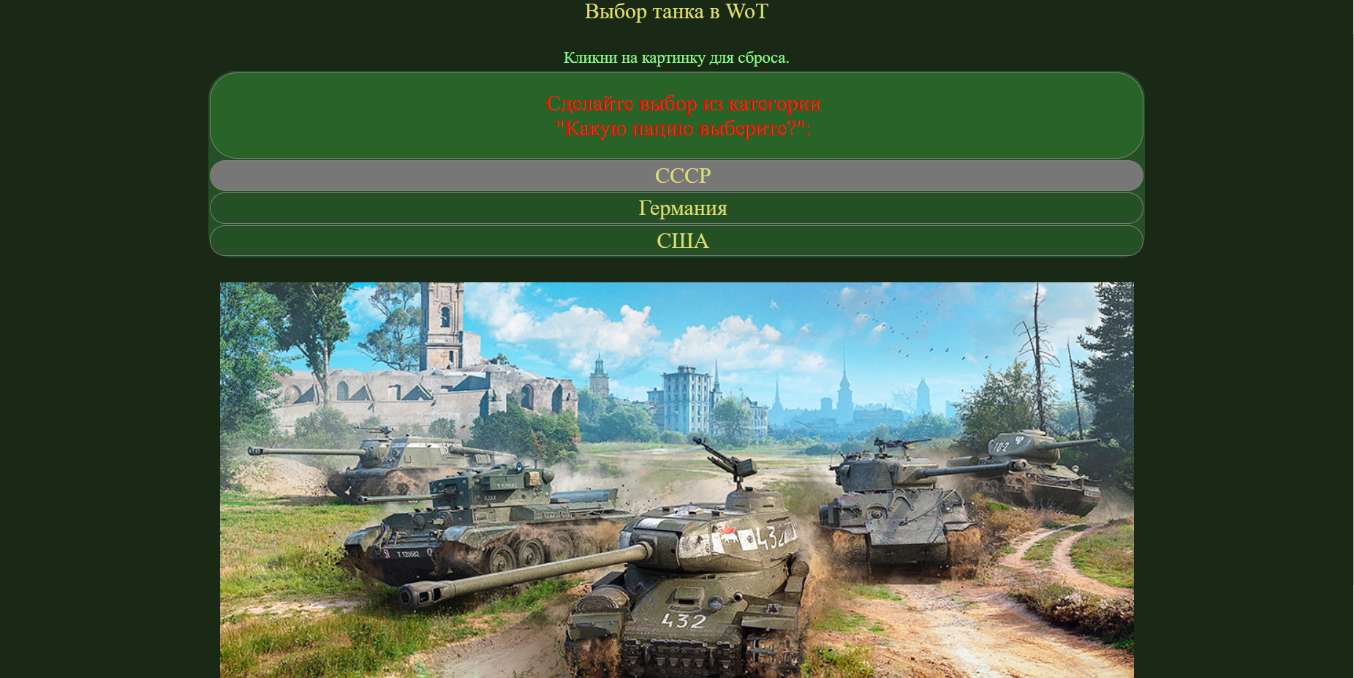
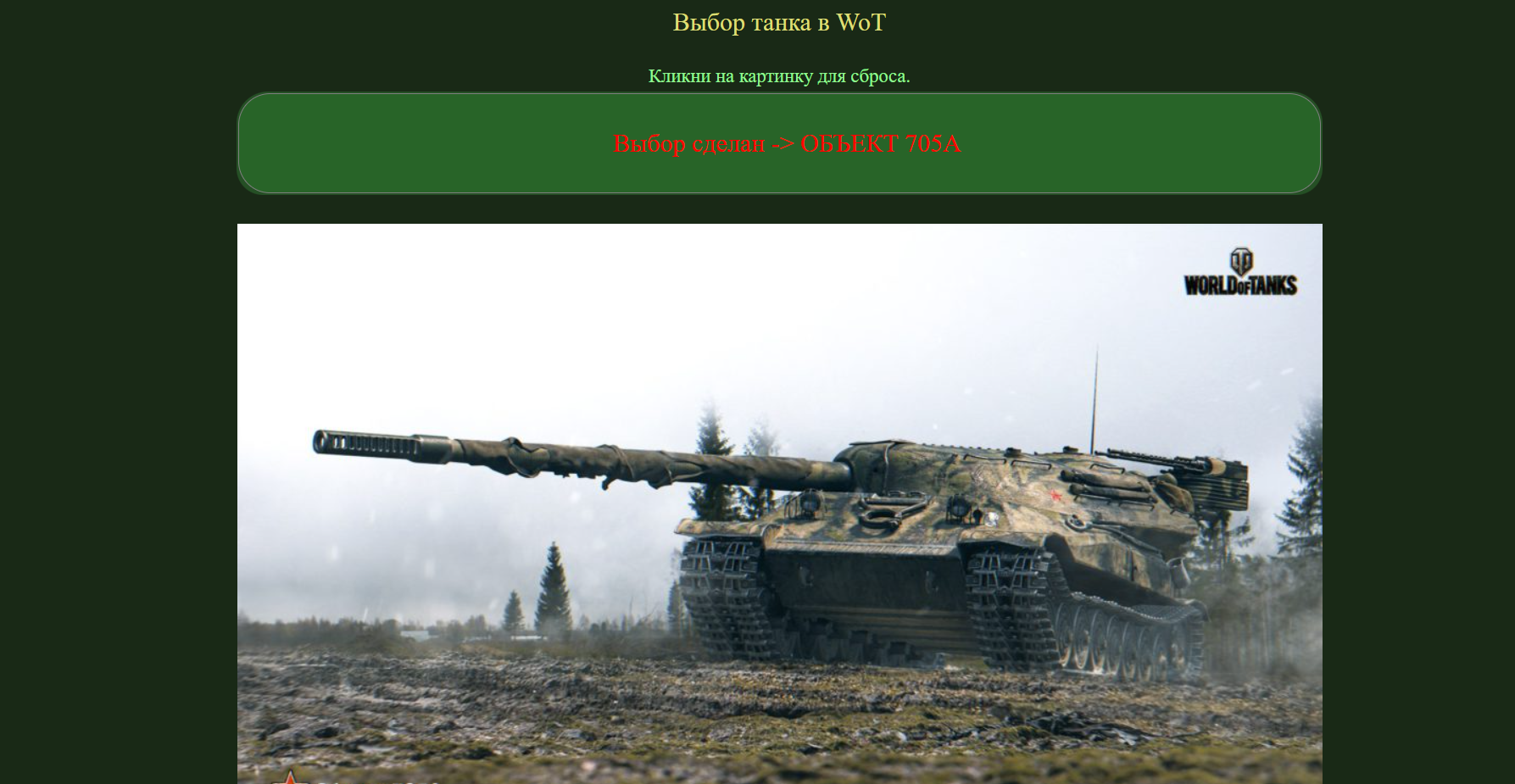
Разработка производилась в системе программирования Visual Studio Code, на таком языке программирования как JavaScript, так как данный язык хорошо подходит для разработки экспертной системы с использование, так же для создания интерфейса пользователя я использовал язык html и css.

Рисунок 1 – Стартовая форма

Рисунок 2 – Результат

Разработка структуры для хранения данных

В базе знаний структура хранений данных разработана в формате JSON. JSON – это текстовый формат обмена данными, основанный на JavaScript. Несмотря на происхождение от JavaScript, формат считается независимым от языка и может использоваться практически с любым [языком программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F). Для многих языков существует готовый код для создания и обработки данных в формате JSON.

Объект JSON это формат данных — ключ-значение, который обычно рендерится в фигурных скобках. Пример показан на листинге 1.

Листинг 1 – Пример кода JSON

"Какую нацию выберите?": ["СССР", "Германия", "США",],

"СССР": ["Лёгкий Танк","Средний Танк","Тяжёлый Танк", "Противотанковая Сау"],

"Германия": ["Лёгкие танки","Средние танки","Тяжёлые танки", "Противотанковые сау"],

"США": ["лёгкий танк","средний танк","тяжёлый танк", "противотанковая сау"],

"Лёгкий Танк": ["Т-100 ЛТ"],

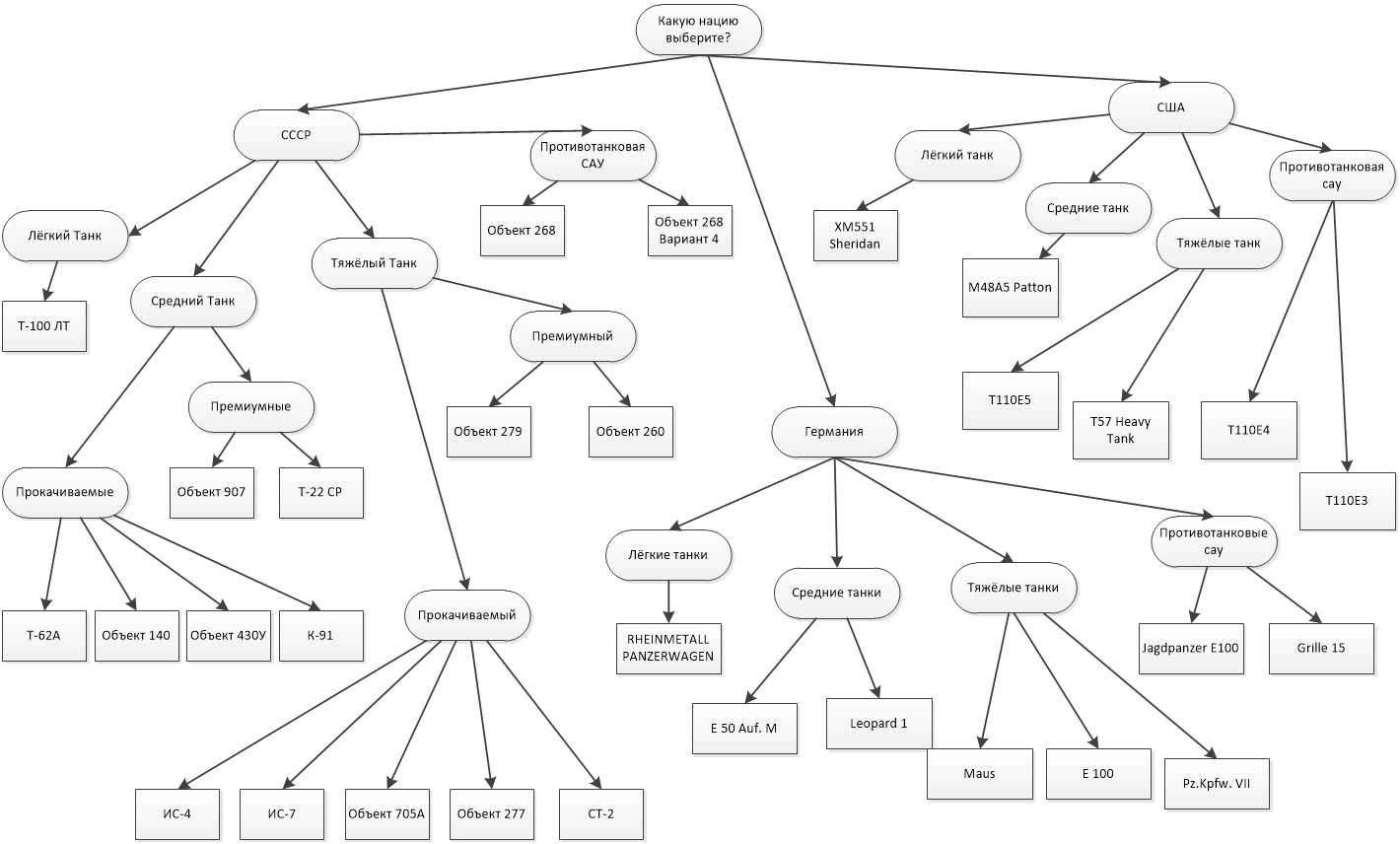
"Средний Танк": ["Прокачиваемые", "Премиумные"],

"Тяжёлый Танк": ["Прокачиваемый", "Премиумный"],

"Противотанковая Сау": ["Объект 268", "Объект 268 Вариант 4"],

Пары ключ-значение разделены двоеточием, например "key" : "value". Каждая пара значений разделена двоеточием, таким образом середина JSON выглядит так: "key" : "value", "key" : "value", "key" : "value".

На рисунке 3 представлено дерево решений экспертной системы.

Рисунок 3 – дерево решени

программная реализация

При создании продукционной экспертной системы была создана машина вывода, которая занимается обработкой событий выбора из значений ответа на текущий вопрос.

Были объявлены глобальные переменные, которые необходимы для взаимодействия с базой знаний и интерфейсом пользователя, они представлены на листинге 1.

Листинг 1 – Глобальные переменные, объявленные в модуле es\_bin.js

let colors = ['', '#777'];

let run;

let \_quest = document.querySelector('#quest');

let \_title = document.querySelector('#title');

let \_help = document.querySelector('#help');

На листинге 2 можно увидеть код функции запуска приложения, а также выведение на экран первого вопроса и заголовка страницы.

Листинг 2 – код запуска приложения модуля es\_bin.js

const init = function () {

run = true;

\_title.innerHTML = es.title;

\_help.innerHTML = 'Кликни на картинку для сброса.';

print\_dialog(es.start);

};

Далее на листинге 3 представлен код реализации обработчика событий. Обрабатываются все варианты ответов и задаётся переменная с текущими ответами.

В случае, если вариантов ответов больше нет, выдаётся вывод, а также меняется картинка в соответствии с ответом. Далее, если в данной категории не будет конечного ответа, то тогда идёт вывод на интерфейсе пользователя сообщение о том, что в данной категории ответов нет. Дальше представлен код отвечающий за выведение на интерфейс пользователя вопрос и варианты ответов.

Листинг 3 – часть программного кода модуля es\_bin.js

const print\_dialog = function (post) {

document

.querySelectorAll('.line')

.forEach(line => line.hidden = true);

let answers = es.dict[post];

if (typeof answers === 'undefined') {

run = false;

\_quest.innerHTML = `Выбор сделан -> ${post.toUpperCase()}`

document.getElementById('img').src = "img/" + post + ".jpg";

}

else {

if (answers.length === 0) {

run = false;

\_quest.innerHTML = `Для категории "${post}" нет выбора.`;

}

else {

\_quest.innerHTML = `Сделайте выбор из категории<br>"${post}":`;

answers

.forEach((answer, index) => {

document.querySelector('#answer' + String(index)).innerHTML = answer;

document.querySelector('.line' + String(index)).hidden = false;

});

}

}

}

Интерфейс приложения разработан с помощью языка гипертекстовой разметки HTML и каскадных таблиц стилей CSS. HTML позволяет создавать и структурировать разделы, параграфы, заголовки, ссылки и блоки для веб-страниц и приложений.

HTML не является языком программирования, то есть он не имеет возможности создавать динамические функции.

При работе с HTML используются простые структуры кода (теги и атрибуты), чтобы разметить страницу веб-сайта.

CSS служит для описания оформления внешнего вида документа, созданного с использованием языка разметки (HTML, XHTML, XML).

Назначение CSS – отделять то, что задает внешний вид страницы, от ее содержания. Если документ создан только с использованием HTML, то в нем определяется не только каждый элемент, но и способ его отображения (цвет, шрифт, положение блока и т. д.). Если же подключены каскадные таблицы стилей, то HTML описывает только очередность объектов. А за все их свойства отвечает CSS. В HTML достаточно прописывать класс, не перечисляя все стили каждый раз. Пример использования HTML представлен на листинге 4

Листинг 4 – пример программного кода модуля index.htnl

<body>

<br><div id="title"></div>

<br><div class="help" id="help"></div>

<table id="dialog">

<tr id="reload">

<td class="quest" id="quest"></td>

</tr>

Листинг 6 – пример программного кода модуля es.css

body {

background-color: rgb(25, 41, 22);

color: #dd7;

font-family: "geneva";

font-size: 24px;

}

table {

width: 70%;

background-color: rgb(37, 80, 37);

border-radius: 30px 30px 25px 25px;

margin: auto;

}

td {

padding-left: 15px;

height: 100px;

border-color: grey;

border-width: 1px;

border-style: solid;

border-radius: 30px;

text-align: center;

}

заключение

Экспертная система - компьютерная система, способная частично заменить специалиста-эксперта в разрешении проблемной ситуации.

Экспертные системы сыскали широкое применение в медицине, математике, химии, геологии, а также в образовании и тестирование.

В ходе выполнения курсового проекта мною было проанализирована информация о технике присутствующая в игре World of Tanks и на основе информации была построена база знаний экспертной системы. Были изучены методы построения базы знаний экспертной системы. Разработана модель базы знаний экспертной системы «Выбор танка в World of Tanks». В процессе написания курсовой работы были изучены механизмы логического вывода. Экспертная система была разработана в виде веб-приложения на языках JavaScript, HTML, CSS.

Экспертная система показала свою работоспособность и эффективность в рамках определённой темы. Но у данной программы так же есть и недостатки, к ним можно отнести малый объём базы знаний и простой дизайн. Для введения проекта в работу необходимо как можно больше расширить базу знаний и разработать более привлекательный дизайн.

Список литературы

1. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский - СПб: Питер, 2000 - 384 с.;
2. Гаскаров, Д.Б. Интеллектуальные информационные системы. - М.: Высшая школа, 2003.;
3. Интернет ресурс – Танковедение URL : https://worldoftanks.ru/ru/tankopedia/#wot&w\_m=tanks (дата обращения 21.01.2022);
4. Интернет ресурс – Браузер: документ, события, интерфейсы URL: https://learn.javascript.ru/ui (дата обращения 21.01.2022);
5. Интернет ресурс – Справочник CSS URL: http://htmlbook.ru/css/help (дата обращения 21.01.2022);

Приложение

Приложение 1 – Структура хранения данных JSON

let es = {

"title": "Выбор танка в WoT",

"start": "Какую нацию выберите?",

"dict": {

"Какую нацию выберите?": ["СССР", "Германия", "США",],

"СССР": ["Лёгкий Танк","Средний Танк","Тяжёлый Танк", "Противотанковая Сау"],

"Германия": ["Лёгкие танки","Средние танки","Тяжёлые танки", "Противотанковые сау"],

"США": ["лёгкий танк","средний танк","тяжёлый танк", "противотанковая сау"],

"Лёгкий Танк": ["Т-100 ЛТ"],

"Средний Танк": ["Прокачиваемые", "Премиумные"],

"Тяжёлый Танк": ["Прокачиваемый", "Премиумный"],

"Противотанковая Сау": ["Объект 268", "Объект 268 Вариант 4"],

"Лёгкие танки": ["RHEINMETALL PANZERWAGEN"],

"Средние танки": ["E 50 Auf. M","Leopard 1"],

"Тяжёлые танки": ["Maus","E 100","Pz.Kpfw. VII"],

"Противотанковые сау": ["Jagdpanzer E100", "Grille 15"],

"лёгкий танк": ["XM551 Sheridan"],

"средний танк": ["M48A5 Patton"],

"тяжёлый танк": ["T110E5", "T57 Heavy Tank"],

"противотанковая сау": ["T110E3", "T110E4"],

"Прокачиваемые": ["Т-62А", "Объект 140", "Объект 430У", "К-91"],

"Премиумные": ["Объект 907", "Т-22 СР"],

"Прокачиваемый": ["ИС-4", "ИС-7", "Объект 705А", "Объект 277", "СТ-2"],

"Премиумный": ["Объект 279", "Объект 260"],

}

}

Приложение 2 – Текст программы на языке JavaScript

let colors = ['', '#777'];

let run;

let \_quest = document.querySelector('#quest');

let \_title = document.querySelector('#title');

let \_help = document.querySelector('#help');

const init = function () {

run = true;

\_title.innerHTML = es.title;

\_help.innerHTML = 'Кликни на картинку для сброса.';

print\_dialog(es.start);

};

const print\_dialog = function (post) {

document

.querySelectorAll('.line')

.forEach(line => line.hidden = true);

let answers = es.dict[post];

if (typeof answers === 'undefined') {

run = false;

\_quest.innerHTML = `Выбор сделан -> ${post.toUpperCase()}`

document.getElementById('img').src = "img/" + post + ".jpg";

}

else {

if (answers.length === 0) {

run = false;

\_quest.innerHTML = `Для категории "${post}" нет выбора.`;

}

else {

\_quest.innerHTML = `Сделайте выбор из категории<br>"${post}":`;

answers

.forEach((answer, index) => {

document.querySelector('#answer' + String(index)).innerHTML = answer;

document.querySelector('.line' + String(index)).hidden = false;

});

}

}

}

Document

.addEventListener("DOMContentLoaded", init);

document

.querySelectorAll('#dialog .answer')

.forEach(td\_answer => {

td\_answer.addEventListener("click", () => print\_dialog(td\_answer.innerHTML));

td\_answer.addEventListener('mouseenter', () => td\_answer.style.backgroundColor = colors[1]);

td\_answer.addEventListener('mouseleave', () => td\_answer.style.backgroundColor = colors[0]);});

Приложение 3 – Оформление страницы CSS

body {

background-color: rgb(25, 41, 22);

color: #dd7;

font-family: "geneva";

font-size: 24px;

}

table {

width: 70%;

background-color: rgb(37, 80, 37);

border-radius: 30px 30px 25px 25px;

margin: auto;

}

td {

padding-left: 15px;

height: 100px;

border-color: grey;

border-width: 1px;

border-style: solid;

border-radius: 30px;

text-align: center;

}

td.quest {

height: 90px;

background-color: rgb(40, 100, 40);

color: rgb(255, 0, 0);

}

td.answer {

height: 30px;

}

div {

text-align: center;

}

div.help {

color: rgb(146, 245, 146);

font-size: 18px;

margin-bottom: 4px;

}

Приложение 4 – Разметка страницы HTML

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Экспертная система</title>

<link rel="stylesheet" href="es.css">

</head>

<body>

<br><div id="title"></div>

<br><div class="help" id="help"></div>

<table id="dialog">

<tr id="reload">

<td class="quest" id="quest"></td>

</tr>

<tr class="line line0">

<td class="answer" id="answer0"></td>

</tr>

<tr class="line line1">

<td class="answer" id="answer1"></td>

</tr>

<tr class="line line2">

<td class="answer" id="answer2"></td>

</tr>

<tr class="line line3">

<td class="answer" id="answer3"></td>

</tr>

<tr class="line line4">

<td class="answer" id="answer4"></td>

</tr>

<tr class="line line5">

<td class="answer" id="answer5"></td>

</tr>

</table>

<div>

<br>

<img src="img/1.jpg" id="img" onclick="window.location.reload()"/>

</div>

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

<script src="es\_bin.js"></script>

</body>

</html>

РЕЦЕНЗИЯ

на курсовой проект \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

доцент кафедры ИТиПИ, к.т.н., доцент Беляков Андрей Юрьевич