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 «Выбор автомобиля».

Выполнил:

студент 3-го курса очного отделения

специальности 09.03.03 Прикладная информатика

шифр ПИб-681-2019

Дашковский Игнат Николаевич

Проверил:

доцент кафедры ИТиПИ, к.т.н., доцент

Беляков Андрей Юрьевич

Пермь – 2022

**Содержание**

[1. Постановка задачи 3](#_Toc93407631)

[2. Структура хранения данных 4](#_Toc93407632)

[3. Разработка приложения 6](#_Toc93407633)

[Заключение 9](#_Toc93407634)

[Список литературы 10](#_Toc93407635)

[Приложения 11](#_Toc93407636)

# Постановка задачи

В автомобильный салон приходит покупатель, желающий приобрести автомобиль, он представляет, что ему требуется от автомобиля и какие характеристики должны быть. Сотрудник автосалона знает характеристики всех автомобилей, т.е. является экспертом и может подобрать для покупателя автомобиль. Для этого ему необходимо задать покупателю такие вопросы, ответы на которые позволят сделать выбор модели автомобиля, чтобы удовлетворить его функционал. Таким образом необходимо разработать ЭС, которая будет содержать знания специалиста автосалона и поможет выбрать автомобиль для покупателя. Пользователями такой системы будут покупатели или начинающий специалист отдела продаж.

Рисунок 1. Стартовая форма

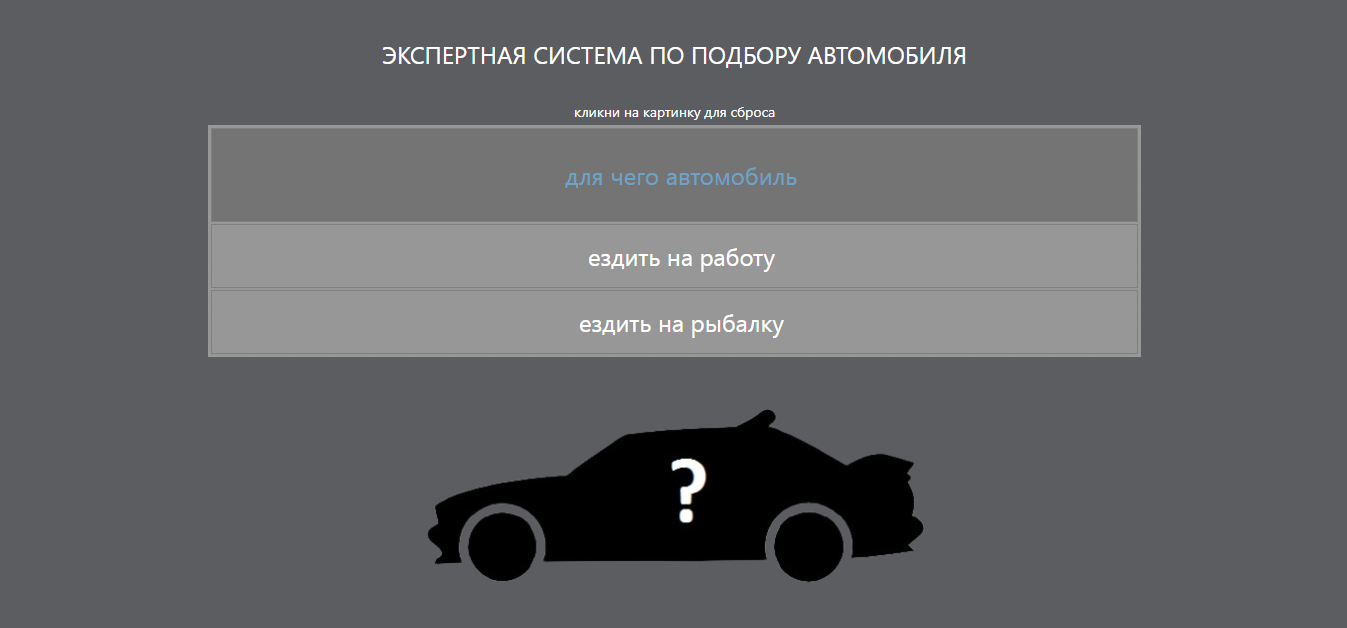


Рисунок 2. Результат



# Структура хранения данных

Структура хранения данных разработана в формате JSON. JSON – это сокращение от JavaScript Object Notation — формата передачи данных. Как можно понять из названия, JSON произошел из JavaScript, но он доступен для использования на многих других языках, включая Python, Ruby, PHP и Java.

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

Листинг 1. Пример кода JSON

    "title": "ЭКСПЕРТНАЯ СИСТЕМА ПО ПОДБОРУ АВТОМОБИЛЯ",

    "start": "для чего автомобиль",

    "dict": {

        "для чего автомобиль": ["ездить на работу", "ездить на рыбалку"],

        "ездить на работу": ["автоматическая кпп", "ручная кпп"],

        "автоматическая кпп": ["полный привод", "передний привод"],

        "полный привод": ["Volkswagen Tiguan", "Toyota RAV4", "Nissan Qashqai"],

        "передний привод": ["Volkswagen Polo", "Hundai Solaris", "Kia Rio"],

        "ручная кпп": ["седан", "хэтчбек", "универсал"],

        "седан": ["Лада Приора", "Nissan Almera", "Toyota Corolla"],

        "хэтчбек": ["Volkswagen Golf", "Nissan Note", "ВАЗ-2114"],

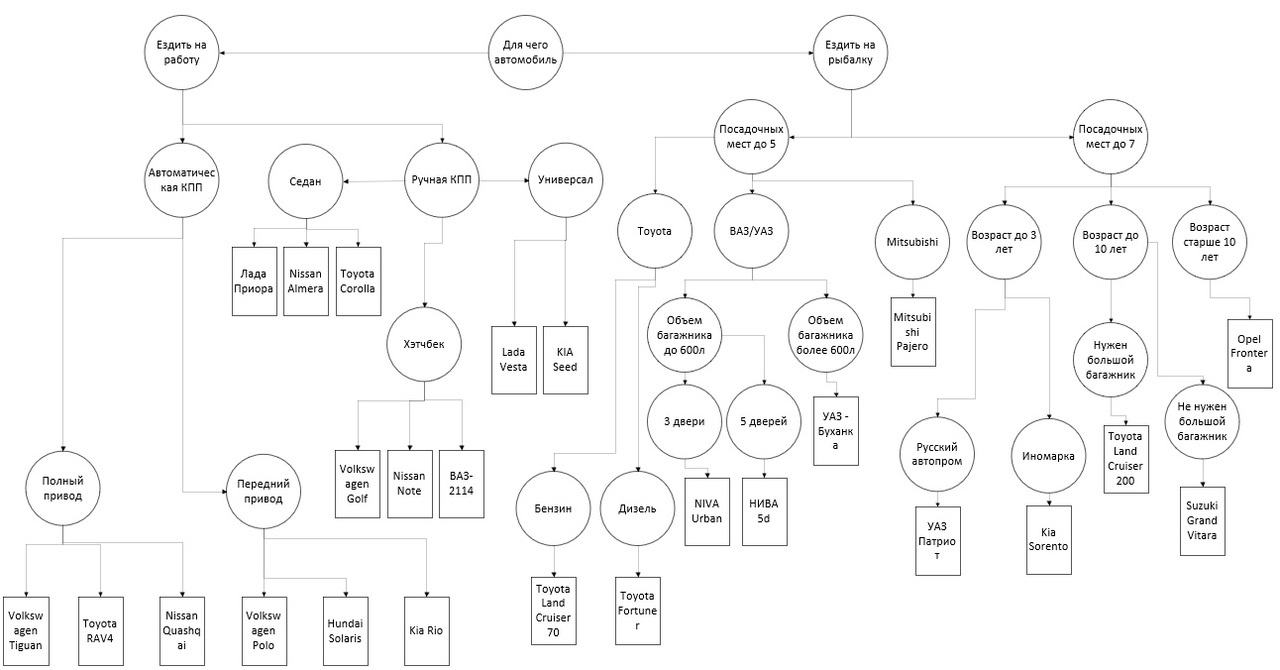
        "универсал": ["Lada Vesta", "KIA Seed"]

}

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".

Дерево решений экспертной системы представлено на Рисунке 3.

Рисунок 3. Дерево решений



# Разработка приложения

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.

Листинг 2. Пример кода 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>

</body>

</html>

Листинг 3. Пример кода CSS

body {

    background-color: #5c5d61;

    color: white;

    font-family: "system-ui";

    font-size: 24px;

}

table {

    width: 70%;

    background-color: #979797;

    border-width: 1px;

    border-style: dashed;

    border-color: #9b9b9b;

    margin: auto;

}

td {

    padding-left: 15px;

    height: 100px;

    border-color: grey;

    border-width: 1px;

    border-style: solid;

    text-align: center;

}

Избирательное отображение данных реализовано на языке JavaScript. JavaScript – это мультипарадигменный язык программирования, который обычно применяется в качестве встраиваемого инструмента для программного доступа к различным объектам приложений. С точки зрения веб-разработки, без знаний этой технологии невозможно заниматься созданием современных интерактивных сайтов.

Программы на этом языке называются скриптами. Они могут встраиваться в HTML и выполняться автоматически при загрузке веб-страницы. Скрипты распространяются и выполняются, как простой текст. Им не нужна специальная подготовка или компиляция для запуска.

Программа печатает первый вопрос и заголовок из JSON файла, которые обозначены тэгами “start” и “title” соответственно, а также подсказку (листинг 4).

Листинг 4. Пример кода отображения данных на JavaScript

const init = function () {

    run = true;

    \_title.innerHTML = es.title;

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

    print\_dialog(es.start);

};

Вопросами экспертной системы являются ключи JSON файла, а варианты ответов – значения этих ключей.

Программа узнает текущий ответ на вопрос и выдает следующий вопрос(ключ) с таким названием и его варианты ответа(значения), до тех пор, пока пользователь не дойдет до того варианта ответа, у которого не будет одноименного вопроса(ключа). Этот вариант и будет конечным выбором (листинг 5)

Листинг 5. Пример кода избирательного

отображения данных на JavaScript

const print\_dialog = function (post) {

    document

        .querySelectorAll('.line')

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

    let answers = es.dict[post];

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

        run = false; // остановить работу ЭС

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

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

    }

    else {

        if (answers.length === 0) {

            run = false; // остановить работу ЭС

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

        }

        else {

            \_quest.innerHTML = `${post}`;

            answers

                .forEach((answer, index) => {

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

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

                });

# Заключение

Экспертные системы – это яркое и быстро прогрессирующее направление в области искусственного интеллекта. Главным достоинством экспертных систем является возможность накопления знаний и сохранение их длительное время. В отличие от человека к любой информации экспертные системы подходят объективно, что улучшает качество проводимой экспертизы.

Причиной повышенного интереса, который ЭС вызывают к себе является возможность их применения к решению задач из самых различных областей человеческой деятельности. Широкий спектр применения ЭС позволяет сделать вывод о том, что это перспективное направление развития в области ИИ и программирования.

В курсовой работе была проанализирована информация об автомобилях, как основа базы знаний ЭС. Были изучены методы построения базы знаний ЭС. Разработана модель базы знаний ЭС «Выбор автомобиля». В процессе написания курсовой работы были изучены механизмы логического вывода. ЭС была разработана в виде веб-приложения на языках JavaScript, HTML, CSS.

Результаты разработанного проекта в рамках определенной темы показали ее эффективность и работоспособность.

Необходимо отметить так же недостатки программы, к ним можно отнести малый объем базы знаний и простой дизайн. Для внедрения данного проекта в работу необходимо значительно расширить количество параметров и их вариативность и разработать более привлекательный для пользователя 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. Джексон, П. Введение в экспертные системы / П. Джексон. - 3-е изд. - М., 2007.

4. Муромцев Д.И. Введение в технологию экспертных систем. СПб: СПб ГУ ИТМО, 2005.

5. Тельнов Ю.Ф. Интеллектуальные информационные системы в экономике. - Уч. пособие. - М.: Синтег, 1998. - 216 с.

6. Убейко, В.Н. Экспертные системы.- М.: МАИ, 1992.

# Приложения

Приложение 1. Разметка страниц 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>

    <br><img src="img/0.png" id="img" onClick="window.location.reload()"> </br>

    <script src="data.js"></script> <!--база знаний-->

    <script src="es\_multi.js"></script> <!--машина вывода-->

</body>

</html>

Приложение 2. Стили оформления CSS

body {

    background-color: #5c5d61;

    color: white;

    font-family: "system-ui";

    font-size: 24px;

}

table {

    width: 70%;

    background-color: #979797;

    border-width: 1px;

    border-style: dashed;

    border-color: #9b9b9b;

    margin: auto;

}

td {

    padding-left: 15px;

    height: 100px;

    border-color: grey;

    border-width: 1px;

    border-style: solid;

    text-align: center;

}

td.quest {

    height: 90px;

    background-color: #747474;

    color: #6ca3c7;

}

td.answer {

    height: 60px;

}

div {

    text-align: center;

}

div.help {

    color: #ffffff;

    font-size: 14px;

    margin-bottom: 4px;

}

img {

    border-radius: 30px;

    display: block;

    margin: 0 auto;

    width: 500px;

}

Приложение 3. Структура хранения данных JSON

    "title": "ЭКСПЕРТНАЯ СИСТЕМА ПО ПОДБОРУ АВТОМОБИЛЯ",

    "start": "для чего автомобиль",

    "dict": {

        "для чего автомобиль": ["ездить на работу", "ездить на рыбалку"],

        "ездить на работу": ["автоматическая кпп", "ручная кпп"],

        "автоматическая кпп": ["полный привод", "передний привод"],

        "полный привод": ["Volkswagen Tiguan", "Toyota RAV4", "Nissan Qashqai"],

        "передний привод": ["Volkswagen Polo", "Hundai Solaris", "Kia Rio"],

        "ручная кпп": ["седан", "хэтчбек", "универсал"],

        "седан": ["Лада Приора", "Nissan Almera", "Toyota Corolla"],

        "хэтчбек": ["Volkswagen Golf", "Nissan Note", "ВАЗ-2114"],

        "универсал": ["Lada Vesta", "KIA Seed"],

        "ездить на рыбалку": ["посадочных мест до 5", "посадочных мест до 7"],

        "посадочных мест до 5": ["Toyota", "ВАЗ/УАЗ", "Mitsubishi"],

        "Toyota" : ["бензин", "дизель"],

        "бензин": ["Toyota Land Cruiser 70"],

        "дизель": ["Toyota Fortuner"],

        "ВАЗ/УАЗ": ["объем багажника до 600л", "объем багажника более 600л"],

        "объем багажника до 600л": ["3 двери", "5 дверей"],

        "3 двери": ["NIVA Urban"],

        "5 дверей": ["НИВА 5d"],

        "объем багажника более 600л": ["УАЗ - Буханка"],

        "Mitsubishi": ["Mitsubishi Pajero"],

        "посадочных мест до 7": ["возраст до 3 лет", "возраст до 10 лет", "возраст старше 10 лет"],

        "возраст до 3 лет": ["русский автопром", "иномарка"],

        "русский автопром": ["УАЗ Патриот"],

        "иномарка": ["Kia Sorento"],

        "возраст до 10 лет": ["нужен большой багажник", "не нужен большой багажник"],

        "нужен большой багажник": ["Toyota Land Cruiser 200"],

        "не нужен большой багажник": ["Suzuki Grand Vitara"],

        "возраст старше 10 лет": ["Opel Frontera"]

    }

Приложение 4. Текст программы на языке JavaScript

let colors = ['#979797', '#7395ae']; // массив цветов выкл/вкл

let run; // запущено или остановлено приложение

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

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

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

// functions

const init = function () { // инициализация

    run = true; // запускаем приложение

    \_title.innerHTML = es.title;

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

    print\_dialog(es.start); // печатаем первый вопрос и ответы

};

const print\_dialog = function (post) { // варианты ответов

    document

        .querySelectorAll('.line') // все по имени класса

        .forEach(line => line.hidden = true); // делаем скрытыми

    let answers = es.dict[post]; // узнаем текущие ответы

    if (typeof answers === 'undefined') { // если уже достигнут листок дерева

        run = false; // остановить работу ЭС

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

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

    }

    else {

        if (answers.length === 0) { // если нет ответов на вопрос

            run = false; // остановить работу ЭС

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

        }

        else {  // публикуем вопрос и варианты ответов

            \_quest.innerHTML = `${post}`;

            answers

                .forEach((answer, index) => {

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

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

                });

        }

    }

}

// event handlers

document // при загрузке страницы

    .addEventListener("DOMContentLoaded", init);

document // при клике по ячейке таблицы

    .querySelectorAll('#dialog .answer') // найти массив ячеек таблицы с ответами

    .forEach(td\_answer => { // для каждой ячейки назначить обработчики событий

        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]);

    });

РЕЦЕНЗИЯ

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

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

доцент кафедры ИТиПИ, к.т.н., доцент

Беляков Андрей Юрьевич