

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего
образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им.
Р.Е.АЛЕКСЕЕВА.

Институт радиоэлектроники и информационных технологий

Отчет

по лабораторной работе № 3

“Реализация проекта многопользовательской системы интеллектуальной поддержки
с использованием технологии Клиент-сервер”

по дисциплине

«Основы теории интеллектуальных вычислительных систем»

Руководитель:

_____ Мисевич П. В.

Студент:

_____ Игнаков К. М.

_____ Тихонов В. Д.

группа 19-ВМ

Работа защищена «__» _____

С оценкой _____

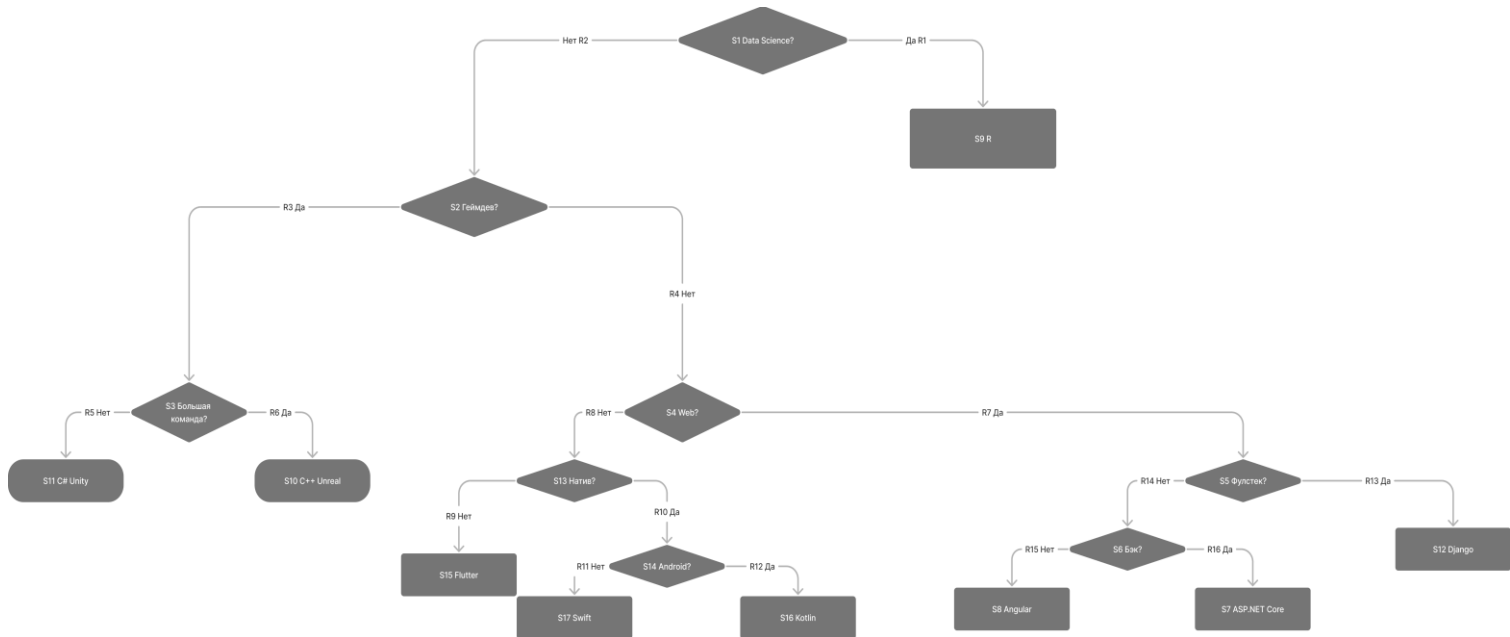
Нижний Новгород 2022

Постановка задачи

Реализовать многопользовательскую систему интеллектуальной поддержки с применением технологии клиент-сервер.

Решение задачи

Дерево правил:



Средствами СУБД MS SQL создадим бд OTIVS, в которой создадим таблицу Events, которой будут храниться все действия:

Имя столбца	Тип данных	Разрешить ...
ID	int	<input type="checkbox"/>
SI	nvarchar(MAX)	<input checked="" type="checkbox"/>
Ind_Start	bit	<input checked="" type="checkbox"/>
Ind_Stop	bit	<input checked="" type="checkbox"/>
Memo	nvarchar(MAX)	<input checked="" type="checkbox"/>

ID	SI	Ind_Start	Ind_Stop	Memo
1	Data Science?	True	False	NULL
2	Game-dev?	False	False	NULL
3	Big company?	False	False	NULL
4	Web?	False	False	NULL
5	Fullstack devel...	False	False	NULL
6	Back-end dev?	False	False	NULL
7	ASP.NET Core	False	True	NULL
8	Angular	False	True	NULL
9	R	False	True	NULL
10	C++ Unreal	False	True	NULL
11	C# Unity	False	True	NULL
12	Django	False	True	NULL
13	Native?	False	False	NULL
14	Android?	False	False	NULL
15	Flutter	False	True	NULL
16	Kotlin	False	True	NULL
17	Swift	False	True	NULL
**	NULL	NULL	NULL	NULL

- | | | | | |
|---|----------------------|-------|-------|------|
| 1 | Data Science? | True | False | NULL |
| 2 | Game-dev? | False | False | NULL |
| 3 | Big company? | False | False | NULL |
| 4 | Web? | False | False | NULL |
| 5 | Fullstack developer? | False | False | NULL |
| 6 | Back-end dev? | False | False | NULL |

7	ASP.NET Core	False	True	NULL
8	Angular	False	True	NULL
9	R	False	True	NULL
10	C++ Unreal	False	True	NULL
11	C# Unity	False	True	NULL
12	Django	False	True	NULL
13	Native?	False	False	NULL
14	Android?	False	False	NULL
15	Flutter	False	True	NULL
16	Kotlin	False	True	NULL
17	Swift	False	True	NULL

Создадим таблицу Rools, в которой будет храниться логика перехода между вопросами в СИП:

Результаты				Сообщения	
	ID	Si	Rj	Sk	Memo
1	1	1	1	9	NULL
2	2	1	0	2	NULL
3	3	2	1	3	NULL
4	4	2	0	4	NULL
5	5	3	1	10	NULL
6	6	3	0	11	NULL
7	7	4	1	5	NULL
8	8	4	0	13	NULL
9	9	5	1	12	NULL
10	10	5	0	6	NULL
11	11	6	1	7	NULL
12	12	6	0	8	NULL
13	13	13	1	10	NULL
14	14	13	0	15	NULL
15	15	14	1	16	NULL

Имя столбца	Тип данных	Разрешить ...
ID	int	<input type="checkbox"/>
Si	int	<input type="checkbox"/>
Rj	int	<input type="checkbox"/>
Sk	int	<input type="checkbox"/>
Memo	nvarchar(MAX)	<input checked="" type="checkbox"/>

Таким образом, мы формируем информационное обеспечение работы нашей СИП

Рассмотрим клиентское приложение. При загрузке формы инициализируется подключение к серверу:

```

/SELECT      ID, SI, Ind_Stop FROM          dbo.Events WHERE          (Ind_Stast = 1)
DataSet ds = new DataSet();// создаем объект DataSet ds = new DataSet();// создаем объект
string connectionString = @"Data Source=localhost;Initial Catalog=OTIVS;User
Id=sa;Password=QZWXECRVasdf1234!";// создаем строку подключения

string commandString = "SELECT      ID, SI, Ind_Stop FROM          dbo.Events WHERE
(Ind_Start = 1)";// создаем запрос

SqlDataAdapter adapter = new SqlDataAdapter(commandString, connectionString);// создаем
SqlDataAdapter adapter

```

Текст и Si берется из таблицы Events и отображается в графическом интерфейсе, как во второй лабораторной работе:

The screenshot shows a graphical user interface for defining a rule. At the top, there is a text input field containing "Data Science?". Below it, the rule structure is displayed: "IF S" followed by a text box containing the number "1", then "AND R" followed by an empty text box, and finally "THEN S" followed by another empty text box. A red circle is drawn around the text box containing "1".

При нажатии на кнопку «Да» или «Нет» будет сформировано правило по которому мы перейдем на другой вопрос из нашего дерева СИП:

The image contains two screenshots of the rule editor interface, illustrating the flow between different rules in a decision tree.

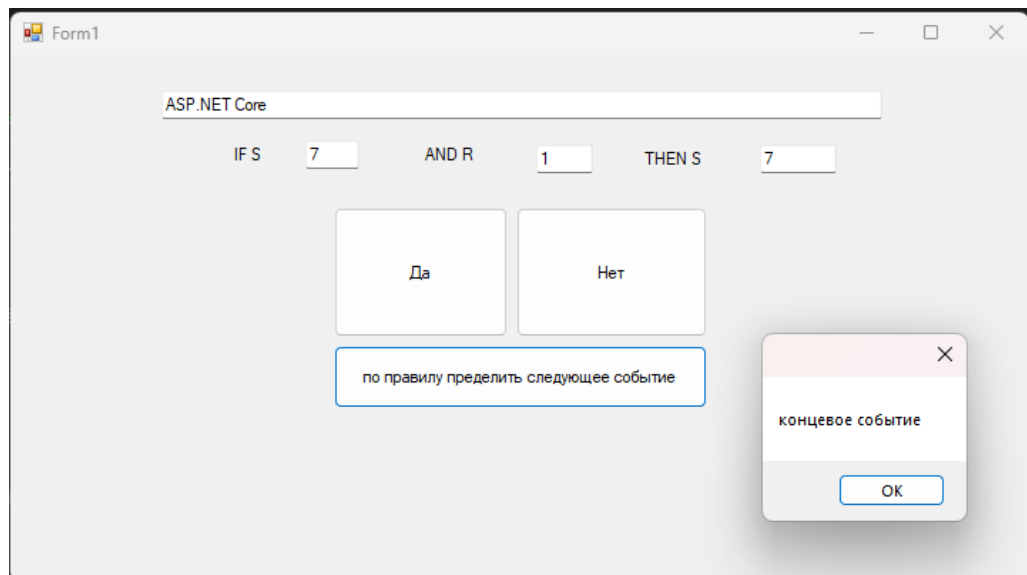
The top screenshot shows a rule for "Data Science?". The rule structure is: IF S [1] AND R [0] THEN S []. Below the rule structure, there are two buttons: "Да" (Yes) and "Нет" (No). A red box highlights these two buttons. Below the buttons is a button labeled "по правилу определить следующее событие" (Determine the next event by rule). Red arrows indicate the flow: one arrow points from the "Да" button to the "R" field (0), and another arrow points from the "Нет" button to the "S" field ().

The bottom screenshot shows a rule for "Game-dev?". The rule structure is: IF S [2] AND R [0] THEN S [2]. Below the rule structure, there are two buttons: "Да" (Yes) and "Нет" (No). A red box highlights these two buttons. Below the buttons is a button labeled "по правилу определить следующее событие" (Determine the next event by rule). Red arrows indicate the flow: one arrow points from the "Да" button to the "S" field (2), and another arrow points from the "Нет" button to the "S" field (2). A large red "1." is written on the left side of the screenshot.

Когда клиент нажимает на кнопку «по правилу определить следующее событие» идет поиск к таблице Rools с соответствующими Si и Rj:

```
string commandString = "SELECT TOP (100) PERCENT ID, Si, Rj, Sk, Memo FROM  
dbo.Rools WHERE (Si =" + Si__ + ") AND (Rj =" + Rj__ + ")";
```

Концевое событие определяется полем «Ind_Stop» в таблице «Events»



Исходный код

```
using System;
using System.Data;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace ЛР2_Client_Server
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            //SELECT      ID, SI, Ind_Stop FROM          dbo.Events WHERE
            (Ind_Stast = 1)
            DataSet ds = new DataSet();// создаем объект DataSet ds = new DataSet();//
            создаем объект
            string connectionString = @"Data Source=localhost;Initial Catalog=OTIVS;User
            Id=sa;Password=QZWXECRVasdf1234!";// создаем строку подключения

            string commandString = "SELECT      ID, SI, Ind_Stop FROM
            dbo.Events WHERE          (Ind_Start = 1)";// создаем запрос

            SqlDataAdapter adapter = new SqlDataAdapter(commandString,
            connectionString);// создаем SqlDataAdapter adapter

            adapter.Fill(ds); // заполнение DataSet данными с помощью DataAdapter

            // узнаем число строк в объекте ds.Tables[0]
            int rows_ = 0;
            rows_ = ds.Tables[0].Rows.Count;
            if (rows_ > 0)
            {
                DataRow row = ds.Tables[0].Rows[0]; ; //в объект DataRow записали j-й
                ряд таблицы DataTable
                string ii_str; // задали строковую переменную
            }
        }
    }
}
```

```

        ii_str = row["SI"].ToString();// значение поля до изменения
        textBox1.Text = ii_str;
        ii_str = row["ID"].ToString();// значение поля до изменения
        textBox3.Text = ii_str;
    } //if (rows_ > 0)
}

private void textBox2_TextChanged(object sender, EventArgs e)
{
}

private void button1_Click(object sender, EventArgs e)
{
    textBox2.Text = "1";
}

private void button2_Click(object sender, EventArgs e)
{
    textBox2.Text = "0";
}

private void button3_Click(object sender, EventArgs e)
{
    //
    DataSet ds = new DataSet();// создаем объект DataSet ds = new DataSet();//
создаем объект
    string Rj__; // задали строковую переменную
    //Si
    string Si__; // задали строковую переменную
    Rj__ = textBox2.Text;
    Si__ = textBox3.Text;
    string connectionString = @"Data Source=localhost;Initial Catalog=OTIVS;User
Id=sa;Password=QZWXECRVasdf1234!";// создаем строку подключения
    string commandString = "SELECT TOP (100) PERCENT ID, Si, Rj, Sk, Memo
FROM      dbo.Rools WHERE      (Si =" + Si__ + ") AND (Rj =" + Rj__ + ")";//
создаем запрос
    SqlDataAdapter adapter = new SqlDataAdapter(commandString,
connectionString);// создаем SqlDataAdapter adapter
    adapter.Fill(ds); // заполнение DataSet данными с помощью DataAdapter
    // узнаем число строк в объекте ds.Tables[0]
    int rows_ = 0;
    rows_ = ds.Tables[0].Rows.Count;
    if (rows_ > 0)
    {
        //if (rows_ > 0)
        DataRow row = ds.Tables[0].Rows[0]; ; //в объект DataRow записали j-й ряд
таблицы DataTable
        string ii_str; // задали строковую переменную
        ii_str = row["Sk"].ToString();// значение поля до изменения
        textBox4.Text = ii_str;

    } //if (rows_ > 0)
    button4_Click(sender, e);
}

private void button4_Click(object sender, EventArgs e)
{
    //SELECT ID, SI, Ind_Stop FROM      dbo.Events WHERE
(Ind_Stast = 1)
    DataSet ds = new DataSet();// создаем объект DataSet ds = new DataSet();//
создаем объект
    string connectionString = @"Data Source=localhost;Initial Catalog=OTIVS;User
Id=sa;Password=QZWXECRVasdf1234!";// создаем строку подключения

```

```

        string S__NEXT; // задали строковую переменную
        string STOP_; // задали строковую переменную

        S__NEXT = textBox4.Text;

        string commandString = "SELECT          ID, SI, Ind_Stop FROM
dbo.Events WHERE          (ID = " + S__NEXT + ")"; // создаем запрос

        SqlDataAdapter adapter = new SqlDataAdapter(commandString,
connectionString); // создаем SqlDataAdapter adapter

        adapter.Fill(ds); // заполнение DataSet данными с помощью DataAdapter

        //tb4

        // узнаем число строк в объекте ds.Tables[0]
        int rows_ = 0;
        rows_ = ds.Tables[0].Rows.Count;
        if (rows_ > 0)
        { //if (rows_ > 0)
            DataRow row = ds.Tables[0].Rows[0]; ; //в объект DataRow записали j-й ряд
таблицы DataTable
            string ii_str; // задали строковую переменную
            ii_str = row["SI"].ToString(); // значение поля до изменения
            textBox1.Text = ii_str;
            ii_str = row["ID"].ToString(); // значение поля до изменения
            textBox3.Text = ii_str;
            ii_str = row["Ind_Stop"].ToString();
            STOP_ = ii_str;
            if (STOP_ == "True")
            {
                MessageBox.Show("концевое событие");
            }

        } //if (rows_ > 0)
    }
}
}

```

Решение задачи

В ходе выполнения лабораторной работы было реализовано клиент-серверное приложение, положительная сторона: благодаря подходу клиент-сервер в пользовательской программе нет необходимости обновлять дерево СПИ, так как она расположена в БД. Минусы: человеческий фактор – ошибка при заполнении таблицы Roools может возникнуть если пользователь неверно поставит Id, решением для такой проблемы будет сделать составной ключ для каждой записи, так как это даст проверку что администратор базы данных или иной редактор СПИ не занес в таблице две одинаковые записи.