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 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

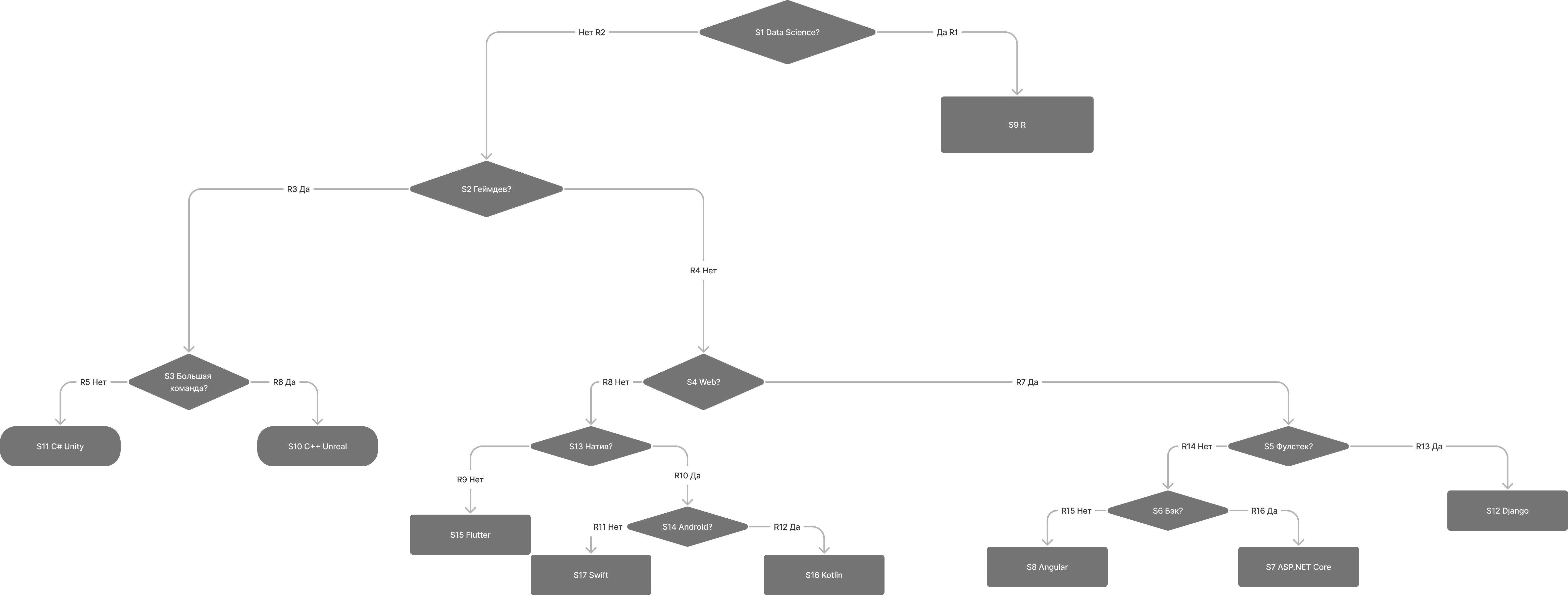
Нижний Новгород 2022

## Постановка задачи

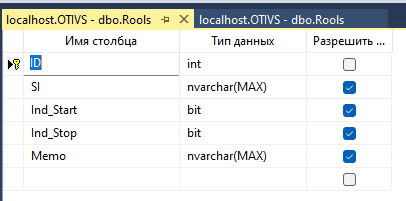
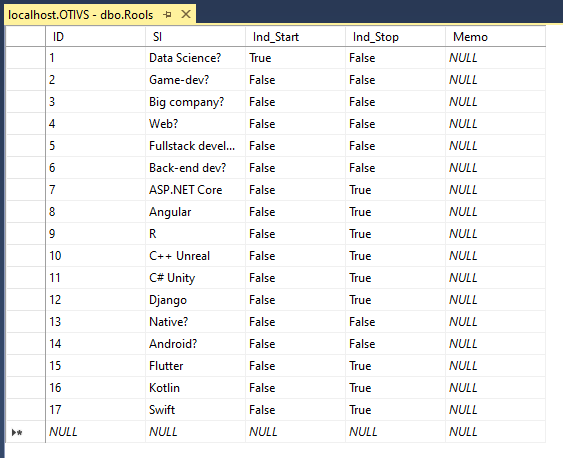
Реализовать многопользовательскую систему интеллектуальной поддержки с применением технологии клиент-сервер.

## Решение задачи

Дерево правил:

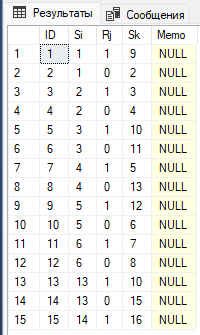
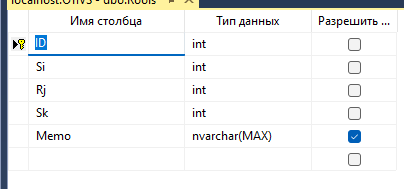


Средствами СУБД MS SQL создадим бд OTIVS, в которой создадим таблицу Events, которой будут храниться все действия:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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:



Таким образом, мы формируем информационное обеспечение работы нашей СИП

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

/SELECT ID, Sl, Ind\_Stop FROM dbo.Events WHERE (Ind\_Stast = 1)

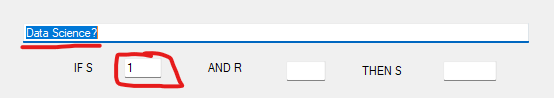
DataSet ds = new DataSet();// создаем объект DataSet ds = new DataSet();// создаем объект

string connectionString = @"Data Source=localhost;Initial Catalog=OTIVS;User Id=sa;Password=QZWXECRVasdf1234!;";// создаем строку подключения

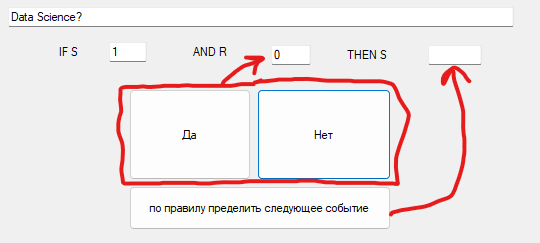
string commandString = "SELECT ID, SI, Ind\_Stop FROM dbo.Events WHERE (Ind\_Start = 1)";// создаем запрос

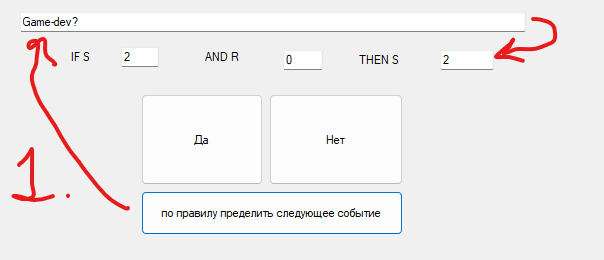
SqlDataAdapter adapter = new SqlDataAdapter(commandString, connectionString);// создаем SqlDataAdapter adapter

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



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

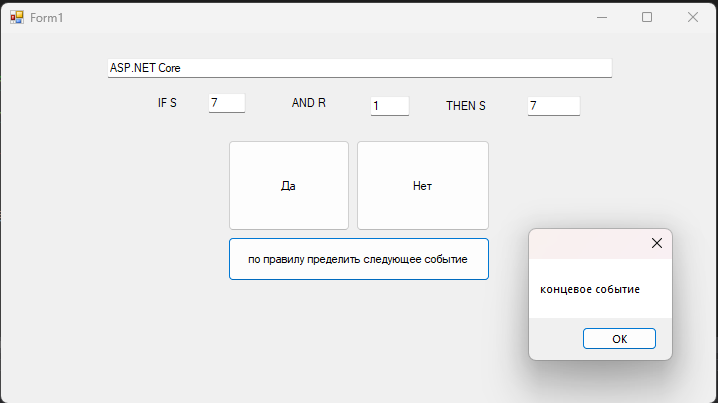




Когда клиент нажимает на кнопку «по правилу определить следующее событие» идет поиск к таблице Rools c соответствующими Si и Rj:

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, Sl, 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)

{//if (rows\_ > 0)

DataRow row = ds.Tables[0].Rows[0]; ; //в объект DataRow записали j-й ряд таблицы DataTable

string ii\_str; // задали стринговую 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 стринговую переменную

//Si

string Si\_\_; // задали стринговую 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 стринговую 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, Sl, 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 стринговую переменную

string STOP\_; // задали стринговую 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 стринговую 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 Rools 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.