ДОДАТОК А

Апробація результатів наукових досліджень


Primedia
eLaunch
PROCEEDINGS OF THE
III INTERNATIONAL SCIENTIFIC
AND THEORETICAL CONFERENCE
CURRENT SCIENTIFIC
GOALS, APPROACHES
AND CHALLENGES
17.01.2025
RIGA
REPUBLIC OF LATVIA
SCIENTIA
COLLECTION OF SCIENTIFIC PAPERS

SECTION 11.
GENERAL MECHANICS AND MECHANICAL ENGINEERING

SECTION 12.
AUTOMATION AND APPLIANCES MAKING

SECTION 13.
ELECTRONICS AND TELECOMMUNICATIONS

SECTION 14.
ENERGY AND POWER ENGINEERING

SECTION 15.
INFORMATION TECHNOLOGIES AND SYSTEMS

Ткачов Микита Русланович
здобувач вищої освіти факультету Автоматики і комп'ютеризованих технологій
Харківський національний університет радіоелектроніки, Україна

Науковий керівник: Новоселов Сергій Павлович
канд. техн. наук, доцент,
професор кафедри комп'ютерно-інтегрованих технологій автоматизації і робототехніки
Харківський національний університет радіоелектроніки, Україна

РОЗРОБЛЕННЯ ХМАРНОГО СЕРЕДОВИЩА ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ СТУДЕНТІВ

Сучасний світ активно переходить до цифрових технологій, а нещодавня пандемія суттєво прискорила процес впровадження дистанційного навчання. Водночас виникли нові виклики, пов'язані з доступністю, якістю та ефективністю навчальних процесів, що потребують інноваційних підходів.

Через воєнний стан багато студентів були змушені залишити свої домівки та переїхати до інших регіонів України або за кордон. Це призвело до серйозних перешкод у доступі до традиційних форм навчання. Хмарне середовище дозволяє студентам продовжувати навчання незалежно від їхнього місця перебування, що є критично важливим у таких умовах.

Загальна архітектура автоматизованої системи на основі хмарного середовища для дистанційного навчання студентів показана на (рис. 1).

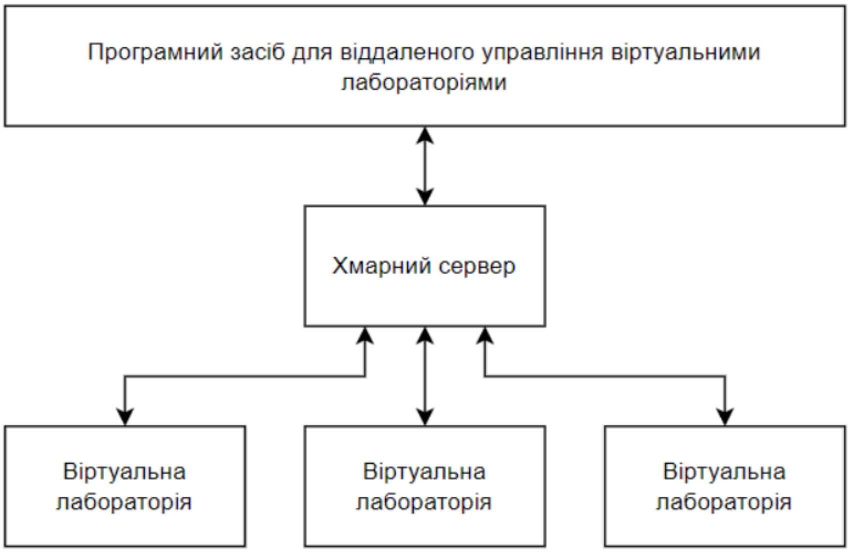

Рис. 1. **Загальна архітектура автоматизованої системи**

Архітектура хмарного середовища для дистанційного навчання студентів складається з трьох основних компонентів [1]:

– програмного засобу для віддаленого управління віртуальними лабораторіями;

– хмарного сервера;

– віртуальної лабораторії.

Зазначені компоненти взаємодіють між собою для забезпечення ефективної організації навчального процесу, що дозволяє студентам виконувати лабораторні завдання дистанційно, використовуючи ресурси хмарних технологій.

Хмарний сервер є центральним елементом цієї архітектури [1, 2]. Він відповідає за зберігання і обробку даних, забезпечує управління всіма підключеними компонентами та дозволяє користувачам взаємодіяти з системою в режимі реального часу. На хмарному сервері функціонують кілька програмних модулів, включаючи сервер управління базою даних (наприклад, PostgreSQL), що зберігає інформацію про користувачів, лабораторні завдання, результати та інші дані. Також на сервері встановлено програмний модуль моніторингу процесів, який забезпечує стабільність роботи системи, відстежує її продуктивність і контролює активність користувачів та виконання завдань.

Програмний засіб для віддаленого управління віртуальними лабораторіями відіграє важливу роль в управлінні освітнім процесом. Він дозволяє адміністраторам і викладачам створювати і налаштовувати лабораторні завдання для студентів, а також керувати користувачами і відслідковувати їхню діяльність. До складу цього засобу входить кілька модулів:

Програмний модуль управління користувачами, який відповідає за автентифікацію, реєстрацію користувачів та управління їхніми правами доступу. Цей засіб дозволяє проводити контроль за виконанням завдань студентами, управляти ресурсами віртуальних лабораторій та обробляти дані, що надходять від користувачів.

Віртуальна лабораторія є інтегрованим середовищем, яке імітує реальні лабораторні установки або процеси і дозволяє студентам виконувати експерименти в дистанційному режимі. Віртуальна лабораторія містить цифрові двійники лабораторних пристроїв, які взаємодіють з програмним модулем збору даних і програмними інструментами. Через хмарний сервер студенти можуть виконувати різноманітні завдання, отримувати доступ до лабораторного обладнання та аналізувати результати в реальному часі. Програмний модуль управління віртуальною лабораторією контролює всі процеси, що відбуваються в лабораторії, і передає дані до хмарного сервера для подальшої обробки.

Висновки. Загалом, архітектура хмарного середовища для дистанційного навчання студентів забезпечує гнучкість і масштабованість навчального процесу. Використання хмарних серверів дозволяє зберігати і обробляти великі обсяги даних, надавати студентам віддалений доступ до віртуальних лабораторій і дозволяє викладачам ефективно керувати навчальним процесом. Це середовище не тільки дозволяє студентам взаємодіяти з цифровими моделями реальних лабораторних процесів, але і робить можливим навчання незалежно від місця перебування користувача, що значно підвищує доступність і якість освіти.

Список використаних джерел:

1. Node-RED та технологія промислового Інтернету речей: Навчальний посібник / І. Ш. Невлюдов, С. П. Новоселов, О. В. Сичова. – Харків: Видавництво Іванченка І. С., 2024. – 207 с.
2. Three Layer Architecture in the IoT. Specifications and Security Threats. KITRUM. URL: https://kitrum.com/blog/three-layer-architecture-in-the-internet-of-things/ (date of access: 31.12.2024).

ДОДАТОК Б

Вихідний код програми

Вихідний код клієнту:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace DistanceStudy
{
  public partial class Form1 : Form
  {
    public static string npgsqlConStr = "Server=178.250.222.333; Port=5432; User Id=student; Password=cloud_study; Database=study;";

    String SelectUser = "0";
    String old_SelectUser = "0";
    String SelectTask = "0";

    public Form1()
    {
      InitializeComponent();
    }

    private void Form1_Load(object sender, EventArgs e)
    {
      DataBase.CreateDatabase(npgsqlConStr);
```

```
            control_ViewUserOnline.SelectEvent += new
EventHandler(ViewUserOnline_SelectEvent);
            control_ViewOfline.SelectEvent += new
EventHandler(ViewOfline_SelectEvent);
            FillUserOnline();
            FillUserOfline();
        }

        private void ViewUserOnline_SelectEvent(object sender, EventArgs e)
        {
            if (control_ViewUserOnline == null) return;

            if (control_ViewUserOnline.selectElement.ID == "0")
            {
                SelectUser = "0";
                old_SelectUser = "0";
                return;
            }
            else
            {
                SelectUser = control_ViewUserOnline.selectElement.ID;
                FillTaskOnLine();
                FillAnswerOnLine();
                label7.Text = GetStudent_byID(SelectUser);
                if (SelectUser != old_SelectUser)
                {
                    old_SelectUser = SelectUser;
                }
            }
        }
```

```
private void FillTaskOnLine()
{
    SelectTask = "0";
    textBox1.Text = "";
    string sql = String.Format("Select TL.* from Task_list as TL where TL.ID_user = '{0}';", SelectUser);
    List<System.Object> rows_user = (List<System.Object>)DataBase.LoadList("DistanceStudy.Task_listMember", sql);
    foreach (System.Object ob in rows_user)
    {
        string ID_Task = (ob as Task_listMember).ID_task.ToString();
        string Description = (ob as Task_listMember).Description.ToString();
        textBox1.Text = Description;
        SelectTask = ID_Task;
    }
}

private void FillAnswerOnLine()
{
    textBox2.Text = "";
    string sql = String.Format("Select A.* from Answer as A where A.ID_user = '{0}';", SelectUser);
    List<System.Object> rows_user = (List<System.Object>)DataBase.LoadList("DistanceStudy.AnswerMember", sql);
    foreach (System.Object ob in rows_user)
    {
        string ID_Task = (ob as AnswerMember).ID_task.ToString();
```

```
                    string Description = (ob as AnswerMember).Description.ToString();
                    textBox2.Text = Description;
                }
            }

            private void FillTaskOfLine()
            {
                SelectTask = "0";
                textBox1.Text = "";
                string sql = String.Format("Select TL.* from Task_list as TL where TL.ID_user = '{0}';", SelectUser);
                List<System.Object> rows_user = (List<System.Object>)DataBase.LoadList("DistanceStudy.Task_listMember", sql);
                foreach (System.Object ob in rows_user)
                {
                    string ID_Task = (ob as Task_listMember).ID_task.ToString();
                    string Description = (ob as Task_listMember).Description.ToString();
                    textBox1.Text = Description;
                    SelectTask = ID_Task;
                }
            }

            private void FillAnswerOfLine()
            {
                textBox2.Text = "";
                string sql = String.Format("Select A.* from Answer as A where A.ID_user = '{0}';", SelectUser);
```

```
                List<System.Object> rows_user = (List<System.Object>)DataBase.LoadList("DistanceStudy.AnswerMember", sql);
                foreach (System.Object ob in rows_user)
                {
                    string ID_Task = (ob as AnswerMember).ID_task.ToString();
                    string Description = (ob as AnswerMember).Description.ToString();
                    textBox2.Text = Description;
                }
            }

            private void ViewOfline_SelectEvent(object sender, EventArgs e)
            {
                if (control_ViewOfline == null) return;

                if (control_ViewOfline.selectElement.ID == "0")
                {
                    SelectUser = "0";
                    old_SelectUser = "0";
                    return;
                }
                else
                {
                    SelectUser = control_ViewOfline.selectElement.ID;
                    FillTaskOfLine();
                    FillAnswerOfLine();
                    label7.Text = GetStudent_byID(SelectUser);
                    if (SelectUser != old_SelectUser)
                    {
                        old_SelectUser = SelectUser;
```

```
            }
        }
    }

    private void FillUserOnline()
    {
        Control_ViewTable cv = control_ViewUserOnline;
        cv.RowCount = 1;
        string sql = String.Format("Select U.*, (U.user_sname || ' ' " +
            "|| U.user_name || ' ' || U.user_fname) " +
            "as PIB from tUser as U where online = 'true';");

        cv.ID_Field_Name = "ID_User";
        cv.tableStructure.Clear();
        cv.tableStructure.Add(new
Control_ViewTable.TableStructure("PIB",
            "string", "PIB", 150, HorizontalAlignment.Left, 0, true));
        cv.createStructure();

        cv.sql_d = sql;
        cv.FillDataList();
    }

    private void FillUserOfline()
    {
        Control_ViewTable cv = control_ViewOfline;
        cv.RowCount = 1;
        string sql = String.Format("Select U.*, (U.user_sname || ' ' ||
U.user_name || ' ' || U.user_fname) as PIB from tUser as U where online = 'false';");
```

```
            cv.ID_Field_Name = "ID_User";
            cv.tableStructure.Clear();
            cv.tableStructure.Add(new
Control_ViewTable.TableStructure("PIB",        "string",        "PIB",        150,
HorizontalAlignment.Left, 0, true));
            cv.createStructure();
            cv.sql_d = sql;
            cv.FillDataList();
        }

        private string GetStudent_byID(string ID_User)
        {
            string sql = String.Format("Select (U.user_sname || ' ' || U.user_name
|| ' ' || U.user_fname) as PIB from tUser as U where U.ID_user = '{0}';", ID_User);
            string PIB = DataBase.Find_Value(sql, "PIB");
            return PIB;
        }

        private void button10_Click(object sender, EventArgs e)
        {
            if (SelectTask == "0")
            {
                String sql = String.Format("Insert into Task_list " +
                    "(ID_user, Status, Mark, Description) values ('{0}', " +
                    "'uncomplete', '0', '{1}');", SelectUser, textBox1.Text);
                DataBase.Exec_SQL(sql);
            }
            else
            {
                String sql = String.Format("Update Task_list " +
```

```
                "set Description = '{0}' where ID_Task = '{1}';",
                textBox1.Text, SelectTask);
            DataBase.Exec_SQL(sql);
        }
    }
  }
}
```

ДОДАТОК В

Демонстраційний матеріал

№ Доку-мента	Позначення	Найменування	Додаткові відомості
		Текстові документи	
1	ГЮИК. 503300.022 ПЗ	*Пояснювальна записка*	*А4, 79с*
		Інші документи	
2		*Апробація результатів наукових досліджень*	*А4, 5с*
3		*Вихідний код програми*	*А4, 8с*
4		*Демонстраційний матеріал*	*А4, 12с*

Змі	Арк	№ докум.	Підпис	Дата	ГЮИК. 503300.022 ВД			
Розроб		Ткачов М.Р.			*Розроблення програми на основі хмарного середовища для дистанційного навчання студентів*	Літера	Аркуш	Аркушів
Перевірив		Сичова О.В.				Н	1	1
						ХНУРЕ кафедра КІТАР		
Н. Контр.		Стародубцев М.Г.						
Затвер.		Невлюдов І.Ш.						