

Sau khi thực hành xong bài này, học viên có khả năng :

- Xây dựng được ứng dụng WinForm thực hiện các chức năng thêm , xóa và tìm kiếm dữ liệu thông qua Web API

Xây dựng ứng dụng Windows Form thực hiện chức năng quản lý thông tin các mặt hàng thông qua Web API (sử dụng List để lưu dữ liệu)

ProductID	ProductName	UnitPrice	Quantity
2	Coca	30	20
1	caphe	12	13

Bước 1 . Phát triển ứng dụng WinForm

1.1 Mở ứng dụng Visual Studio 2015 (**quyền Administrator**), chọn File | New Project | Other Project Type | Visual Studio Solutions | Blank Solution và đặt tên là "MySolution".

1.2 Từ menu File | Add New Project | Visual C# | WinForm , trên hộp thoại "Add New Project " điền các thông tin trong hộp thoại như sau : Name : **MaintainProducts**; **.NET Framework 4.6** , nhấp OK để thêm vào Solution và đổi tên **Form1.cs** thành **frmMain.cs** và các control được thiết lập các thuộc tính như sau :

Loại Control	Tên thuộc tính/ Tên Sự Kiện	Giá trị thiết lập
--------------	-----------------------------	-------------------

Form	Name	frmMain
	Text	Maintain Products
Label	Text	Product ID
Label	Text	Product Name
Label	Text	UnitPrice
Label	Text	Quantity
TextBox	Name	txtProductID
TextBox	Name	txtProductName
TextBox	Name	txtUnitPrice
TextBox	Name	txtQuantity
DataGridView	Name	dgvProducts
Button	Name	btnAddNew
	Text	Add New
	Sự kiện : Click	btnAddNew_Click
Button	Name	btnRemove
	Text	Remove
	Sự kiện : Click	btnRemove_Click
Button	Name	btnFind
	Text	Find
	Sự kiện : Click	btnFind_Click
Button	Name	btnClear
	Text	Clear
	Sự kiện : Click	btnClear_Click
Button	Name	btnLoad
	Text	Load
	Sự kiện : Click	btnLoad_Click

Bước 2 . Thêm vào ứng dụng Web API để thực hiện các thao tác thêm, xóa, tìm mặt hàng

2.1 Từ menu File | Add New Project | Visual C# | Web | ASP.NET Web Application , trên hộp thoại "New Project " điền các thông tin trong hộp thoại như sau : Name : **RemoteService**; **.NET Framework 4.6** , nhấn OK để qua hộp thoại "New ASP.NET Project", chọn ASP.NET 5 Templates | Web API và bỏ chọn **Host in the cloud** , nhấp OK để thêm project .

2.2 Nhấp phải chuột trên project RemoteService , chọn Add | New Folder , đặt tên là **Models**, nhấp phải trên Models chọn Add | Class , chọn ServerSide | Class, đặt tên cho lớp là Product.cs , viết code như sau :

```
using System.Collections;
namespace RemoteService.Models{
    //Khai bao lop Product
    public class Product{
        public int ProductID { get; set; }
        public string ProductName { get; set; }
        public float UnitPrice { get; set; }
        public int Quantity { get; set; }
    }//end class
    //Khai bao lop thuc hien cac thao tac : them, xoa, tim
    public static class ProductData {
        public static List<Product> Products = new List<Product>() {
            new Product { ProductID=1, ProductName="Caphe", Quantity=10, UnitPrice=20 },
            new Product { ProductID=2, ProductName="Coca", Quantity=20, UnitPrice=30 }
        };

    }//end class
}//end namespace
```

2.2 Chọn thư mục Controllers , nhấp phải trên Models chọn Add | Class , chọn ServerSide | WebAPI Controller Class, đặt tên cho lớp là ProductController.cs , viết code như sau :

.....

```

using System.Net.Http;
using System.Net;
using RemoteService.Models;
namespace RemoteService.Controllers{
    [Route("api/[controller]")]
    public class ProductController : Controller {
        //Khai bao phuong thuc lay danh sach mat hang
        [HttpGet] // GET: api/product
        public IEnumerable<Product> Get(){
            return ProductData.Products;
        }
        //Khai bao phuong thuc lay mat hang theo id
        [HttpGet("{id}")] // GET api/product/5
        public IActionResult Get(int id){
            Product p = ProductData.Products.SingleOrDefault(x => x.ProductID == id);
            if (p==null){
                return HttpNotFound();
            }
            return Ok(p);
        }
        //Khai bao phuong thuc them mat hang
        [HttpPost()] // POST api/product
        public void Post([FromBody] Product p) {
            ProductData.Products.Add(p);
        }
        //Khai bao phuong thuc xoa mat hang
        [HttpDelete("{id}")] // DELETE api/product/5
        public void Delete(int id){
            Product p = ProductData.Products.SingleOrDefault(x => x.ProductID == id);
            ProductData.Products.Remove(p);
        }
    }
}

```

2.3 . Kiểm tra ứng dụng Web API

-Nhấp phải chuột trên project "**RemoteService**", chọn "**Set as Startup Project**", nhấn Ctrl+F5 để chạy ứng dụng. Trên trình duyệt web thêm vào cuối URL chuỗi "**/api/product**": http://localhost:xxxx/api/product , kết quả như hình sau.



Bước 3: Viết mã cho các sự kiện của các nút trên ứng dụng WinForm

```
using System.Collections;
using System.Net.Http;
namespace MaintainProducts{
    public partial class frmMain : Form {
        public frmMain(){
            InitializeComponent();
        }
        HttpClient client = new HttpClient();
        string baseUri = "http://localhost:60634/api/product/";
        private void btnLoad_Click(object sender, EventArgs e){
            try{
                HttpResponseMessage resp = client.GetAsync(baseUri).Result;
                //kiem tra co loi hay khong ?
                resp.EnsureSuccessStatusCode();
                List<Product> productList = resp.Content.ReadAsAsync<List<Product>>().Result;
                //hien thi len GridView
                dgvProducts.DataSource = null;
                dgvProducts.DataSource = productList;
            }
            catch (Exception ex){
                MessageBox.Show(ex.Message);
            }
        }

        private void btnAddNew_Click(object sender, EventArgs e){
            try {
                int ProID = int.Parse(txtProductID.Text);
                string ProName = txtProductName.Text;
                float Price = float.Parse(txtUnitPrice.Text);
                int Quantity = int.Parse(txtQuantity.Text);
                //Tao doi tuong Product
                Product p = new Product {ProductID = ProID, ProductName = ProName,
                    UnitPrice = Price, Quantity = Quantity };
                HttpResponseMessage resp = client.PostAsJsonAsync(baseUri, p).Result;
                resp.EnsureSuccessStatusCode();
                MessageBox.Show("Product is saved.");
            }
            catch(Exception ex) {
                MessageBox.Show(ex.Message);
            }
        }

        private void btnRemove_Click(object sender, EventArgs e) {
            try{
```

```

        int ProID = int.Parse(txtProductID.Text);
        HttpResponseMessage resp = client.DeleteAsync(baseUri+ProID).Result;
        resp.EnsureSuccessStatusCode();
        MessageBox.Show("Delete successful.");
    }
    catch(Exception ex){
        MessageBox.Show(ex.Message);
    }
}

private void btnFind_Click(object sender, EventArgs e) {
    try{
        int ProID = int.Parse(txtProductID.Text);
        HttpResponseMessage resp = client.GetAsync(baseUri+ProID).Result;
        //kiem tra truy cap thanh cong
        resp.EnsureSuccessStatusCode();
        Product p = resp.Content.ReadAsAsync<Product>().Result;
        if (p == null){
            MessageBox.Show("Product not found .");
        }
        else{
            txtProductName.Text = p.ProductName;
            txtQuantity.Text = p.Quantity.ToString();
            txtUnitPrice.Text = p.UnitPrice.ToString();
        }
    }
    catch(Exception ex){
        MessageBox.Show(ex.Message);
    }
}

private void btnClear_Click(object sender, EventArgs e){
    txtProductID.Text = string.Empty;
    txtProductName.Text = string.Empty;
    txtQuantity.Text = string.Empty;
    txtUnitPrice.Text = string.Empty;
    dgvProducts.DataSource = null;
}

} //end class
} //end namespace

```

Bước 4: Chạy ứng dụng WinForm và kiểm tra các chức năng . Nhấp phải chuột trên project "**MaintainProduct**", chọn "**Set as Startup Project**", nhấn Ctrl+F5 để chạy ứng dụng.