



Epicor ERP软件二次开发：EpicorERP二次开发之WebServices与RESTAPI应用

Epicor ERP软件二次开发：WebServices与RESTAPI应用

1. 简介与预备知识

1.1 Epicor ERP软件概述

Epicor ERP是一款功能强大的企业资源规划软件，旨在帮助中大型企业优化其业务流程，提高运营效率。它提供了涵盖财务、供应链、制造、销售和服务等领域的全面解决方案。在Epicor ERP中，二次开发是指在软件标准功能的基础上，根据企业特定需求进行的定制化开发，以增强软件的功能性和适应性。

1.2 二次开发的重要性

二次开发对于Epicor ERP用户来说至关重要，因为它允许企业根据自身业务流程的特殊需求，定制和扩展ERP系统的功能。这不仅能够提高系统的使用效率，还能确保ERP系统与企业的业务策略保持一致，从而提升整体的业务管理能力。

1.3 WebServices与RESTAPI基础

WebServices和RESTAPI是Epicor ERP二次开发中常用的两种技术，它们提供了与ERP系统进行交互的途径。

WebServices

WebServices是一种允许不同应用程序之间进行通信的技术，通常基于SOAP协议。在Epicor ERP中，WebServices可以用于访问ERP系统的业务逻辑和数据，实现外部系统与ERP的集成。

示例代码：

```
// C#示例：调用Epicor ERP的WebServices
using System;
using System.Net;
using System.Xml;
using System.IO;

public class EpicorWebServiceExample
{
```

```

public static void Main()
{
    string url = "http://your-epicor-server/YourService.asmx";
    string soapAction = "http://tempuri.org/IService/YourMethod";
    string xmlRequest = "<soapenv:Envelope
xmlns:soapenv=\"http://schemas.xmlsoap.org/soap/envelope/\"
xmlns:tem=\"http://tempuri.org/\">\" +
        "<soapenv:Header/>\" +
        "<soapenv:Body>\" +
        "<tem:YourMethod>\" +

    "<tem:parameter1>value1</tem:parameter1>\" +

    "<tem:parameter2>value2</tem:parameter2>\" +
        "</tem:YourMethod>\" +
        "</soapenv:Body>\" +
        "</soapenv:Envelope>";

    HttpWebRequest request =
    (HttpWebRequest)WebRequest.Create(url);
    request.Method = "POST";
    request.ContentType = "text/xml; charset=utf-8";
    request.Headers.Add("SOAPAction", soapAction);

    using (StreamWriter writer = new
    StreamWriter(request.GetRequestStream()))
    {
        writer.Write(xmlRequest);
    }

    HttpWebResponse response =
    (HttpWebResponse)request.GetResponse();
    using (StreamReader reader = new
    StreamReader(response.GetResponseStream()))
    {
        string xmlResponse = reader.ReadToEnd();
        Console.WriteLine(xmlResponse);
    }
}
}

```

此示例展示了如何使用C#代码调用Epicor ERP的WebServices。通过构造SOAP请求并发送到指定的URL，可以获取ERP系统的响应数据。

RESTAPI

RESTAPI (Representational State Transfer Application Programming Interface) 是一种基于HTTP协议的Web服务，它使用标准的HTTP方法（如GET、POST、PUT、DELETE）来操作资源。Epicor ERP的RESTAPI提供了更现代、更灵活的接口，允许开发者以JSON或XML格式与ERP系统进行数据交换。

示例代码：

```
// C#示例：通过RESTAPI获取Epicor ERP的数据
using System;
using System.Net.Http;
using System.Threading.Tasks;

public class EpicorRestApiExample
{
    public static async Task Main()
    {
        string url = "http://your-epicor-server/api/YourResource";
        using (HttpClient client = new HttpClient())
        {
            client.DefaultRequestHeaders.Add("Authorization", "Bearer your-token");
            HttpResponseMessage response = await client.GetAsync(url);
            if (response.IsSuccessStatusCode)
            {
                string jsonResponse = await response.Content.ReadAsStringAsync();
                Console.WriteLine(jsonResponse);
            }
            else
            {
                Console.WriteLine($"Error: {response.StatusCode}");
            }
        }
    }
}
```

在上述示例中，我们使用C#的HttpClient类来发送GET请求到Epicor ERP的RESTAPI。通过添加授权头，可以确保请求被ERP系统正确处理。如果请求成功，将输出JSON格式的响应数据。

2. 结论

通过掌握Epicor ERP的WebServices和RESTAPI，企业可以有效地进行二次开发，定制和扩展ERP

系统的功能，以满足特定的业务需求。这不仅增强了ERP系统的实用性，还促进了企业业务流程的优化和自动化。

Epicor ERP软件二次开发：WebServices应用开发

3. 理解Epicor WebServices架构

在Epicor ERP系统中，WebServices架构是实现系统与外部应用或自定义模块之间通信的关键。Epicor提供了多种WebServices，包括SOAP和REST，以适应不同的开发需求和环境。

3.1 SOAP WebServices

SOAP（Simple Object Access Protocol）是一种基于XML的协议，用于在Web上交换结构化信息。在Epicor中，SOAP WebServices主要用于复杂的数据交互和业务逻辑处理。例如，通过SOAP WebServices，你可以调用Epicor中的业务方法，进行数据的读取、更新、创建和删除操作。

3.2 REST API

REST（Representational State Transfer）是一种网络应用程序的设计风格和开发方式，基于HTTP协议，可以使用XML、JSON等格式传输数据。Epicor的REST API提供了更现代、更灵活的接口，适用于需要快速响应和轻量级数据交换的场景。REST API通常用于移动应用开发、云服务集成和实时数据访问。

4. 使用Epicor WebServices进行数据访问

4.1 示例：使用SOAP WebServices查询客户信息

假设我们需要从Epicor ERP中查询客户信息，可以使用SOAP WebServices来实现。下面是一个使用C#调用Epicor SOAP WebServices的示例代码：

```
using System;
using System.Net;
using System.Xml;
using Epicor.E10.Customer;

namespace EpicorWebServiceExample
{
    class Program
    {
        static void Main(string[] args)
        {
```

```

        // 创建WebServices客户端
        CustomerServiceClient client = new
CustomerServiceClient();
        client.Url = "http://your-epicor-server:8080/
ERPWebServices/Customer/CustomerService";
        client.Credentials = new NetworkCredential("your-
username", "your-password");

        // 设置SOAP请求头
        client.ClientCredentials.Windows.ClientCredential = new
NetworkCredential("your-username", "your-password");

        // 查询客户信息
        CustomerQueryRequest request = new
CustomerQueryRequest();
        request.CustomerQuery = new CustomerQuery();
        request.CustomerQuery.CustomerID = "C00001";
        CustomerQueryResponse response =
client.CustomerQuery(request);

        // 处理响应
        Customer customer = response.CustomerQueryResponse;
        Console.WriteLine("Customer Name: " +
customer.CustomerName);
        Console.WriteLine("Customer Address: " +
customer.CustomerAddress);
    }
}
}

```

4.2 示例：使用REST API更新库存

对于REST API的使用，假设我们需要更新Epicor ERP中的库存信息，下面是一个使用Python调用Epicor REST API的示例代码：

```

import requests
import json

# 设置REST API的URL和认证信息url = "http://your-epicor-server:8080/
ERPWebServices/Inventory/InventoryItem"
headers = {
    'Content-Type': 'application/json',
    'Authorization': 'Basic ' + base64.b64encode(f'your-
username:your-password'.encode()).decode()

```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/837006141031006143>