

Лабораторная работа #1

3В Реализация платежной системы

Кустарева Екатерина, Тулунов Матвей ФИ-12 мн

Для реализации платежной системы был выбран C для удобной работы со встроенными библиотеками WebAPI.

Есть 2 идеи реализации:

- 1) Идея описания классов, между которыми осуществляются транзакции.
- 2) Идея в реализации API с эндпоинтами для связи frontEnd и backEnd. Для это идеи нужно продумать параметры и их типы

Например, формат данных - json, протоколы обмена - https.

	A	B	C	D	E	F	G	H	I
1		user			flag		flagDescriptor		
2	Login	string		Descriptor	flagDescriptor		Identifier	int64	
3	Password	string		Dependencies	flag[]		Details	string (optional)	
4	Sign	string		Conflicts	flag[]				
5	Flags	flag[]							
6									
7		address			wallet		currency		
8	Identifier	string		Identifier	string		Identifier	int64	
9	Currency	currency		Currency	currency		Symbol	string	
10	Amount	decimal (optional)		Amount	decimal				
11				Owner	user				
12				Address	address				
13									
14		transaction			receipt		status		
15	Identifier	string		Identifier	string				
16	Address	address		Address	address				
17	Sender	wallet							
18	Currency	currency (option)							
19	Amount	decimal (option)							
20	Status	status							
21									
22									
23									
24									
25									
26									

```

using System;
using System.Threading.Tasks;

namespace ConsoleApp1
{
    public record User(string Login, string Password, string Sign, Enum[] Flags);
    public record Currency(long Id, string Symbol);
    public record Address(string Id, Currency Currency, double? Amount);
    public record Wallet(string Id, Currency Currency, double Amount, User Owner, Address Address);
    public record Transaction(string Id, Address Address, Wallet Sender, Currency Currency);
    public record Receipt(string Id, Address Address);
    public enum TransactionStatus
    {
        Da,
        Net
    }

    class Program
    {
        static async Task Main(string[] args)
        {
            Console.WriteLine("Initializing bank infrastructure, please wait...");
            while (true)
            {
            }
        }
    }
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;

namespace ServiceExample
{
    public class Product
    {
        public string Id { get; set; }
        public string Name { get; set; }
    }

    [ServiceContract(Namespace = "")]

    public interface IProductService
    {
        [WebGet(UriTemplate = "Product/{id}", ResponseFormat = WebMessageFormat.Json)]
        [OperationContract]
        Product Product(string id);
    }

    public class ProductService : IProductService
    {
        public Product Product(string id)
        {
            return new Product { Id = id, Name = "A Sample Product" };
        }
    }

    class Program
    {
        private static ServiceHost servHost;
        static void Main(string[] args)
        {
            StartService();
            Console.WriteLine("\n\nPress any key to exit...");
            Console.ReadKey();
        }

        public static void StartService()
        {
            servHost = new ServiceHost(typeof(ProductService));
            servHost.Open();
        }
    }
}

```

```
~Program()  
{  
    if (servHost != null)  
    {  
        servHost.Close();  
    }  
}  
}
```