МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА «ИИТ»

Лабораторная работа №6

По дисциплине «Проектирование интернет систем»

Выполнил: студент ФЭИС

группы ПО-2

Шелест П.Ю.

Проверила: Слинко Е.В.

Брест, 2020

**Цель:** Познакомиться с практической реализацией принципа инверсии зависимостей, принципа единственной ответственности и механизмом внедрения зависимостей

**Задание:**

Реализуйте сервисы на уровне приложения, реализующие с доменными моделями те же действия, что и сценарий транзакции из ЛР №1. Не забудьте о принципе единственной ответственности. Хотя бы одно действие должно реализовываться доменным сервисом.

**Вариант**: *продажа электрооборудования*

**Список сервисов**:

-ClientUseCase – сервис клиентов

-ProductUseCase – сервис электрооборудования

**Текст программы:**

class ClientUseCase

{

    private $repos;

    public function \_\_construct(ClientRepository $rep)

    {

        $this->repos=$rep;

    }

    public function authorization(string $login, string $pass)

    {

        $client = $this->repos->getClientByLP($login, $pass);

        if($client==null)

        {

            return 0;}

        else

        {

            $\_SESSION['user']=2;

                $\_SESSION['user\_id']=$client->getUserId();

                $client->authorize();

                $disp=new OrderDispatcher();

                $disp->dispatch($client->releaseOrders());

                return $client;

        }

    }

}

class ProductUseCase

{

    private $client\_rep;

    private $Product\_rep;

    public function \_\_construct(ClientRepository $clientR, ProductRepository $ProductR)

    {

        $this->client\_rep=$clientR;

        $this->Product\_rep=$ProductR;

    }

    public function addProduct(string $name, string $price,string $infrom, string $image)

    {

        $this->Product\_rep-> addProduct($name, $price, $inform, $image);

    }

    public function removeProduct(string $id\_Product)

    {

        $Product = $this->Product\_rep->getProductById($id);

        if($Product)

        $this->Product\_rep->removeProduct($Product);

    }

    public function getProducts()

    {

        $Products=[];

        $Products = $this->Product\_rep->getProduct();

            return $Products;

    }

    public function getUserProducts()

    {

        $userProducts=[];

        $id = $\_SESSION['user\_id'] ;

        $user = new Client("", "","", "", "", $id);

        $userProducts = $Product\_rep->getClientProduct($user);

        return $userProducts;

    }

    public function ClientByProduct(Client $client, Product $Product)

    {

        $Product->ClientBy($client);

        $result = $this->serviceRepository->ClientByProduct($client, $Product);

        $disp=new OrderDispatcher();

        $disp->dispatch($Product->releaseOrders());

    }

}

**Выводы:**

В результате выполнения работы я ознакомился с практической реализацией принципа инверсии зависимостей, принципа единственной ответственности и механизмом внедрения зависимостей.