

Київський національний університет імені Тараса
Шевченка Факультет інформаційних технологій

Кафедра програмних систем і технологій

ЗВІТ

з лабораторної роботи № 5

Тема: “Автоматизація бізнес-процесів архітектурного

3Д-дизайну”

Дисципліна «Об’єктно-орієнтоване конструювання

програм»

Підготував:

студент гр. ІПЗ-23(1)

Шевченко Андрій Віталійович

Дата : 22.05.22

Перевірила:

Зубик Л.В

Київ – 2022

Тема: Структурні патерни проектування.

Мета роботи: ознайомитися з структурними паттернами проектування.

Умова: Розробити класи (згідно варіанту) з використанням структурного патерну проектування. Реалізувати програму на C# з простим графічним інтерфейсом. У висновках обґрунтувати вибір патерну і пояснити, яким чином він дав змогу спростити виконання вашого варіанту завдання.

Завдання: Програма обліку персоналу в організації. Організація має директора, заступника директора і складається з відділів (підрозділів). Кожен відділ має керівника, заступників, спеціалістів, службовців і робітників. Передбачити ієрархію відділів та ієрархію працівників у відділі. Методи отримання інформації про ієрархію організації і про будь-якого її відділу. Методи прийняття на роботу та звільнення працівників, переведення з одного відділу до іншого, зміни посади. Передбачити можливість створення або ліквідації відділу.

Хід роботи: Для реалізації даної лабораторної роботи з запропонованих паттернів я обрав паттерн “Фасад”.

Реалізація:

Для реалізації поставленого завдання згідно варіанту ми будемо використовувати Orm Dapper та паттерн Facade.

Для початку створимо моделі, які будуть відповідати структурі бд.

```
public class Organization
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string DirectorPIB { get; set; }
    public string DeputyDirectorPIB { get; set; }
    public int DepartmentId { get; set; }
}
```

Ссылка: 8

```
public class Department
{
    Ссылка: 0
    public int Id { get; set; }
    Ссылка: 0
    public string Name { get; set; }
    Ссылка: 0
    public string SupervisorPIB { get; set; }
    Ссылка: 0
    public string DeputySupervisorPIB { get; set; }
    Ссылка: 0
    public string SpecialistPIB { get; set; }
    Ссылка: 0
    public string EmployeePIB { get; set; }
}
```

Далі потрібно налаштувати зв'язок з нашою базою даних за допомогою Dapper, та створити сервіси які будуть комунікувати з бд та відображенням застосунку.

```
public class DatabaseConnection
{
    public static SqlConnection CreateConnection()
    {
        return new SqlConnection("Data Source=DESKTOP-ELR5B8P;Initial Catalog=Organization;Integrated Security=True;");
    }
}
```

Ссылка: 1

```
public IEnumerable<Organization> GetAllOrganizations()
{
    using (var connection = DatabaseConnection.CreateConnection())
    {
        return connection.Query<Organization>("SELECT * FROM Organization");
    }
}
```

Ссылка: 1

```
public void AddOrganization(Organization organization)
{
    using (var connection = DatabaseConnection.CreateConnection())
    {
        connection.Execute("INSERT INTO Organization (Name, DirectorPIB, DeputyDirectorPIB, DepartmentId) VALUES(@Name, @DirectorPIB, @DeputyD");
    }
}
```

Ссылка: 0

```
public IEnumerable<Organization> GetOrganizationByName(string name)
{
    using (var connection = DatabaseConnection.CreateConnection())
    {
        var sql = $"SELECT * FROM Organization Where Name = {name}";
        return connection.Query<Organization>(sql);
    }
}
```

```

Ссылка: 1
public IEnumerable<Department> GetAllDepartments()
{
    using (var connection = DatabaseConnection.CreateConnection())
    {
        return connection.Query<Department>("SELECT * FROM Department");
    }
}

Ссылка: 1
public void AddDepartment(Department department)
{
    using (var connection = DatabaseConnection.CreateConnection())
    {
        connection.Execute("INSERT INTO Department (Name, SupervisorPIB, DeputySupervisorPIB, SpecialistPIB, EmployeePIB) VALUES(@Name, @Supe");
    }
}

Ссылка: 0
public IEnumerable<Department> GetByDepartmentByName(string name)
{
    using (var connection = DatabaseConnection.CreateConnection())
    {
        var sql = $"SELECT * FROM Department WHERE Name = {name}";
        return connection.Query<Department>(sql);
    }
}

```

Після цього ми можемо реалізувати наш паттерн, він реалізується дуже простим чином, нам потрібно створити клас, який буде відповідати за створення екземплярів сервісів, та викликати потрібні нам методи.

```

Ссылка: 0
public class Facade
{
    private readonly OrganizationService _organizationService;
    private readonly DepartmentService _departmentService;

    Ссылка: 0
    public Facade()
    {
        _organizationService = new OrganizationService();
        _departmentService = new DepartmentService();
    }
}

```

```
Ссылка: 1
public IEnumerable<Organization> GetAllOrganizations()
{
    return _organizationService.GetAllOrganizations();
}

Ссылка: 1
public IEnumerable<Department> GetAllDepartments()
{
    return _departmentService.GetAllDepartments();
}

Ссылка: 1
public void AddDepartment(Department department)
{
    _departmentService.AddDepartment(department);
}

Ссылка: 1
public void AddOrganization(Organization organization)
{
    _organizationService.AddOrganization(organization);
}
```

```
Ссылка: 0
public IEnumerable<Organization> GetOrganizationByName(string name)
{
    return _organizationService.GetOrganizationByName(name);
}

Ссылка: 0
public IEnumerable<Department> GetDepartmentByName(string name)
{
    return _departmentService.GetDepartmentByName(name);
}
```

Одним з останніх кроків буде створення контролерів, які будуть проміжним слоєм комунікації між бд та користувацьким інтерфейсом.

Останній крок це створення користувацького інтерфейсу за допомогою Razor, я продемонструє на прикладі коду для організації.

```

1 <body>
2 <div>
3 <table>
4 <tr>
5 <td>GetAllOrganizations</td>
6 <td>AddOrganization</td>
7 </tr>
8 <tr>
9 <td>
10 <form action="@Url.Action("GetAllOrganizations", "Organization")" method="get">
11 <input type="hidden"/>
12 <button type="submit">GetAllOrganizations</button>
13 </form>
14 </td>
15 <td>
16 <form action="@Url.Action("AddOrganization", "Organization")" method="get">
17 <input type="hidden" name="Id"/>
18 <button type="submit">AddOrganization</button>
19 </form>
20 </td>
21 <td>
22 <form action="@Url.Action("GetById", "Organization")" method="get">
23 <input type="hidden" name="Id"/>
24 <button type="submit">GetById</button>
25 </form>
26 </td>
27 </tr>
28 </table>
29 </div>

```

```

1 <body>
2 <div>
3 <table>
4 <tr>
5 <td>Id</td>
6 <td>Name</td>
7 <td>DirectorPIB</td>
8 <td>DeputyDirectorPIB</td>
9 <td>DepartmentId</td>
10 </tr>
11 @foreach (var value in Model)
12 {
13 <tr>
14 <td>@value.Id</td>
15 <td>@value.Name</td>
16 <td>@value.DirectorPIB</td>
17 <td>@value.DeputyDirectorPIB</td>
18 <td>@value.DepartmentId</td>
19 </tr>
20 }
21 </table>
22 </div>
23 </body>

```

```

1  <form method="post" action="">
2      <input type="hidden" value="@ViewBag.Id" name="Id"/>
3      <table>
4          <tr>
5              <td><p>Enter Name:</p></td>
6              <td><input type="text" name="Name"/></td>
7          </tr>
8          <tr>
9              <td><p>Enter DirectorPIB:</p></td>
10             <td><input type="text" name="DirectorPIB"/></td>
11          </tr>
12          <tr>
13              <td><p>Enter DeputyDirectorPIB:</p></td>
14              <td><input type="text" name="DeputyDirectorPIB"/></td>
15          </tr>
16          <tr>
17              <td><p>Enter DepartmentId:</p></td>
18              <td><input type="text" name="DepartmentId"/></td>
19          </tr>
20          <tr>
21              <td><input type="submit" value="Add Department"/></td>
22          </tr>
23      </table>
24  </form>

```

```

1  <form method="get" action="">
2      <input type="hidden" value="@ViewBag.Id" name="Id"/>
3      <table>
4          <tr>
5              <td><p>Enter Name:</p></td>
6              <td><input type="text" name="Name"/></td>
7          </tr>
8          <tr>
9              <td><input type="submit" value="GetByName"/></td>
10          </tr>
11      </table>
12  </form>

```

Результат:

Lab5 Organization Department

Lab5 Organization Department

GetAllOrganizations AddOrganization

GetAllOrganizations AddOrganization GetByName

Lab5 Organization Department

Id >Name	DirectorPIB	DeputyDirectorPIB	DepartmentId
1	AndreyOrganization	AndreyOrganization	AndreyOrganization 1

Висновки: Під час виконання цієї лабораторної роботи я ознайомився з структурними патернами проектування, які входять в Gof, та допомагають створювати більш гнучкі та великі системи. Для реалізації завдання я обрав патерн “Фасад”, тому що вважаю він більш доцільно підходить під це завдання.