

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Кафедра інженерії програмного забезпечення

**КУРСОВИЙ ПРОЕКТ**  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

з дисципліни: «Бази даних»  
на тему:

**«База даних інформаційної системи управління  
індивідуальними замовленнями підприємства (на прикладі  
підприємства з виготовлення меблів)»**

студента II курсу групи ППЗ-19-3  
спеціальності 121 «Інженерія  
програмного забезпечення»  
Линника Максима Руслановича

(прізвище, ім'я та по-батькові)

Керівник: Чижмотря О.В.

Дата захисту: "11" червня 2021 р.

Національна шкала \_\_\_\_\_

Кількість балів: \_\_\_\_\_

Оцінка: ECTS \_\_\_\_\_

Члени комісії

\_\_\_\_\_ (підпис) \_\_\_\_\_ (прізвище та ініціали)

\_\_\_\_\_ (підпис) \_\_\_\_\_ (прізвище та ініціали)

\_\_\_\_\_ (підпис) \_\_\_\_\_ (прізвище та ініціали)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»  
Факультет інформаційно-комп'ютерних технологій  
Кафедра інженерії програмного забезпечення  
Освітній рівень: бакалавр  
Спеціальність 121 «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»  
Зав. кафедри

“ \_\_\_\_ ” \_\_\_\_\_ 2021р.

ЗАВДАННЯ  
НА КУРСОВИЙ ПРОЕКТ СТУДЕНТУ  
Линнику Максиму Руслановичу

1. Тема роботи: База даних інформаційної системи управління індивідуальними замовленнями підприємства (на прикладі підприємства з виготовлення меблів),  
керівник роботи : ст. викладач каф. ІІЗ Чижмотря Олексій Володимирович.
2. Строк подання студентом: “\_” \_\_\_\_\_ 2021\_\_р.
3. Вхідні дані до роботи: Розробити базу даних інформаційної системи управління індивідуальними замовленнями підприємства(на прикладі підприємства виготовлення меблів)
4. Зміст розрахунково-пояснювальної записки(перелік питань. Які підлягають розробці)
  - аналіз методів та засобів вирішення задачі \_\_\_\_\_
  - проектування та розробка програмного додатку \_\_\_\_\_
  - опис роботи з програмним додатком та його тестування \_\_\_\_\_
5. Перелік графічного матеріалу(з точним зазначенням обов'язкових креслень)
  - 1) Презентація до курсового проекту
  - 2) Посилання на GitHub: [https://github.com/MaksHW/BD\\_K](https://github.com/MaksHW/BD_K)
6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посади консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання “06” березня 2021 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів курсового проекту	Строк виконання етапів проекту	Примітки
1	Постановка задачі	06.03	виконано
2	Пошук, огляд та аналіз аналогічних розробок	07.03	виконано
3	Формулювання технічного завдання	02.04	виконано
4	Опрацювання літературних джерел	20.04	виконано
5	Проектування структури	30.04	виконано
6	Написання програмного коду	15.05	виконано
7	Налагодження	27.05	виконано
8	Написання пояснювальної записки	04.06	виконано
9	Захист	11.06	виконано

Студент \_\_\_\_\_  
(підпис)

Линник М. Р.  
(прізвище та ініціали)

Керівник проекту \_\_\_\_\_  
(підпис)

Чижмотря О.В.  
(прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка до курсового проекту на тему «База даних інформаційної системи управління індивідуальними замовленнями підприємства (на прикладі підприємства з виготовлення меблів)» складається з переліку умовних скорочень, вступу, чотирьох розділів, висновків, списку використаної літератури та додатку.

Текстова частина викладена на 30 сторінках друкованого тексту.

Пояснювальна записка має 7 сторінок додатків. Список використаних джерел містить 4 найменувань і займає 1 сторінку. В роботі наведено 12 рисунків та 8 таблиць. Загальний обсяг роботи – 37 сторінки.

У першому розділі було проаналізовано потреби та визначено засоби реалізації.

У другому розділі проведено проектування бази даних.

У третьому розділі було розроблено сайт.

У четвертому розділі було розроблено систему захисту інформації в базі даних та налаштовано роботу сервера.

Висновок містить в собі результати виконаної роботи при створенні сайту на тему «База даних інформаційної системи управління індивідуальними замовленнями підприємства (на прикладі підприємства з виготовлення меблів)».

У додатку представлений лістинг розробленого програмного продукту.

Ключові слова: C#, MSSQL, WEB-САЙТ, СЕРВЕР, АВТОРИЗАЦІЯ, ДАНІ.

					«Житомирська політехніка».21.121.10.000 – ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка бази даних підприємства меблів		
Розроб.		Линник М.Р.					
Керівник		Чижмоторя О. В.					
Реценз.							
Н. Контр.							
Затверд.					ФІКТ, гр. ПЗ-19-3		
					Літ.	Арк.	Аркушів
						4	30

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ .....	7
ВСТУП .....	8
РОЗДІЛ 1. ТЕОРЕТИЧНИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ ПОТОКІВ ТА ОСОБЛИВОСТЕЙ СТВОРЕННЯ ВЕБ-САЙТУ .....	9
1.1 Аналіз інформаційних потреб та визначення предметної області .....	9
1.2 Обґрунтування вибору засобів реалізації .....	9
Висновки до першого розділу .....	11
РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ІНДИВІДУАЛЬНИМИ ЗАМОВЛЕННЯМИ ПІДПРИЄМСТВА (НА ПРИКЛАДІ ПІДПРИЄМСТВА З ВИГОТОВЛЕННЯ МЕБЛІВ).....	12
2.1 Аналіз інформаційних процесів.....	12
2.2 Проектування структури бази даних сайту інформаційної системи управління індивідуальними замовленнями підприємства(на прикладі підприємства з виготовлення меблів) .....	14
Висновки до другого розділу .....	17
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПІДСИСТЕМИ ОБРОБКИ ДАНИХ САЙТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ІНДИВІДУАЛЬНИМИ ЗАМОВЛЕННЯМИ ПІДПРИЄМСТВА (НА ПРИКЛАДІ ПІДПРИЄМСТВА З ВИГОТОВЛЕННЯ МЕБЛІВ) .....	18
3.1 Проектування інтерфейсу обробки даних .....	18
3.2 Реалізація операцій обробки даних на сайті інформаційної системи управління індивідуальними замовленнями підприємства(на прикладі підприємства з виготовлення меблів) .....	23
3.3 Організація звітності системи.....	26
Висновки до 3 розділу: .....	26

РОЗДІЛ 4. АДМІНІСТРУВАННЯ БАЗ ДАНИХ .....	27
4.1 Розробка заходів захисту інформації в БД .....	27
4.2 Налаштування параметрів роботи сервера .....	28
Висновки до 4 розділу: .....	28
ВИСНОВКИ.....	29
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	30
ДОДАТКИ.....	31
Додаток А.....	32

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД – База даних

СУБД – система управління базами даних

КП – курсовий проект

MVC – Модель–представлення–контролер — архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення.

ПЗ – Програмне забезпечення

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		7

## ВСТУП

### **Актуальність теми.**

В наш час майже все можна замовити в Інтернеті, це дешевше та доступніше, ніж ходити кудись, особливо з кур'єрською доставкою. Але не завжди можна замовити те, що тобі потрібно з індивідуальними корективами. А ті рідкісні можливості замовити авторську річ індивідуального створення зазвичай проходять по телефону або очно, без комфорту Інтернет замовлень. Особливо це відчутно з меблями, які потрібні всім без виключень. А от місце, де можна замовити унікальну річ не виходячи з квартири – надзвичайно спростить життя усім.

Тому, зараз інформаційні системи управління індивідуальними замовленнями потрібні.

### **Мета створення.**

Метою створення є розробка бази даних інформаційної системи управління індивідуальними замовленнями підприємства (на прикладі підприємства з виготовлення меблів).

**Об'єктом дослідження** є розробка бази даних для сайту в середовищі Microsoft SQL Server та використання мови C# для роботи з нею.

**Предметом дослідження** є вивчення можливостей Microsoft SQL Server для роботи з базою даних та Visual Studio для роботи з серверною частиною проекту.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		8



# РОЗДІЛ 1. ТЕОРЕТИЧНИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ ПОТОКІВ ТА ОСОБЛИВОСТЕЙ СТВОРЕННЯ ВЕБ-САЙТУ

## 1.1 Аналіз інформаційних потреб та визначення предметної області

База даних (БД) – це сукупність структурованих даних, які стосуються певної області. Основною задачею є зберігання інформації доступно й зручно, щоб була можливість без проблем отримати, модифікувати та видалити дані.

Будь-який сайт потребує наявності бази даних, якщо є потреба працювати з даними навіть малих розмірів, хоча б для комфорту.

На даний момент, існує достатньо сайтів з можливістю замовлень товарів з Інтернету. На таких сайтах є перелік товарів, обов'язкова реєстрація, контроль замовлень. Будь-який користувач може переглядати товари, але тільки зареєстрований може купувати.

На таких сайтах неможливо замовити товар з індивідуальними характеристиками. Але вони мають можливість комфортно, та не виходячи з приміщення контролювати своє замовлення.

Для авторських меблів або з можливістю кастомізувати її сайтів не так багато, та й ті не популярні, з поганим інтерфейсом та лише контактним телефоном для зв'язку. Щодо виробництва меблів – така проблема дуже гостро відчувається.

Тому, рішення створити власний сайт з можливістю модифікувати виріб та слідкувати за замовленою одиницею меблів є вірним.

## 1.2 Обґрунтування вибору засобів реалізації

СУБД (система управління базами даних) - набір взаємозалежних даних (база даних) і програми для доступу та взаємодії до них. Функції: створення, збереження, оновлення, видалення та пошук, контроль доступу.

Існує достатньо систем управління базами даних. Вони можуть по-різному працювати з різними об'єктами і надають користувачу різні функції і засоби.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

Найбільш важливий момент у виборі СУБД – чи є вона реляційної (SQL) чи нереляційної (NoSQL) структури. У обох варіантів є свої переваги, а також кілька ключових особливостей, які варто мати на увазі при виборі.

Таблиця 1

### Характеристики СУБД

Характеристика	MS SQL Server	MongoDb	MySQL
Модель даних	Реляційна	Документно-орієнтована	Реляційна
Підключення к Web	Добре	Добре	Добре
Мова SQL	Відмінно	Відсутня (нереляційна СУБД)	Добре
Інтеграція з іншими СУБД	Добре	Добре	Задовільно
Адміністративне керування	Добре	Добре	Відмінно
Графічні інструменти	Відмінно	Відмінно	Відмінно
Простота обслуговування	Відмінно	Відмінно	Відмінно
Побудова БД	Добре	Задовільно	Добре
Одночасний доступ декількох користувачів	Добре	Добре	Добре
Робота с декількома ЦП	Задовільно	Добре	Добре
Масштаб застосування	Підприємства малих, середніх та великих розмірів	Підприємства малих, середніх та великих розмірів	Підприємства малих та середніх розмірів

Прийнято рішення використовувати MSSQL.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		10

## Висновки до першого розділу

Було проаналізовано потреби та поставлено задачу проекту, а саме який в загальному має бути функціонал. Переглянуто проекти по схожій тематиці та визначено їх переваги та недоліки.

Обрано та обгрунтовано СУБД, для цього було проаналізовано та порівняно MS SQL Server, MongoDB та MySQL.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		11

## РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ІНДИВІДУАЛЬНИМИ ЗАМОВЛЕННЯМИ ПІДПРИЄМСТВА (НА ПРИКЛАДІ ПІДПРИЄМСТВА З ВИГОТОВЛЕННЯ МЕБЛІВ)

### 2.1 Аналіз інформаційних процесів

На етапі проектування потрібно скласти загальний алгоритм роботи програми (рис.2.1).

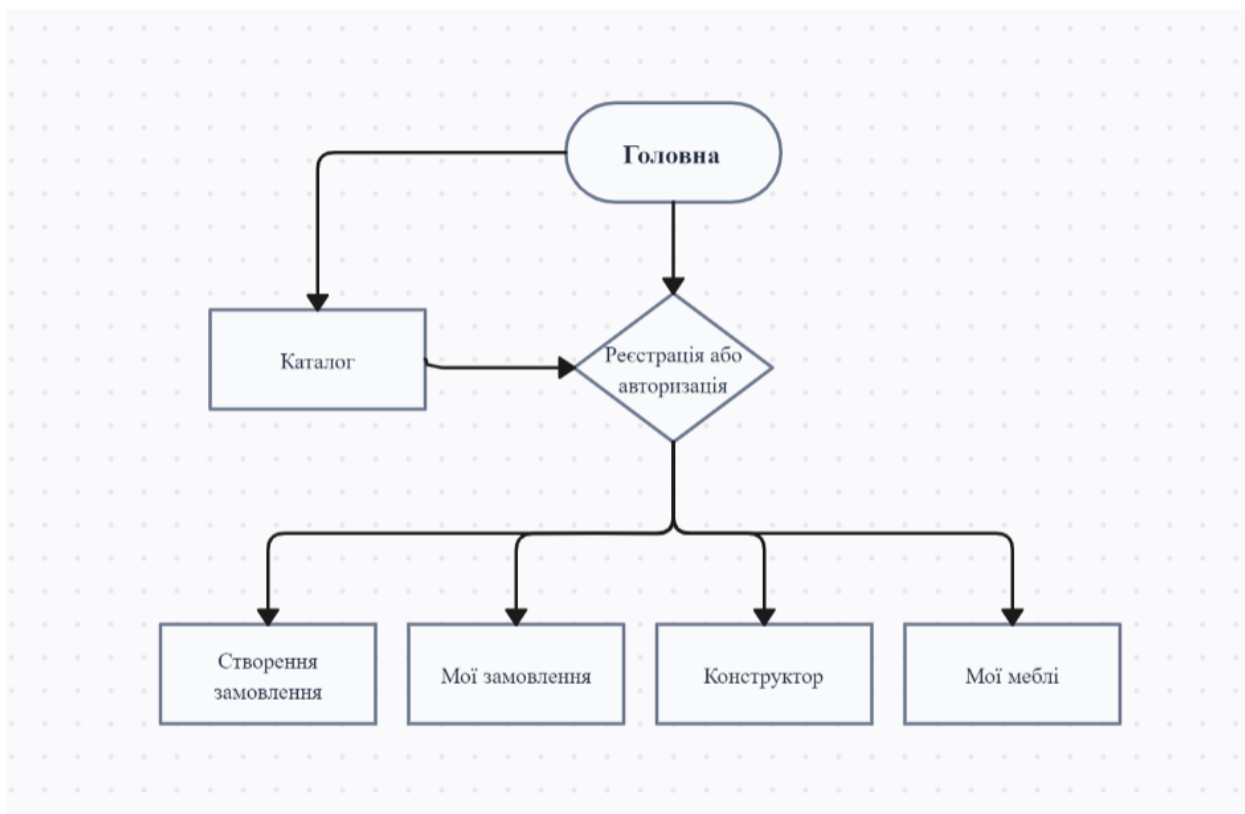


Рис. 2.1. Загальна схема роботи програми

Після завантаження сайту з'являється головна сторінка. Після цього очікується активність користувача.

Можна переключатись між вкладками, але тільки чотирма, після реєстрації або авторизації надається доступ до всіх вкладок.

#### **Вкладка «Головна\Home»:**

- Доступ до інших вкладок

#### **Вкладка «Каталог\Catalog»:**

- Перегляд меблів

#### **Вкладка «Реєстрація\Sing in»:**

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

- Можливість зареєструватися

**Вкладка «Авторизація\Register»:**

- Можливість авторизуватись

**Вкладка «Конструктор\Construct»:**

- Створити власні меблі

**Вкладка «Мої меблі\My furniture»:**

- Перегляд створених меблів

**Вкладка «Замовлення\Order»:**

- Можливість замовлення

**Вкладка «Мої замовлення\My order»:**

- Перегляд замовлень
- Видалення замовлень

**Вкладка «Вихід з аккаунту\Logout»:**

- Можливість виходу з аккаунту

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		13

## 2.2 Проектування структури бази даних сайту інформаційної системи управління індивідуальними замовленнями підприємства(на прикладі підприємства з виготовлення меблів)

Проектування бази даних відбувалося у Microsoft SQL Server. Створено та заповнено такі таблиці (рис.2.2):

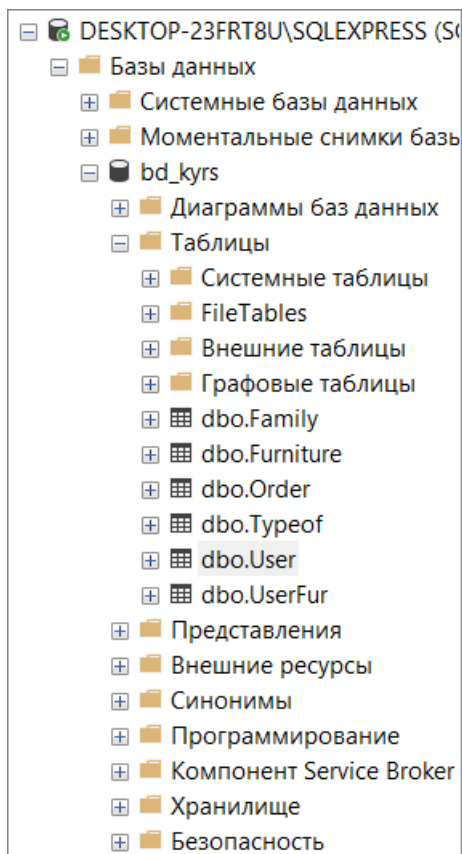


Рис.2.2 Створені таблиці бази даних

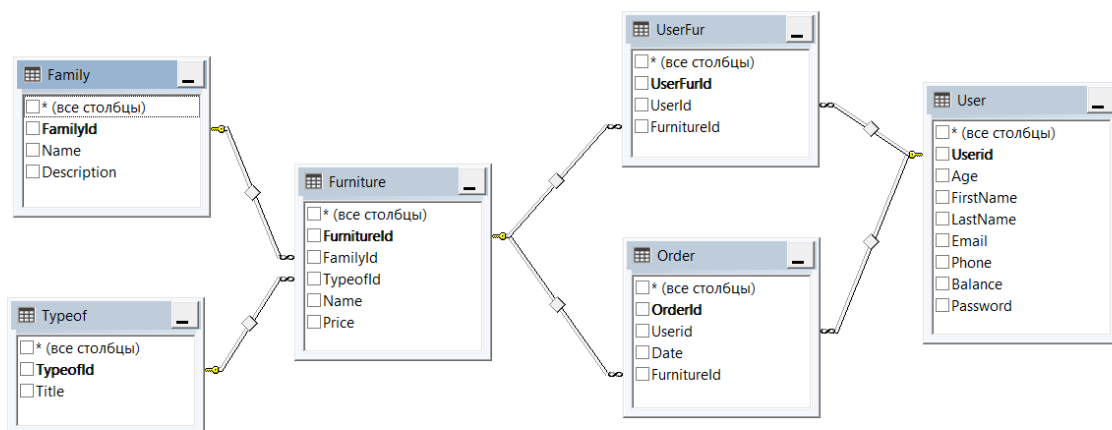


Рис.2.3 Діаграма «сутність-зв'язок» бази даних

Опис кожної таблиці, її полів та їх призначення.

Таблиця 2

Таблиця User			
Поле	Ключ	Тип	Призначення
Userid	PK	int	Ідентифікатор
Age	-	int	Вік
FirstName	-	nvarchar(50)	Ім'я
LastName	-	nvarchar(50)	Прізвище
Email	-	nvarchar(50)	Емейл
Phone	-	nvarchar(50)	Телефон
Balance	-	int	Баланс
Password	-	nvarchar(50)	Пароль

Таблиця 3

Таблиця UserFur			
Поле	Ключ	Тип	Призначення
UserFurId	PK	int	Ідентифікатор
UserId	FK	int	Тип
FurnitureId	FK	int	Тип

Таблиця 4

Таблиця Order			
Поле	Ключ	Тип	Призначення
OrderId	PK	int	Ідентифікатор
UserId	FK	int	Тип
Date	-	date	Дата замовлення
FurnitureId	FK	int	Тип

Таблиця 5

Таблиця Furniture			
Поле	Ключ	Тип	Призначення
FurnitureId	PK	int	Ідентифікатор
FamilyId	FK	int	Тип
TypeofId	FK	int	Тип
Name	-	nvarchar(50)	Назва
Price	-	int	Ціна

Таблиця 6

Таблиця Family			
Поле	Ключ	Тип	Призначення
FamilyId	PK	int	Ідентифікатор
Name	-	navrchar(50)	Назва
Description	-	nvarchar(50)	Опис

Таблиця 7

Таблиця Typeof			
Поле	Ключ	Тип	Призначення
Typeof	PK	int	Ідентифікатор
Title	-	nvarchar(50)	Назва



## Висновки до другого розділу

Складено загальний алгоритм роботи програми та продумано вкладки сайту та можливі дії користувача. Визначено можливості користувачів та авторизованих користувачів.

Створені та заповнено таблиці бази даних. Детально описано поля таблиць та їх призначення.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		17

### РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПІДСИСТЕМИ ОБРОБКИ ДАНИХ САЙТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ІНДИВІДУАЛЬНИМИ ЗАМОВЛЕННЯМИ ПІДПРИЄМСТВА (НА ПРИКЛАДІ ПІДПРИЄМСТВА З ВИГОТОВЛЕННЯ МЕБЛІВ)

#### 3.1 Проектування інтерфейсу обробки даних

Для написання програмного забезпечення інтерфейсу користувача було використано Visual Studio та Microsoft SQL Server. Дані середовища надають широкий вибір інструментів для створення та роботи з сайтом та базою даних.

Після запуску програми з'являється головна сторінка (рис. 3.1), де є привітання та посилання на каталог.

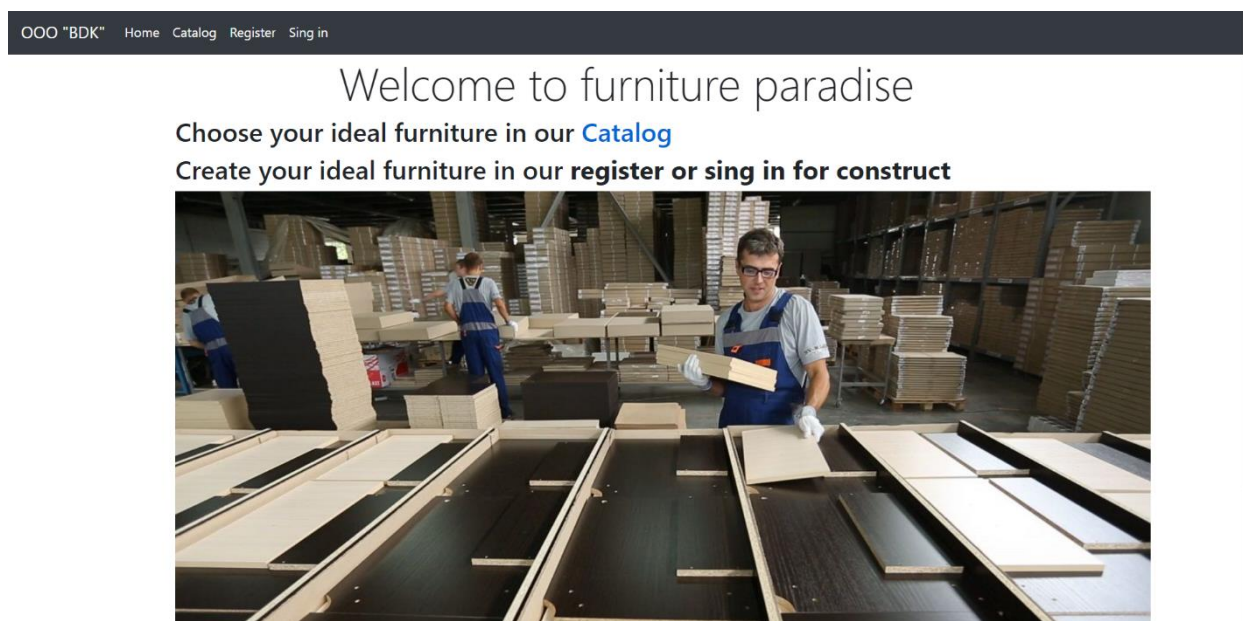


Рис.3.1 Головна сторінка

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		18

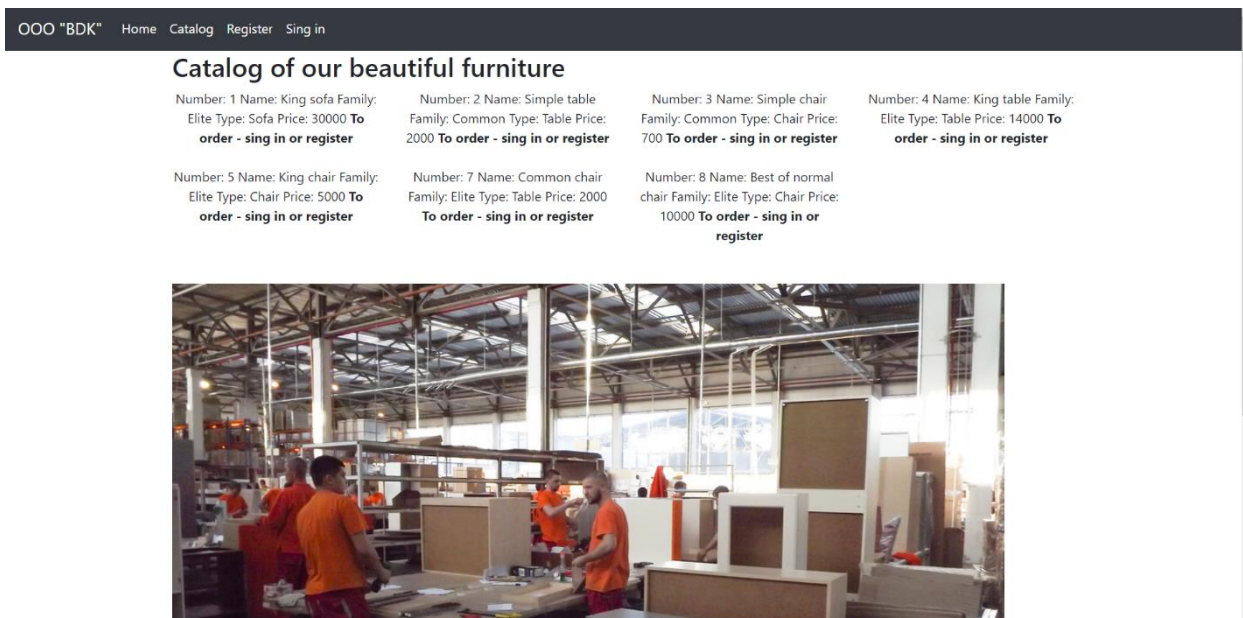


Рис.3.2 Каталог

Рис.3.3 Форма реєстрації

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

ООО "BDK"

Home

Catalog

Register

Sing in

Вход на сайт

Register

Введите Email

Введите пароль

Войти

© 2021 - ООО BDK

Рис.3.4 Авторизація

Після реєстрації або авторизації стають доступними всі вкладки та можливість замовлення.

ООО "BDK"

Home

Catalog

Construct

My furniture

Logout

Order

My order

Catalog of our beautiful furniture

Number: 1 Name: King sofa Family:

Elite Type: Sofa Price: 30000

Order

Number: 2 Name: Simple table

Family: Common Type: Table Price:

2000

Order

Number: 3 Name: Simple chair

Family: Common Type: Chair Price:

700

Order

Number: 4 Name: King table Family:

Elite Type: Table Price: 14000

Order

Number: 5 Name: King chair Family:

Elite Type: Chair Price: 5000

Order

Number: 7 Name: Common chair

Family: Elite Type: Table Price: 2000

Order

Number: 8 Name: Best of normal chair Family: Elite Type: Chair Price:

10000

Order

Рис.3.5 Каталог після авторизації

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				20
Змн.	Арк.	№ докум.	Підпис	Дата		

ООО "BDK"

Home

Catalog

Construct

My furniture

Logout

Order

My order

Construct your beautiful furniture

FamilyId

TypeofId

Name

Price

Construct

To choose type and family Typeof Family

Рис.3.6 Конструктор

ООО "BDK"

Home

Catalog

Construct

My furniture

Logout

Order

My order

Catalog of your beautiful furniture

Number: 8 Name: Best of normal

chair Family: Elite Type: Chair Price: 10000

Order

Рис.3.7 Каталог створених користувачем меблів

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				21
Змн.	Арк.	№ докум.	Підпис	Дата		



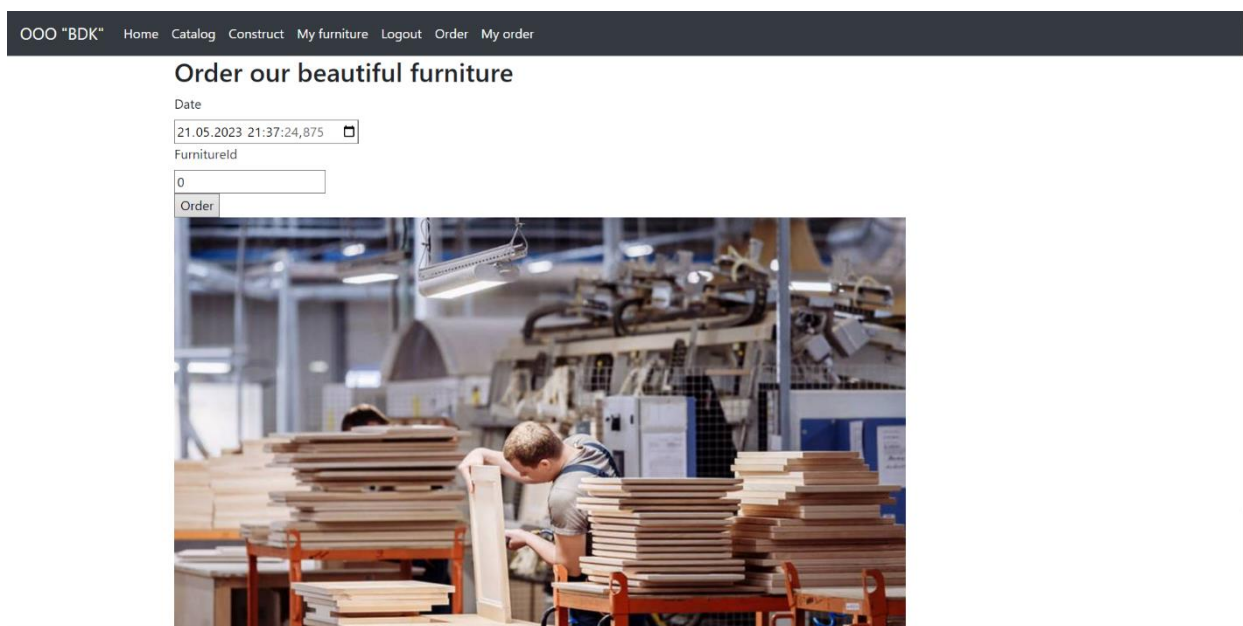


Рис.3.8 Сторінка формування замовлення

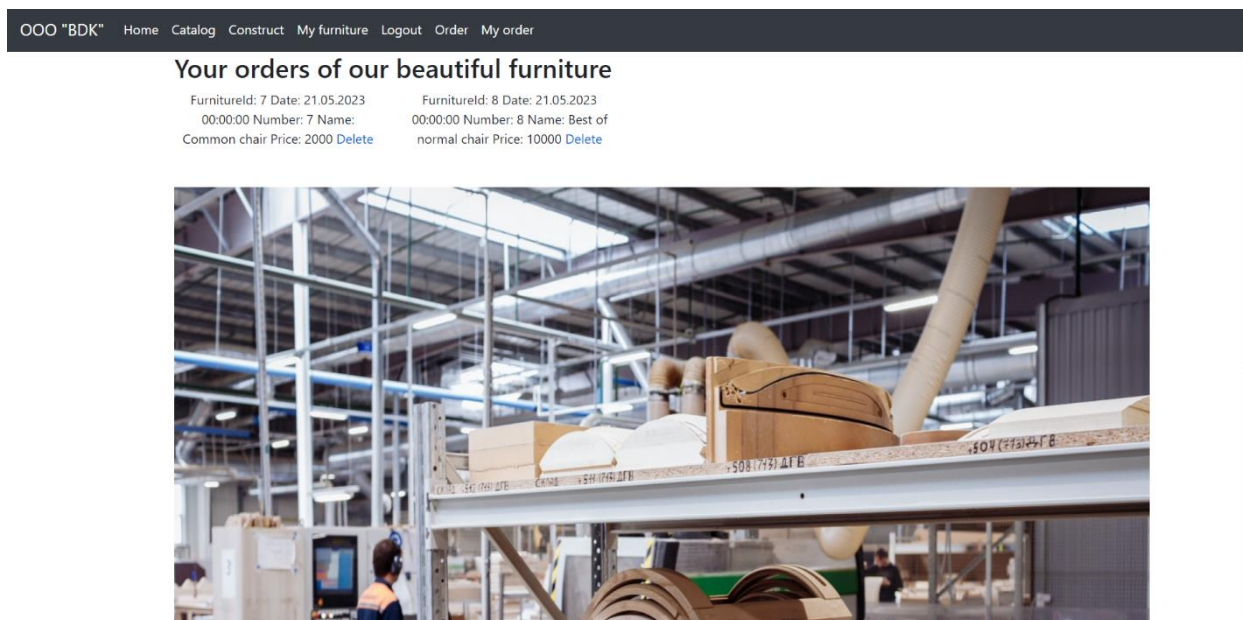


Рис.3.9 Сторінка замовлень користувача

Дії авторизованого та не авторизованого користувача розглянуті в пункті 3.3.

Інтерфейс зрозумілий, лаконічний та мінімалістичний. Все що потрібно користувачу є.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				22
Змн.	Арк.	№ докум.	Підпис	Дата		

### 3.2 Реалізація операцій обробки даних на сайті інформаційної системи управління індивідуальними замовленнями підприємства(на прикладі підприємства з виготовлення меблів)

Для реалізації проекту було використано СУБД MSSQL, Visual Studio та мову програмування C#, для роботи з БД був використаний інструмент ASP.NET.

Строка підключення до бази даних:

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "AllowedHosts": "*",
  "ConnectionStrings": {
    "DefaultConnection": "Server=DESKTOP-23FRT8U\\SQLEXPRESS;Database=bd_kyrs;Trusted_connection=True;TrustServerCertificate = True;MultipleActiveResultSets=true"
  }
}
```

Авторизація користувача.

```
[HttpGet]
public IActionResult Sing_in()
{
    return View();
}
[HttpPost]
[ValidateAntiForgeryToken]

public async Task<IActionResult> Sing_in(Sing_in model)
{
    if (ModelState.IsValid)
    {
        User user = await db.User.FirstOrDefaultAsync(u => u.Email == model.Email
&& u.Password == model.Password);
        if (user != null)
        {
            await Authenticate(model.Email);

            return RedirectToAction("Index", "Home");
        }
        ModelState.AddModelError("", "Некорректные логин и(или) пароль");
    }
    return View(model);
}

private async Task Authenticate(string userName)
{
    var claims = new List<Claim>
    {
        new Claim(ClaimsIdentity.DefaultNameClaimType, userName)
    }
}
```

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		23

```

    };
    ClaimsIdentity id = new ClaimsIdentity(claims, "ApplicationCookie",
ClaimsIdentity.DefaultNameClaimType, ClaimsIdentity.DefaultRoleClaimType);

    await
HttpContext.SignInAsync(CookieAuthenticationDefaults.AuthenticationScheme, new
ClaimsPrincipal(id));
}

```

## Реєстрація:

```

[HttpGet]
public IActionResult Register()
{
    return View();
}
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Register(RegisterViewModel model)
{
    if (ModelState.IsValid)
    {
        User user = await db.User.FirstOrDefaultAsync(u => u.Email ==
model.Email);
        if (user == null)
        {
            db.User.Add(new User { Email = model.Email, Password =
model.Password, Age = model.Age, Balance = model.Balance, FirstName = model.FirstName,
LastName = model.LastName, Phone = model.Phone });
            await db.SaveChangesAsync();

            await Authenticate(model.Email);

            return RedirectToAction("Index", "Home");
        }
        else
            ModelState.AddModelError("", "Некорректные логин и(или) пароль");
    }
    return View(model);
}

private UserContext db;
public AccountController(UserContext context)
{
    db = context;
}

```

## Відображення списку створених меблів користувачем.

```

[Authorize]
public IActionResult My_furniture()
{
    var email = User.Identity.Name;
    var id = DB.GetMyId(Connection, email);
    var idf = DB.GetMyUserFurNames(Connection, id[0].UserId);
    var model4 = DB.GetMyFurnitureNames(Connection, idf);

    foreach (var item in model4)
    {
        item.Name = item.Name + " Family: " + GetFamilyName(Connection,
item.FamilyId)[0].Name;
        item.Name = item.Name + " Type: " + GetTypeofName(Connection,
item.TypeofId)[0].Title;
    }
}

```

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				24
Змн.	Арк.	№ докум.	Підпис	Дата		



```

        return View(model4);
    }

    public static List<UserFur> GetUserFurNames(IDbConnection Connection)
    {
        var result = Connection.Query<UserFur>("SELECT * FROM [UserFur]").ToList();
        return result;
    }

    public static List<UserFur> GetMyUserFurNames(IDbConnection Connection, int id)
    {
        var result = Connection.Query<UserFur>($"SELECT * FROM [UserFur] WHERE [Userid] =('{id}')").ToList();
        return result;
    }

    public static List<User> GetMyId(IDbConnection Connection, string emeil)
    {
        var result = Connection.Query<User>($"SELECT [Userid] FROM [User] WHERE [Email] =('{emeil}')").ToList();
        return result;
    }

    public static List<Furniture> GetMyFurnitureNames(IDbConnection Connection, List<UserFur> id)
    {
        var result = Connection.Query<Furniture>($"SELECT * FROM [Furniture] WHERE [FurnitureId] =('{-1}')").ToList();
        foreach (var item in id)
        {
            var iiii = Connection.Query<Furniture>($"SELECT * FROM [Furniture] WHERE [FurnitureId] =('{item.FurnitureId}')").ToList();
            result.Add(iiii[0]);
        }
        return result;
    }
}

```

Інші важливі методи для роботи з БД та сайтом представлені у додатках.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		25

### 3.3 Організація звітності системи

У не авторизованого користувача є можливість зайти на головну сторінку, авторизуватися та зареєструватися, продивитися каталог, але без можливості замовити меблі. У зареєстрованого користувача є можливість замовляти меблі на сторінці каталогу та заходити на всі інші сторінки, крім авторизації та реєстрації, є можливість вийти з аккаунту. Звітність системи не реалізована.

#### Висновки до 3 розділу:

Спроековано інтерфейс обробки даних, детально описано роботу програмного додатку, очікувані дії користувача та їх результати. До кожного пункту додано відповідні скріншоти програми.

Приведено код найбільш важливих функцій роботи з БД.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		26

## РОЗДІЛ 4. АДМІНІСТРУВАННЯ БАЗ ДАНИХ

### 4.1 Розробка заходів захисту інформації в БД

Вхід відбувається за допомогою форми реєстрації або авторизації. Не авторизовані користувачі можуть лише переглядати головну сторінку та каталог без можливості замовити меблі. Авторизовані можуть як переглядати та замовляти меблі, створювати нові меблі, видаляти замовлення.

Ролі поділяються на авторизованих користувачів та не авторизованих користувачів.

Таблиця 8

Матриця доступу		
Таблиця \ Роль	Авторизований користувач	Не авторизований користувач
User	1	1, 2
UserFur	1, 2, 3	1
Order	1, 2, 3	1
Furniture	1, 2, 3	1
Typeof	1	1
Family	1	1

## 4.2 Налаштування параметрів роботи сервера

Для налаштування авторизації та реєстрації були використані можливості ASP.NET. А також бібліотеки Claims.

```
private async Task Authenticate(string userName)
{
    var claims = new List<Claim>
    {
        new Claim(ClaimsIdentity.DefaultNameClaimType, userName)
    };
    ClaimsIdentity id = new ClaimsIdentity(claims, "ApplicationCookie",
    ClaimsIdentity.DefaultNameClaimType, ClaimsIdentity.DefaultRoleClaimType);

    await
    HttpContext.SignInAsync(CookieAuthenticationDefaults.AuthenticationScheme, new
    ClaimsPrincipal(id));
}
```

У проекті було налаштовано під'єднання до БД.

### Висновки до 4 розділу:

Було вжито засоби захисту інформації в БД шляхом створення різних ролей та обмеженого доступу до ресурсів сайту. Розписано можливості кожної ролі, та в чому їх відмінність. Налаштовано коректну роботу сервера.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмоторя О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		28

## ВИСНОВКИ

Було розглянуто потреби та поставлено задачу проекту, визначено переваги та недоліки існуючих сайтів.

Було проведено порівняння СУБД, після чого обрано MSSQL для створення бази даних.

Складено загальний алгоритм роботи програми та кількість вкладок сайту та можливі дії користувача. Визначено можливості для різних ролей.

Було створено, заповнено та описано призначення таблиць бази даних.

Розписано основну поведінку роботи програмного додатку, дії користувача та їх результати з відповідними скріншотами.

Розписано дозволи кожної ролі.

Налаштовано параметри роботи сервера.

Створено звіт та презентацію. У результаті отримано робочу програму, яка справно виконує усі бажані дії та коректно працює при введенні різних значень.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		29

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Документація WEB стандартів [Електронний ресурс] – Режим доступу до ресурсу: <https://www.w3schools.com/>.
2. Керівництво по MSSQL [Електронний ресурс] – Режим доступу до ресурсу: <https://metanit.com/sql/mssql/>.
3. Форум з питаннями та відповідями [Електронний ресурс] – Режим доступу до ресурсу: <https://stackoverflow.com/>.
4. Переваги та недоліки різних СУБД [Електронний ресурс] – Режим доступу до ресурсу: <https://drach.pro/blog/hi-tech/item/145-db-comparison>.

		Линник М.Р.			«Житомирська політехніка».20.121.10.000 – ПЗ	Арк.
		Чижмотря О. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		30

# ДОДАТКИ

## Моделі таблиць для роботи з базою даних

## Файл ConstructModel.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.ComponentModel.DataAnnotations;

namespace BD_Kyrsach.Models
{
    public class ConstructModel
    {
        [Required]
        [Display(Name = "FamilyId")]
        public int FamilyId { get; set; }
        [Required]
        [Display(Name = "TypeofId")]
        public int TypeofId { get; set; }
        [Required]
        [Display(Name = "Name")]
        public string Name { get; set; }
        [Required]
        [Display(Name = "Price")]
        public int Price { get; set; }
    }
}
```

## Файл OrderModel.cs:

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace BD_Kyrsach.Models
{
    public class OrderModel
    {
        [Required]
        [Display(Name = "Date")]
        public DateTime Date { get; set; }
        [Required]
        [Display(Name = "FurnitureId")]
        public int FurnitureId { get; set; }
        [Display(Name = "UserId")]
        public int UserId { get; set; }
    }
}
```

## Файл RegisterViewModel.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
```



```

using System.Threading.Tasks;
using System.ComponentModel.DataAnnotations;

namespace BD_Kyrsach.Models
{
    public class RegisterViewModel
    {
        [Required]
        [Display(Name = "Email")]
        public string Email { get; set; }

        [Required]
        [Display(Name = "Age")]
        public int Age { get; set; }

        [Required]
        [Display(Name = "FirstName")]
        public string FirstName { get; set; }

        [Required]
        [Display(Name = "LastName")]
        public string LastName { get; set; }

        [Required]
        [Display(Name = "Phone")]
        public string Phone { get; set; }

        [Required]
        [Display(Name = "Balance")]
        public int Balance { get; set; }

        [Required]
        [DataType(DataType.Password)]
        [Display(Name = "Пароль")]
        public string Password { get; set; }
    }
}

```

### Файл Sing\_in.cs:

```

using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace BD_Kyrsach.Models
{
    public class Sing_in
    {
        [Required(ErrorMessage = "Не указан Email")]
        public string Email { get; set; }

        [Required(ErrorMessage = "Не указан пароль")]
        [DataType(DataType.Password)]
        public string Password { get; set; }
    }
}

```

### Файл DB.cs:

```

using Dapper;

```

```

using Microsoft.Data.SqlClient;
using Microsoft.Extensions.Configuration;
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using BD_Kyrsach.Models;
using Microsoft.AspNetCore.Mvc;
using System.Security.Claims;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Authentication.Cookies;
using Microsoft.EntityFrameworkCore;

namespace BD_Kyrsach.Models
{
    public class DB : Microsoft.EntityFrameworkCore.DbContext
    {
        public static List<User> GetUserNames(IDbConnection Connection)
        {
            var result = Connection.Query<User>("SELECT * FROM [User]").ToList();
            return result;
        }

        public class UserContext : Microsoft.EntityFrameworkCore.DbContext
        {
            public Microsoft.EntityFrameworkCore.DbSet<User> User { get; set; }
            public UserContext(DbContextOptions<UserContext> options)
                : base(options)
            {
                Database.EnsureCreated();
            }
        }

        public class FurnitureContext : Microsoft.EntityFrameworkCore.DbContext
        {
            public Microsoft.EntityFrameworkCore.DbSet<Furniture> Furniture { get; set; }
            public FurnitureContext(DbContextOptions<FurnitureContext> options)
                : base(options)
            {
                Database.EnsureCreated();
            }
        }

        public class UserFurContext : Microsoft.EntityFrameworkCore.DbContext
        {
            public Microsoft.EntityFrameworkCore.DbSet<UserFur> UserFur { get; set; }
            public UserFurContext(DbContextOptions<UserFurContext> options)
                : base(options)
            {
                Database.EnsureCreated();
            }
        }

        public class OrderContext : Microsoft.EntityFrameworkCore.DbContext
        {
            public Microsoft.EntityFrameworkCore.DbSet<Order> Order { get; set; }
            public OrderContext(DbContextOptions<OrderContext> options)
                : base(options)
            {
                Database.EnsureCreated();
            }
        }
    }
}

```

```

public static List<UserFur> GetUserFurNames(IDbConnection Connection)
{
    var result = Connection.Query<UserFur>("SELECT * FROM [UserFur]").ToList();
    return result;
}

public static List<UserFur> GetMyUserFurNames(IDbConnection Connection, int id)
{
    var result = Connection.Query<UserFur>($"SELECT * FROM [UserFur] WHERE
[UserId] =('{id}')").ToList();
    return result;
}

public static List<User> GetMyId(IDbConnection Connection, string emeil)
{
    var result = Connection.Query<User>($"SELECT [UserId] FROM [User] WHERE
[Email] =('{emeil}')").ToList();
    return result;
}

public static List<Furniture> GetMyFurnitureNames(IDbConnection Connection,
List<UserFur> id)
{
    var result = Connection.Query<Furniture>($"SELECT * FROM [Furniture] WHERE
[FurnitureId] =('{-1}')").ToList();
    foreach (var item in id)
    {
        var iiii = Connection.Query<Furniture>($"SELECT * FROM [Furniture] WHERE
[FurnitureId] =('{item.FurnitureId}')").ToList();
        result.Add(iiii[0]);
    }
    return result;
}

public static List<Order> GetMyOrderNames(IDbConnection Connection, int id)
{
    var result = Connection.Query<Order>($"SELECT * FROM [Order] WHERE [UserId]
=('{id}')").ToList();
    return result;
}

public static List<Order> GetOrderNames(IDbConnection Connection)
{
    var result = Connection.Query<Order>("SELECT * FROM [Order]").ToList();
    return result;
}

public static List<Furniture> GetFurnitureNames(IDbConnection Connection)
{
    var result = Connection.Query<Furniture>($"SELECT * FROM
[Furniture]").ToList();
    return result;
}

public static List<Furniture> GetFurnitureName(IDbConnection Connection, int id)
{
    var result = Connection.Query<Furniture>($"SELECT * FROM [Furniture] WHERE
[FurnitureId]=('{id}')").ToList();
    return result;
}

public static List<Family> GetFamilyName(IDbConnection Connection, int id)
{

```

```

        var result = Connection.Query<Family>($"SELECT [Name] FROM [Family] WHERE
[Familyid]=('{id}')).ToList();
        return result;
    }

    public static List<Typeof> GetTypeofName(IDbConnection Connection, int id)
    {
        var result = Connection.Query<Typeof>($"SELECT [Title] FROM [Typeof] WHERE
[Typeofid]=('{id}')).ToList();
        return result;
    }

    public static List<Family> GetFamilyNames(IDbConnection Connection)
    {
        var result = Connection.Query<Family>("SELECT * FROM [Family]").ToList();
        return result;
    }

    public static List<Typeof> GetTypeofNames(IDbConnection Connection)
    {
        var result = Connection.Query<Typeof>("SELECT * FROM [Typeof]").ToList();
        return result;
    }

    public static List<Order> GetOrderUserfurnitureId(IDbConnection Connection)
    {
        var result = Connection.Query<Order>("SELECT [UserfurnitureId] FROM
[Order]").ToList();
        return result;
    }

    public static void DeletOrder(IDbConnection Connection, int OrderId)
    {
        Connection.Query<Order>($"DELETE FROM [Order] WHERE [OrderId]=('{OrderId}')");
    }

    public class User
    {
        public int UserId { get; set; }

        public int Age { get; set; }

        public string FirstName { get; set; }

        public string LastName { get; set; }

        public string Email { get; set; }

        public string Phone { get; set; }

        public int Balance { get; set; }

        public string Password { get; set; }
    }

    public class UserFur
    {
        public int UserFurId { get; set; }

        public int UserId { get; set; }

        public int FurnitureId { get; set; }
    }

```

```

public class Order
{
    public int OrderId { get; set; }

    public int UserId { get; set; }

    public int FurnitureId { get; set; }

    public DateTime Date { get; set; }
}

public class Furniture
{
    public int FurnitureId { get; set; }

    public int FamilyId { get; set; }

    public int TypeofId { get; set; }

    public string Name { get; set; }

    public int Price { get; set; }
}

public class Family
{
    public int FamilyId { get; set; }

    public string Name { get; set; }

    public string Description { get; set; }
}

public class Typeof
{
    public int TypeofId { get; set; }

    public string Title { get; set; }
}
}
}

```