**Міністерство освіти і науки України**

**Національний університет «Львівська політехніка»**

**Кафедра ЕОМ**



Звіт

до лабораторної роботи № 3

з дисципліни: «Інженерія програмного забезпечення»

на тему : «Система автоматизації замовлень в закладах швидкого харчування»

Варіант №16

Виконав:

ст.гр. КІ-34

Палій Н.С.

Прийняв:

Цигилик Л.О.

**Львів 2022**

**Тема:** Розробка серверної частини. Розробка комунікації за протоколом TCP. Підключення серверного модуля до БД.

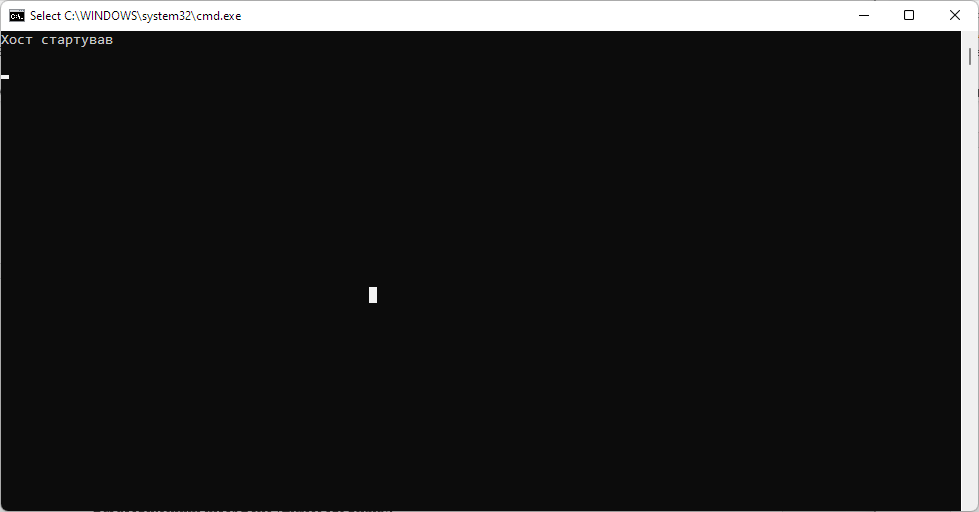
**Мета:** Розробити консольну аплікацію що буде підтримувати зв’язок по протоколу TCP/IP, отримувати дані та записувати у БД. Також, згідно деякої команди, вичитувати з БД необхідну інформацію та передавати по TCP протоколу на клієнтську частину.

**Варіант 16 :**

**Система автоматизації замовлень в закладах швидкого харчування.**

**Виконання :**

**Вигляд запущеної програми-сервера :**

****

*Рис. 1.1 – Вікно програми-сервера.*

**Список функцій сервера :**

**Функції пов’язані з діяльністю користувача :**

1. public string Show\_Cart()  
   Функція виводить дані з БД вікно «Кошика».
2. public void DeleteFromCart(int id)

Функція видаляє дані з БД після того, як замовлення здійснене та оплачене.

1. public void ToCook(int id)  
   Функція приймає дані з інших функцій та статичних змінних та записує замовлення, що оплачене в БД, в процес виконання.

**Функції пов’язані з діяльністю адміністратора та персоналу :**

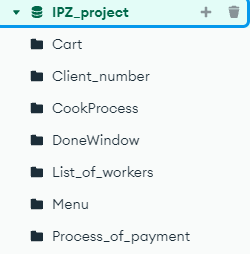
1. public int Authorization(string email, string password, string position)  
   Функція приймає дані електронної пошти, паролю, та позиції працівника та повертає фіксовані значення для опрацювання помилок та корекних відповідей.
2. public string Show\_Orders\_By\_Number(int id)  
   Функція приймає значення id та виводить дані з таблиці, що дорівнюють id.
3. public void Delete\_Done\_Order(int id)

Функція видаляє поля з таблиці, які позначено адміном виконаними.

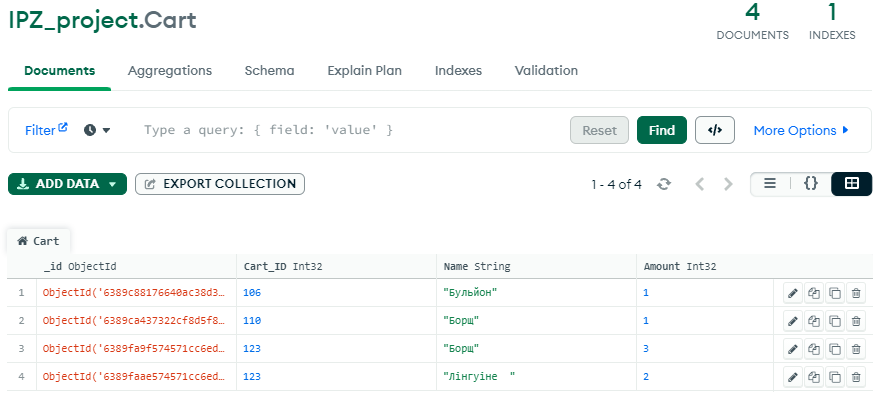
**Функції пов’язані з діяльністю монітора з живою чергою :**

1. public string Show\_Done\_Window(bool stat)  
Функція виводить замовлення на монітор із сортуванням готових та не готових.

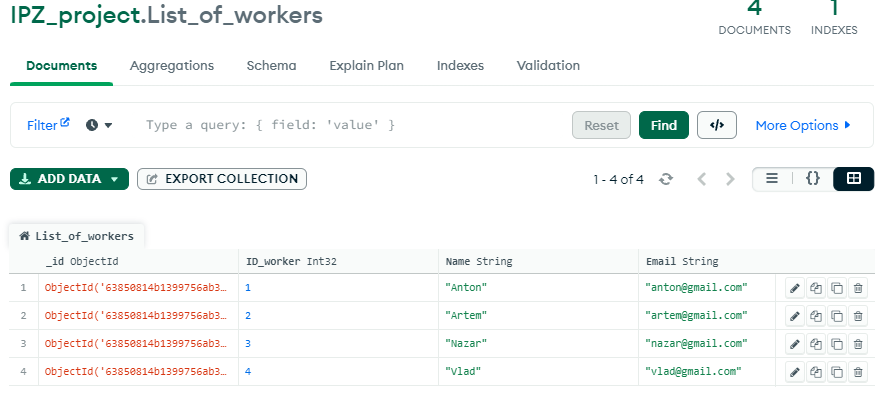
**Вигляд бази даних :**



*Рис. 2.1 – База даних і таблиці в ній.*



*Рис. 2.2 – Таблиця кошика.*

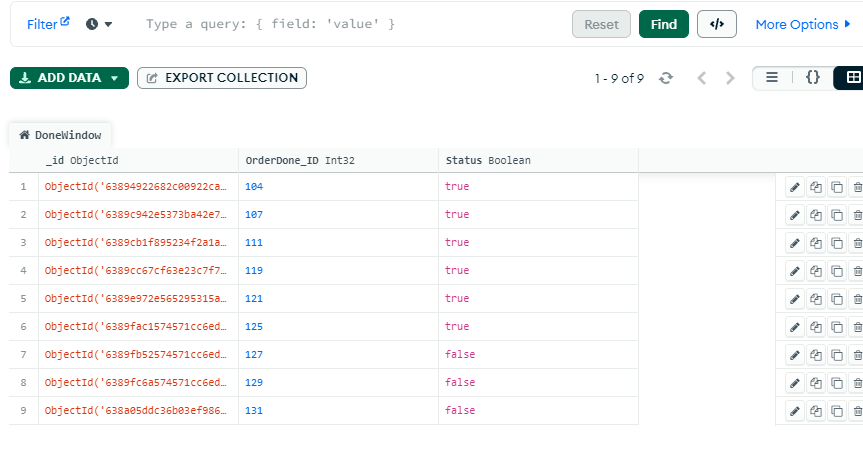


*Рис. 2.3 – Таблиця з працівниками.*

*Зображення, що містить текст

Автоматично згенерований опис*

*Рис. 2.4 – Таблиця з замовленнями в обробці*

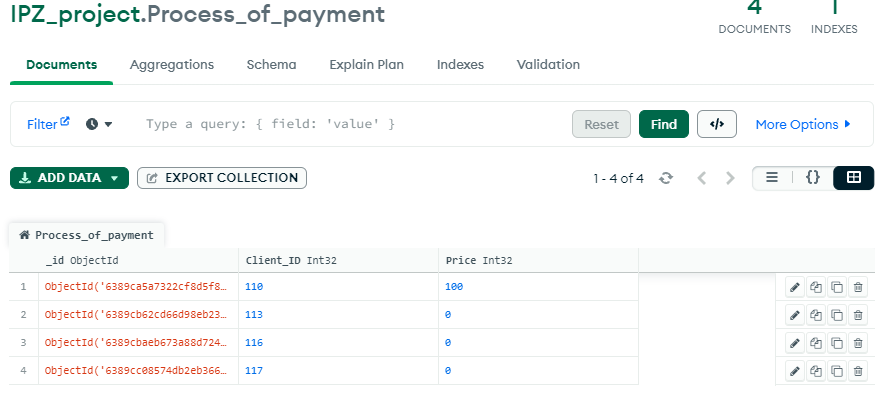
**

*Рис. 2.5 – Таблиця з замовленнями в обробці чи готові (для монітора)*

*Зображення, що містить стіл

Автоматично згенерований опис*

*Рис. 2.6 – Таблиця з стравами*

**

*Рис. 2.7 – Таблиця з процесом оплати*

**Опис функцій запису і зчитування з бд :**

**Запис замовлення :**

public void CreateCartRecord(int id, int amount)

{

string connectionString = "mongodb://localhost:27017";

string databaseName = "IPZ\_project";

string collectionName = "Cart";

var client = new MongoClient(connectionString);

var db = client.GetDatabase(databaseName);

var cartsCollection = db.GetCollection<CartModel>(collectionName);

var cart = new CartModel { Cart\_ID = start\_id, Name = ReturnName(id), Amount = amount, Price = ReturnPrice(id)\*amount, Location = GetLocation()};

cartsCollection.InsertOne(cart);

}

**Оновлення запису :**

public void SetStatusDoneWindow(int id,bool stat)

{

string connectionString = "mongodb://localhost:27017";

string databaseName = "IPZ\_project";

string collectionName = "DoneWindow";

var client = new MongoClient(connectionString);

var db = client.GetDatabase(databaseName);

var collection = db.GetCollection<DoneModel>(collectionName);

DoneModel bill = collection.Find(x => x.OrderDone\_ID.Equals(id)).FirstOrDefault();

bill.Status = stat;

collection.ReplaceOne(x => x.OrderDone\_ID.Equals(id), bill);

}

**Зчитування замовлень:**

public string Show\_Orders(int id)

{

string connectionString = "mongodb://localhost:27017";

string databaseName = "IPZ\_project";

string collectionName = "CookProcess";

var client = new MongoClient(connectionString);

var db = client.GetDatabase(databaseName);

var cartsCollection = db.GetCollection<CookModel>(collectionName);

var result = cartsCollection.Find(\_ => true).ToListAsync().Result;

string[] order = new string[CountOrders()];

int prev\_order = 0;

int counter = 0;

foreach (var user in result)

{

if (prev\_order != user.Order\_ID && user.Status == false)

{

order[counter] = user.Order\_ID.ToString() + ":\n";

order[counter] += user.Order;

counter += 1;

}

else if (user.Status == false)

{

order[counter - 1] += user.Order;

}

prev\_order = user.Order\_ID;

}

return order[id];

}

**Висновок :** Я розробив консольну аплікацію що буде підтримувати зв’язок по протоколу TCP/IP, отримувати дані та записувати у БД. Також, згідно деякої команди, вичитувати з БД необхідну інформацію та передавати по TCP протоколу на клієнтську частину.