**ЗМІСТ**

**[ВСТУП](#_Toc9210382)** [5](#_Toc9210382)

**[1 ПОСТАНОВКА ЗАДАЧІ](#_Toc9210383)** [6](#_Toc9210383)

[1.1 Інженерна постановка задачі 6](#_Toc9210384)

**[2 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ](#_Toc9210407)** [7](#_Toc9210407)

[2.1 Вимоги до функціональних характеристик системи 7](#_Toc9210408)

[2.2 Вимоги до складу та параметрів технічних засобів 8](#_Toc9210416)

[2.3 Вимоги до програмної сумісності 8](#_Toc9210421)

**[3 ПРОЕКТУВАННЯ СИСТЕМИ](#_Toc9210407)** [9](#_Toc9210407)

[3.1 Проектування інтерфейсу 9](#_Toc9210426)

[3.1.1 Інтерфейс сторінки SignIn 9](#_Toc9210445)

[3.1.2 Інтерфейс сторінки Main 1](#_Toc9210452)0

[3.1.3 Інтерфейс сторінки Admin 1](#_Toc9210458)0

[3.2 Проектування бази даних 1](#_Toc9210463)2

[3.3 Проектування структур системи 1](#_Toc9210481)4

[4](#_Toc9210506) **[КЕРІВНИЦТВО КОРИСТУВАЧА](#_Toc9210506)** [1](#_Toc9210506)7

[4.1 Область застосування 1](#_Toc9210507)7

[4.2 Короткий опис можливостей 1](#_Toc9210509)7

[4.3 Запуск програми 1](#_Toc9210515)7

[4.4 Робота з головною формою. 1](#_Toc9210517)7

[4.7 Вихід з програми 1](#_Toc9210532)8

**[6 СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ](#_Toc9210534)** [1](#_Toc9210534)9

**[7 ДОДАТКИ](#_Toc9210535)** [52](#_Toc9210535)

**ВСТУП**

Сьогодні інтернет-магазини дуже популярні, їх розглядають і як майданчик для збільшення обсягів продажів, і для створення бізнесу з нуля. При цьому дійсно корисної інформації про те, що таке інтернет-магазин, навіщо він потрібен, з чого він складається, як організувати його роботу і про багатьох інших важливих нюансах, знайти на просторах Інтернету практично неможливо.

права в тому, що сьогодні інтернет-магазин - це модне поняття, таке ж, як, наприклад, стартап.

Всі чули, що це дуже потрібно і дуже вигідно, але що це таке, кому і навіщо це треба, більшість навіть не замислюється.  
 У цій курсовій я покажу свої навички роботи з платформою .NET.

**1 ПОСТАНОВКА ЗАДАЧІ**

**1.1 Інженерна постановка задачі**

Мета роботи – створення магазину із застосуваннями фреймворку Entity Framework на базі платформи .Net.

Перед розробкою було поставлено наступні задачі, а саме:

* скласти технічне завдання
* скласти UML діаграму і діаграму класів, стану, послідовності
* розробити інтуїтивно зрозумілий ергономічний користувацький інтерфейс.

Перед розробкою магазину було сформовано наступні обмеження:

* розробка повинна проводить на базі платформи .Net. та за допомогою її мов програмування;
* програма повинна мати більше однієї форми;
* додаток повинен адмін вікно и вікно для користувача;
* все повинно зберігатися у базі даних.

**2 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

**2.1 Вимоги до функціональних характеристик системи**

Необхідно зберігати наступну інформацію про книгах:

- назва книги

- ПІБ автора

- назва видавництва

- кількість сторінок

- жанр

- рік видання

- собівартість ціна

- для продажу

Додаток має дозволяти:

- додавати книги

- видаляти книги

- редагувати параметри книг

Додаток має надати функціональність з сортировки книг за такими параметрами:

- назва книги

- автор жанр

- ціна

-рік видання

Необхідно передбачити можливість входу за логіном і паролем

При покупці книги в магазині, дані про покупку відправляються на пошту

**2.2 Вимоги до складу та параметрів технічних засобів**

Програма повинна працювати на ІВМ-сумісних персональних комп’ютерах на базі операційної системи Windows.

**2.3 Вимоги до програмної сумісності**

Для коректної роботи програми, машини, на якій відбуватиметься її запуск повинні відповідати наступним програмним вимогам:

* операційна система Windows 10, збірка 1629;
* платформа – 64х.
* .Net версія 4.7.2

**3 ПРОЕКТУВАННЯ СИСТЕМИ**

**3.1 Проектування інтерфейсу**

Сучасний додаток повинен мати інтуїтивно зрозумілий та привабливий інтерфейс. Проектування інтерфейсу відбувалось на платформі Windows Forms.

Windows Forms дає широку можливість для настроювань візуальних ресурсів. Тут є можливість налаштувати зображення для екрану-заставки, емблеми, логотипу, задати іконки тощо.

**3.1.1 Інтерфейс сторінки SignIn**

Меню авторизації (рис.1.1).



Рисунок 3.1 – Меню авторизації

Меню авторизації складаєтся:

- двух полів для вводу логіну та пароля

- кнопки для перевірки логіну та пароля

Для назв кнопок пунктів меню та кнопок на інших сторінках додатку використано шрифт Microsoft Sans Serif; 15,25pt.

**3.1.2 Інтерфейс сторінки Main**

**Main** – умовна назва сторінки(1.2), яка містить в собі елементи DataGridView для відображення товарів. Кнопку для відображення товарів у корзині, поля для вводу інформації користувача, кнопка “Купити”, кнопка “Вийти”, Combobox у якому є можливість вибрати за якими характеристиками сортирувати товари.

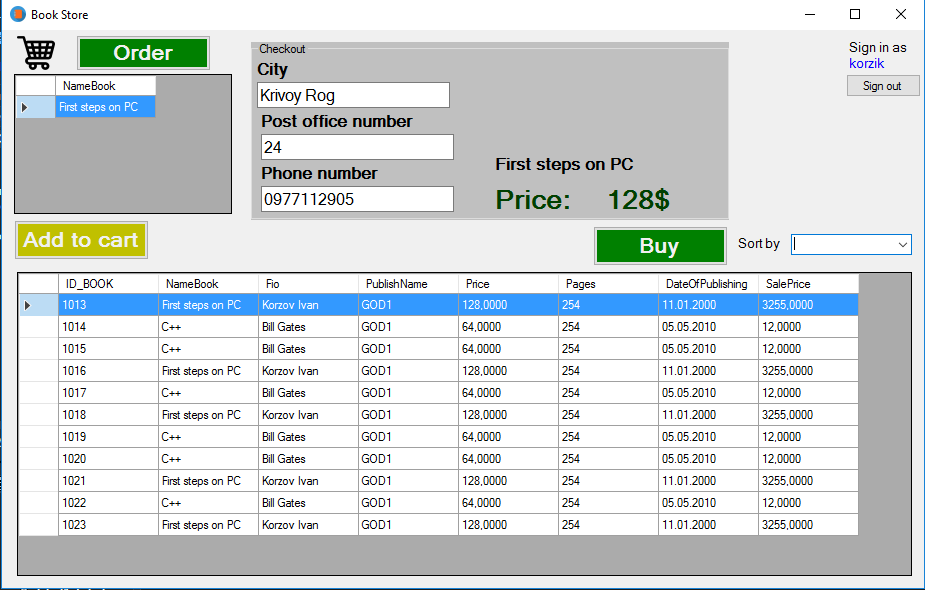
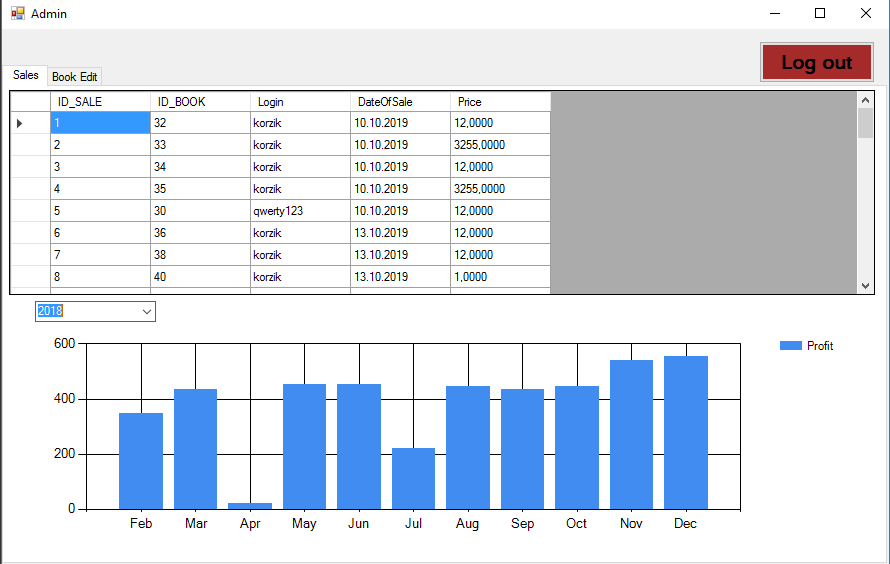


Рисунок 3.2 – Сторінка Main

**3.1.3 Інтерфейс сторінки Admin**

Сторінка Admin дає можливість перегланути закази користувачів, переглянути статистику продаж за рік,переглянути список товарів, додати новий товар(книгу), рудегувати товар, видалити товар.



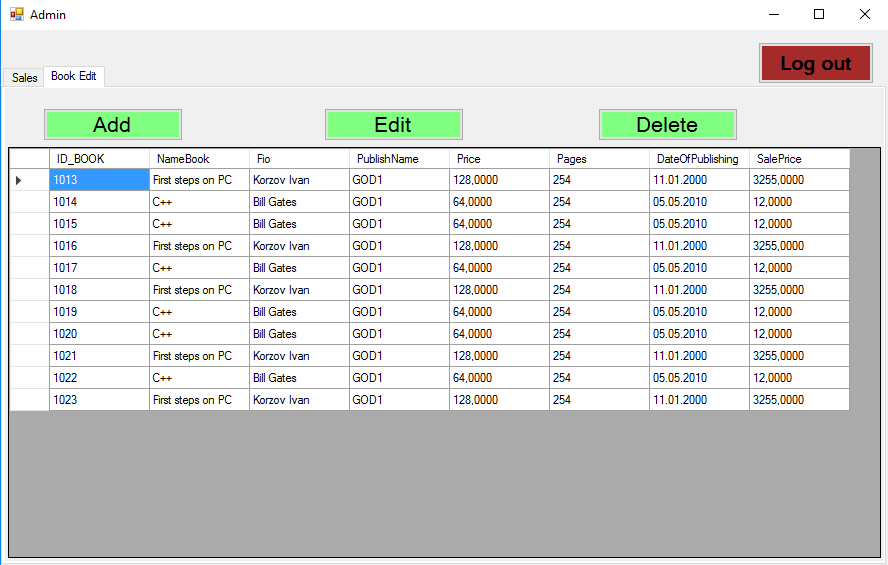


Рисунок 3.3 – Сторінки Admin

**3.2 Проектування бази даних**

База даних (БД) – це сукупність даних, зв’язаних між собою певною тематикою. Ці дані зберігаються на машинних носіях в упорядкованому вигляді. Наряду з поняттям баз даних існує поняття системи управління базами даних (СУБД) - це комп’ютерна програма, котра дозволяє керувати базою даних, тобто створювати та працювати з нею.

У цьому додатку робота ведеться з СУБД Microsoft SQL Server. Для зручного управління базами даних та різноманітними налаштуваннями у MS SQL Server існує спеціальна утіліта SQL Server Management Studio (SSMS). Дану утіліту можна використовувати для створення баз даних, таблиць, написання та виконання запитів до БД тощо.

Корисними даними, які можливо отримати з додатку, є наступні:

- Назва книги

- Автор книги

- Ціна книги

- Дата видання

- Продажі

- Логіни та паролі користувачів

- Корзину користувача

Для оперування цими даними, у Management Studio створимо базу даних. Для цього з’єднаємось із сервером. Аутентифікація – Windows (рис.3.4).

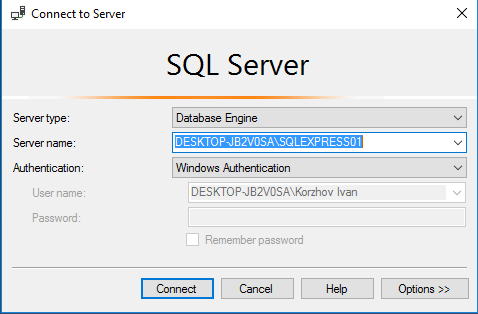


Рисунок 3.4 – Підключення до серверу Sql

Тепер, коли з’єднання з сервером успішне, створимо базу BookStore (рис.3.5).

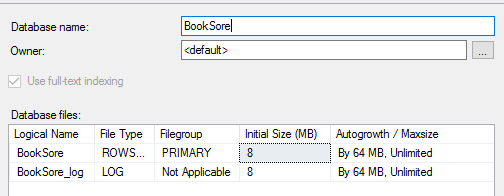


Рисунок 3.5 – Створення бази

Для зберігання даних, отриманих з додатку, створимо таблиці:

Users, Cart, Books, Sales (рис.3.6).

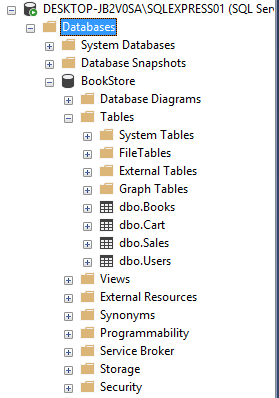


Рисунок 3.6 – Архітектура БД та таблиці

**3.3 Проектування структур системи**

Для побудови діаграми класів скористаємось додатком Astah Community (дод. 2).

Програма ShopStore складається із 5 сторінок, які представляють собою класи, наслідувані від класу Page.

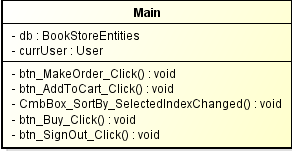


Рисунок 3.7 – Діаграма класу Main

Клас MainPage містить поля (рис.3.7), які становлять клас User який зберігає користувача з яким працює вікно. BookStoreEntities обьєкт збегігає модель для Entity.

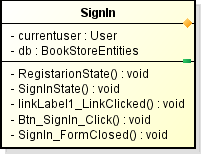


Рисунок 3.8 – Діаграма класу SignIn

Сторінка створенна для можливості аторизаціі та регістрації у додатку,

Містить такі ж поля я і головне вікно.

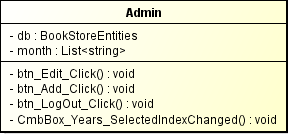


Рисунок 3.9 – Діаграма класу Admin

Сторінка творена для адміну, у який він може додати(метод btn\_Add\_Click), редагувати(метод btn\_Edit\_Click) і видаляти товари(книги) з бази данних. На вікні є комобокс у якого є метод зміни індексу за допомогої якого відбуваєтся зміна графіку.

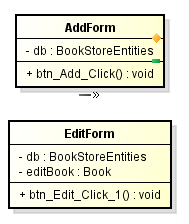


Рисунок 3.10 – Діаграма класів AddForm і EditForm

Ці класси створені для редагування таблиці Books.

**4 КЕРІВНИЦТВО КОРИСТУВАЧА**

**4.1 Область застосування**

Дане керівництво користувача встановлює порядок роботи з програмою магазина книг

**4.2 Короткий опис можливостей**

Можливості які надає програма:

- додавати товари до корзини

- покупати товари

**4.3 Запуск програми**

Під час запуску програми користувачча просить ввести логін та пароль, якщо користувач не має аккаунта, він може зареєструватися.

**4.4 Робота з головною формою.**

Після авторизації, користувача зустрічає головна форма.

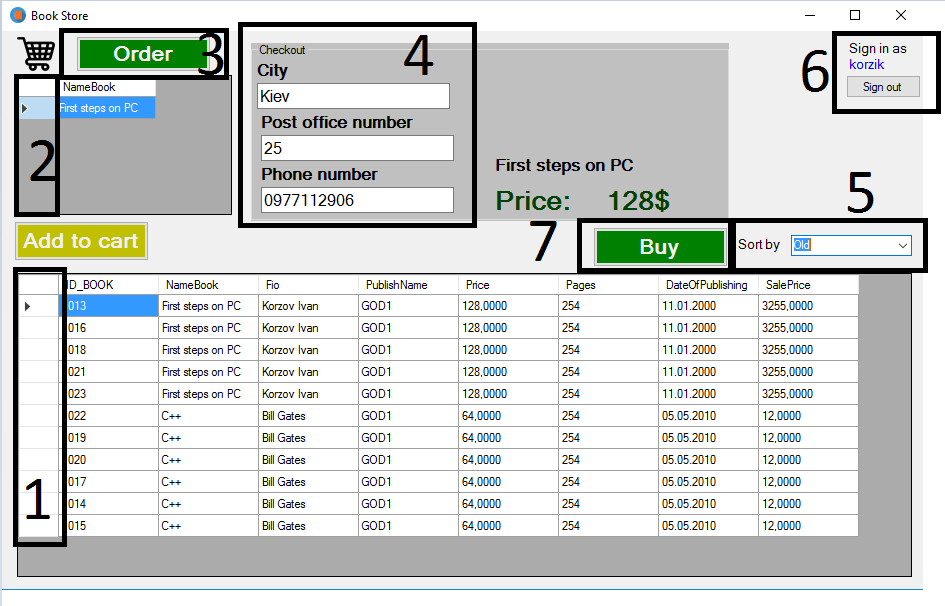


Рисунок 4.1 – Головна форма

Для того щоб додати товар до корзини виберіть товар на полосі зліва, після чого він опинится у корзині(позначенно 1 на рис. 4.1)

Для того щоб купити товар треба вибрати товар на полосі зліва(позначенно 2 на рис. 4.1) и натиснути кнопку “Order”(позначенно 3 на рис. 4.1), після чого інформація про товар буду відображена на панелі зліва. Для оформлення заказу треба ввести у три поля для вводу інформацію(Город, номер почти, номер телефону)(позначенно 4 на рис. 4.1) і натиснути кнопку “Buy”(позначенно 5 на рис. 4.1).

Також є можливість сортувати товари по різнім характеристикам, для цього натисніть на комбобкс и виберіть потрібну характеристику (позначенно 5 на рис. 4.1).

Для виходу з аккаунту натиснить кнопку “Sign out”(позначенно 6 на рис. 4.1).

**4.5 Вихід з програми**



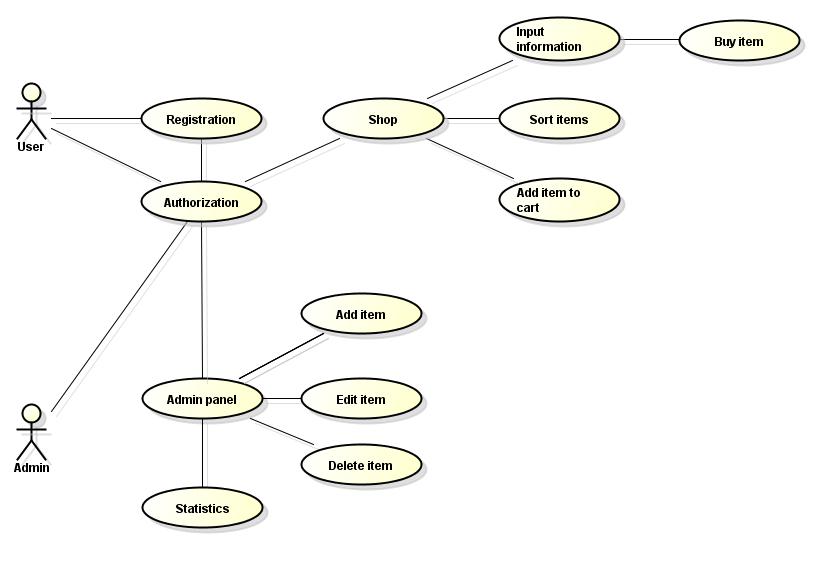
Для вихіду з программи натисніть хрестик у правому верхньому куту.

**6 СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

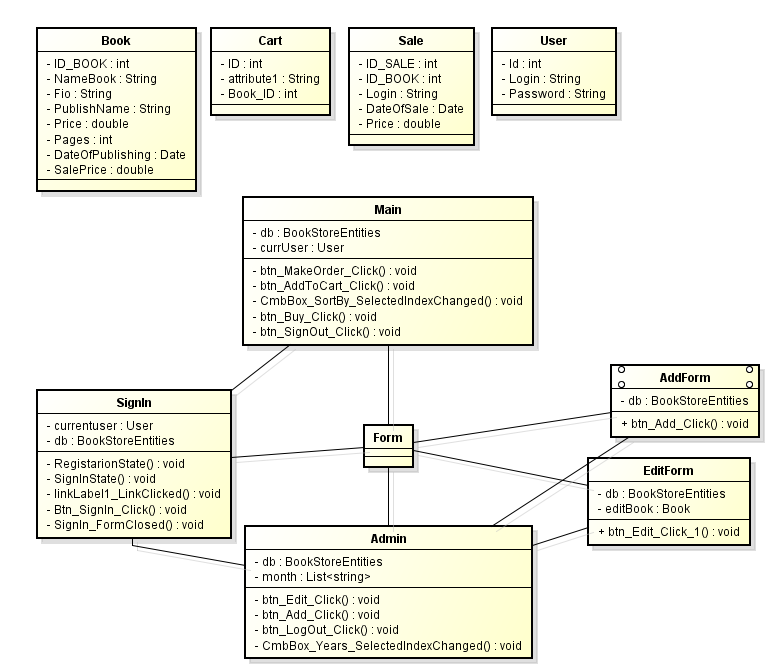
1. What is Entity? [Электронный ресурс]: интернет-статья. – Режим доступа: <https://stackoverflow.com/questions/2785589/what-is-an-entity-why-is-it-called-entity>, свободный. – Язык: Eng.

**7 ДОДАТКИ**

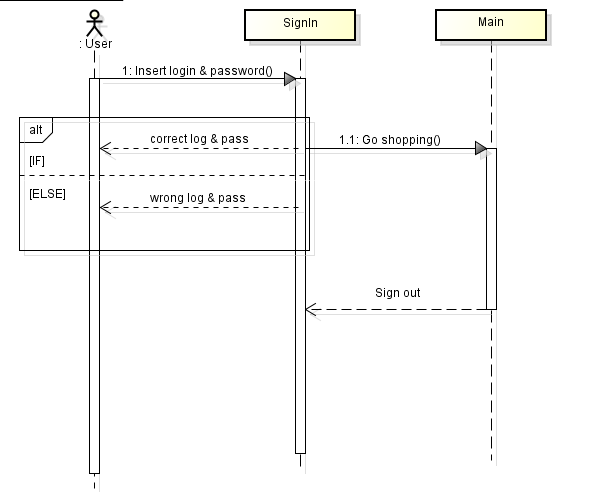
Додаток 1 – Діаграма варіантів користування



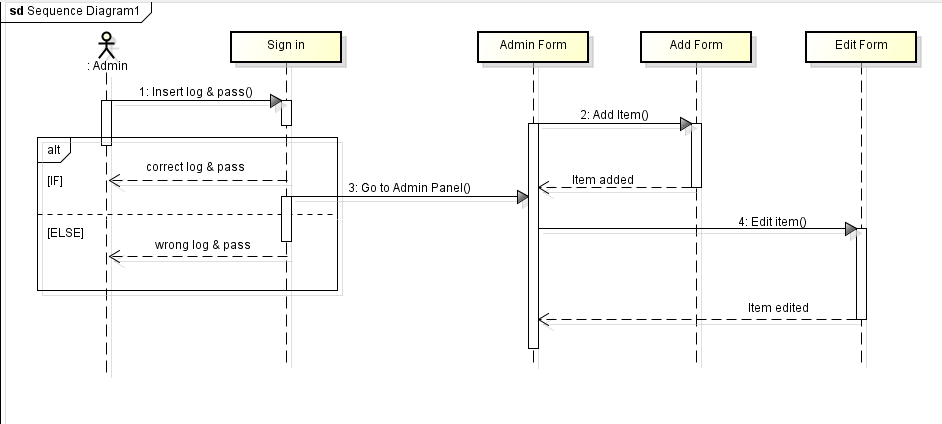
Додаток 2 – Діаграма класів



Додаток 3 – Діаграма станів



Додаток 3 – Діаграма станів (продовження)



Додаток 4 – Код

//Код авторизації і регістрації

Regex regex = new Regex("^[a-zA-Z0-9]+([.\_]?[a-zA-Z0-9]+){5}$");

string input\_login = txtBox\_login.Text;

string input\_password = txtBox\_password.Text;

if (regex.Match(input\_login).Success && regex.Match(input\_password).Success)

{

txtBox\_password.BackColor = Color.White;

txtBox\_login.BackColor = Color.White;

if (signInForm) //SIGN IN

{

//ADMIN

if (input\_login == admin\_login && input\_password == admin\_pass)

{

Admin adminForm = new Admin();

this.Hide();

if (adminForm.ShowDialog() == DialogResult.OK)

{

this.Show();

}

else

{

Application.Exit();

}

}

else

{

int corect\_login = db.Users.Where(user => user.Login == input\_login).Count();

if (corect\_login == 1)

{

var corect\_password = db.Users.Where(user => user.Login == input\_login).Where(user => user.Password == input\_password.ToString()).Count();

if (corect\_password == 1)

{

this.DialogResult = DialogResult.OK;

currentUser = new User() { Login = input\_login, Password = password };

}

else

{

txtBox\_password.BackColor = Color.PaleVioletRed;

MessageBox.Show("Incorrect password");

}

}

else

{

txtBox\_login.BackColor = Color.PaleVioletRed;

MessageBox.Show("Incorrect username");

}

}//END

} //END

else //Create Acount

{

using (BookStoreEntities db = new BookStoreEntities())

{

User newUser = new User() { Login = txtBox\_login.Text, Password = txtBox\_password.Text };

db.Users.Add(newUser);

db.SaveChanges();

MessageBox.Show("Created new user");

}

}//End

}

else

{

if (!regex.Match(input\_password).Success)

{

txtBox\_password.BackColor = Color.PaleVioletRed;

}

if(!regex.Match(input\_login).Success)

{

txtBox\_login.BackColor = Color.PaleVioletRed;

}

}

}