**Лабораторная работа по курсу**

**«Технология ADO.NET» №1**

**Тема: Введение в ADO.NET.**

**Цель:**

Закрепить у слушателей практические навыки и теоретические знания для работы с базами данных с использованием технологии ADO.NET.

**Необходимые инструменты:** MS Visual Studio 2013, SQL Server

**Документация:** Конспект, Литература

**Ориентировочное время исполнения:** 2 часа.

**Задание 1. «Авторизация».**

**Цель:** Разработать приложение, реализующее функционал системы авторизации и регистрации пользователей, с **использованием WinForms и ADO.NET**.

При разработке **можно** воспользоваться шаблоном: ***Authorization(Template)***

**Общие требования:**

При запуске программы открывается окно «Авторизация…» (рисунок 1-3). Пользователь вводит логин и пароль, которые сверяются со значениями, хранящимися в базе данных. При успешной авторизации пользователю выводится страница с уведомлением об успешной авторизации с указанием его личных данных (рисунок 4).

Если пользователь забыл пароль, то он переходит по ссылке в окно «Восстановление пароля» (рисунок 5). После успешного восстановления пароля пользователь может войти в систему.

Если пользователь не зарегистрирован в системе, то переходит по ссылке в окно «Регистрация пользователя» (рисунок 6). После успешной регистрации пользователь может повторить вход в систему.

Если пользователь ввел неправильные логин и пароль, то ему выдается соответствующее сообщение.

**Требования к базе данных:**

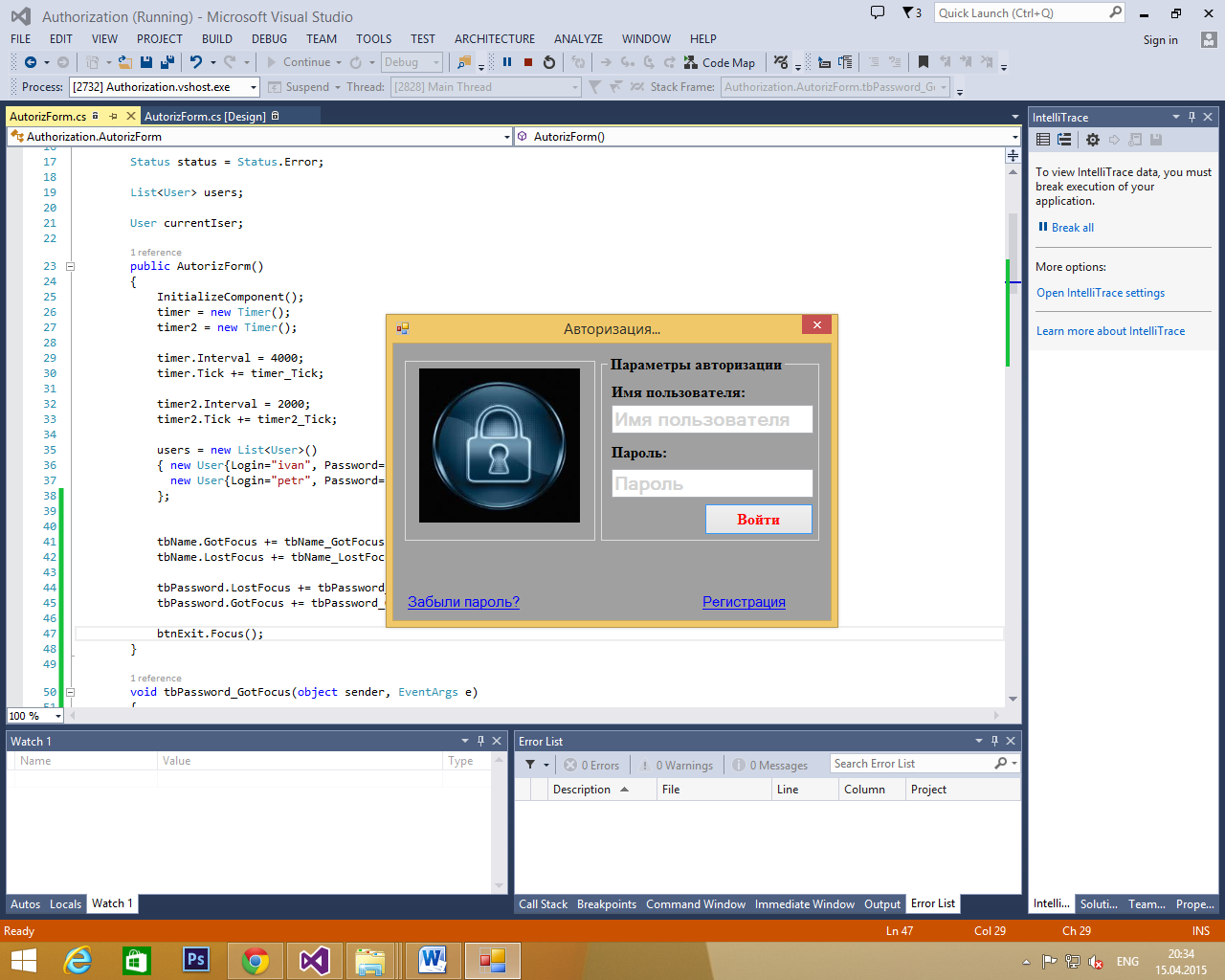
**- MS SQL, имя базы данных «registration»**

- логин - должен быть уникальным в системе;

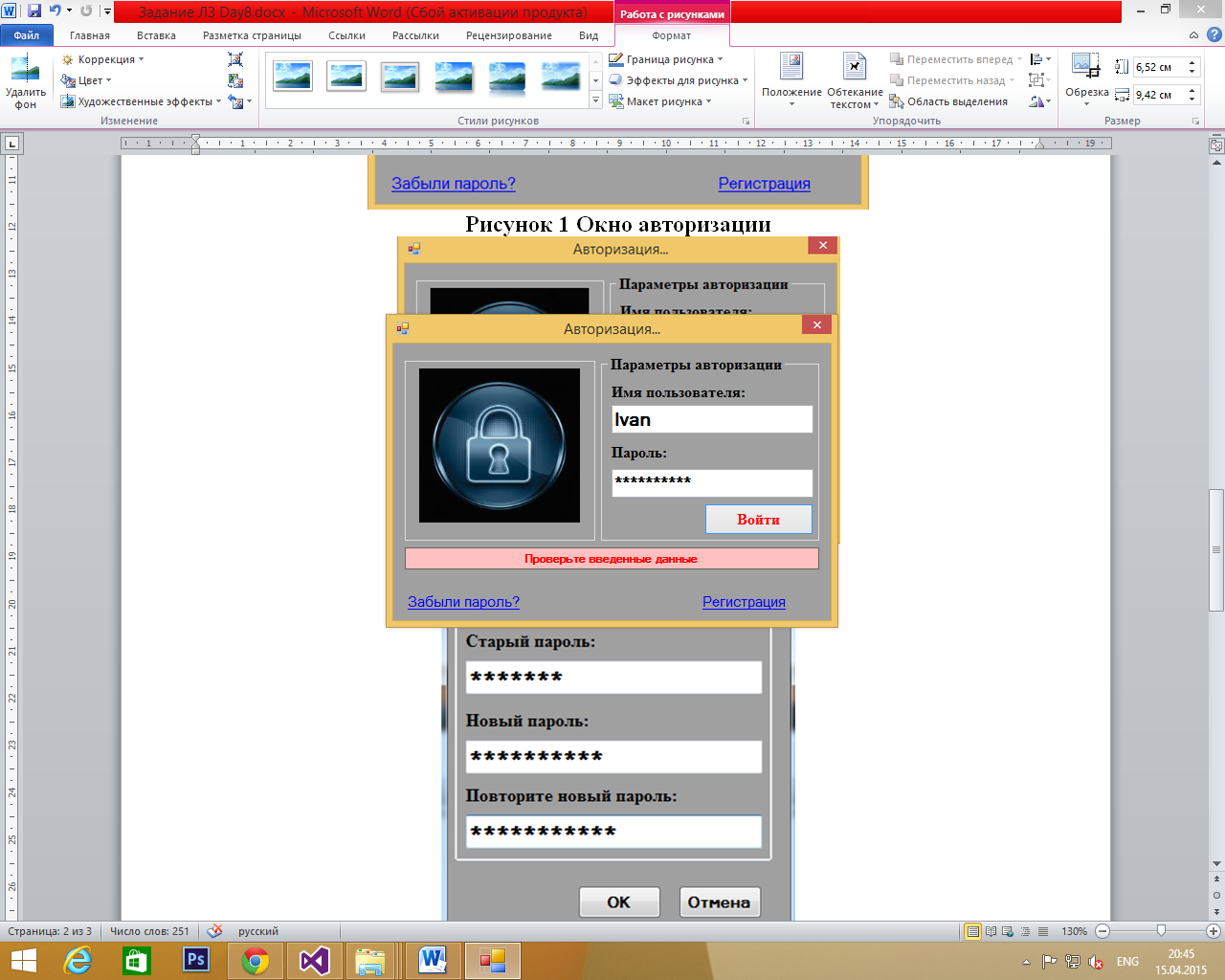
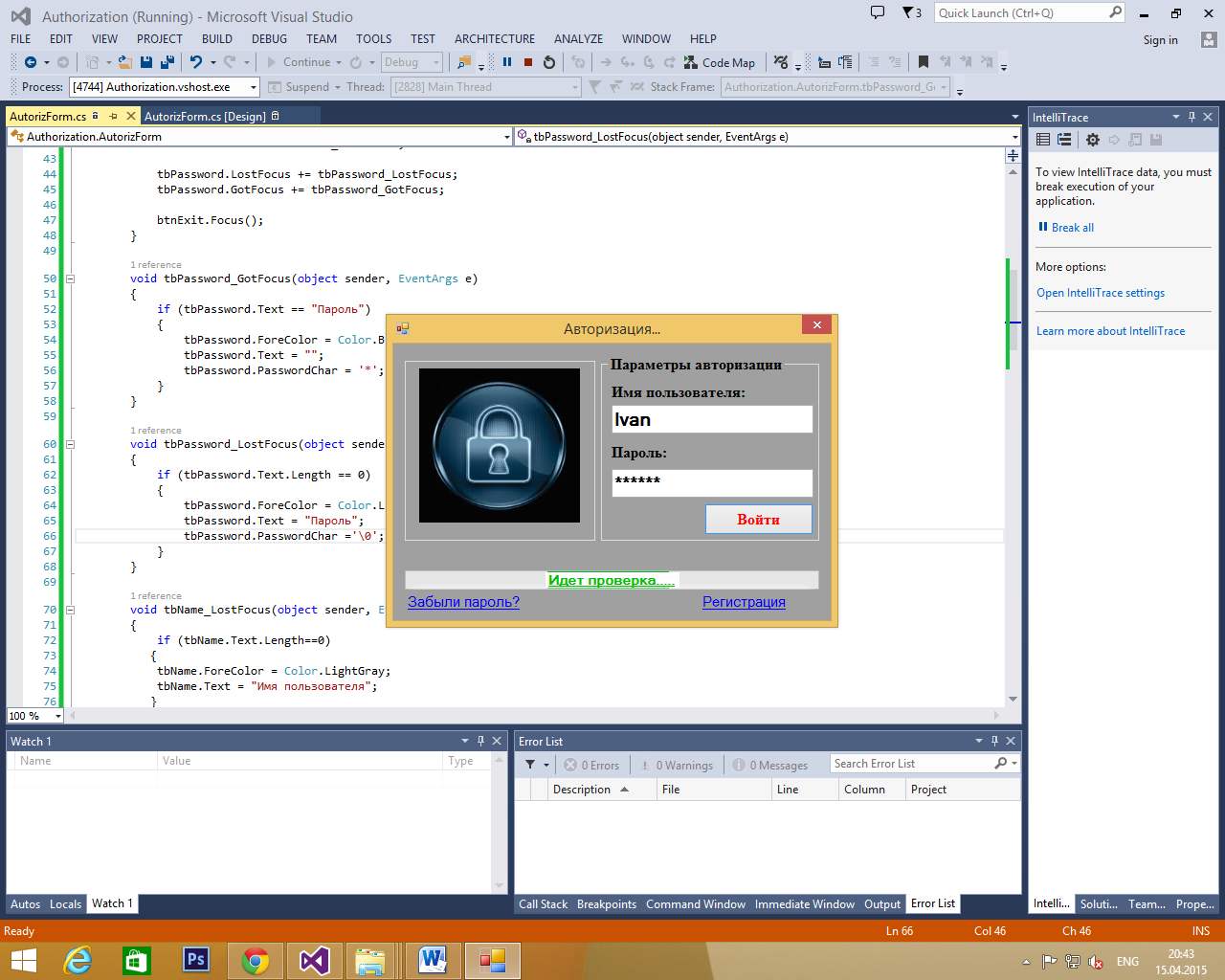
- пароль – не менее 6 символов;

- личные данные (фамилия, имя, e-mail - обязательно).

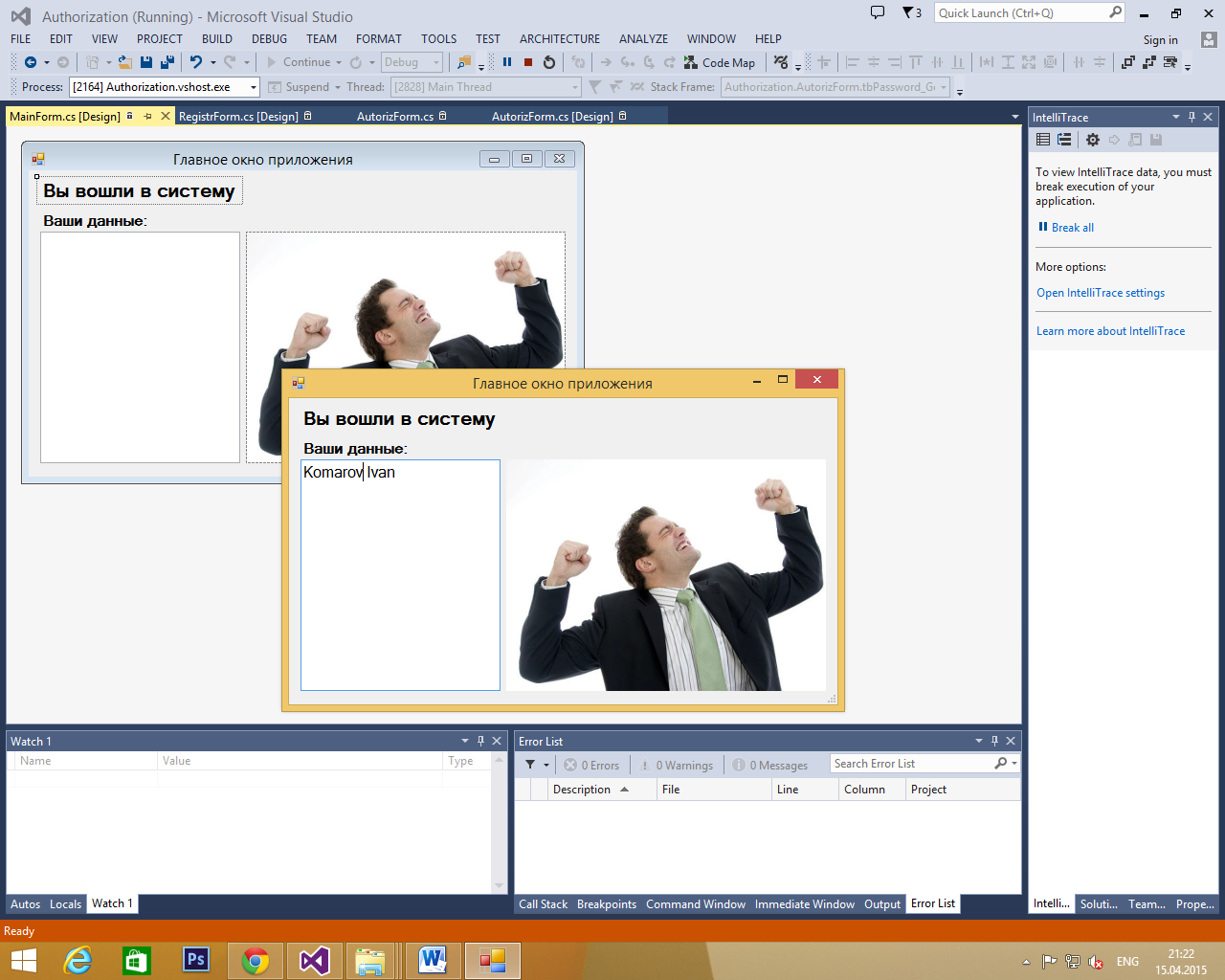
См. дополнительные требования.



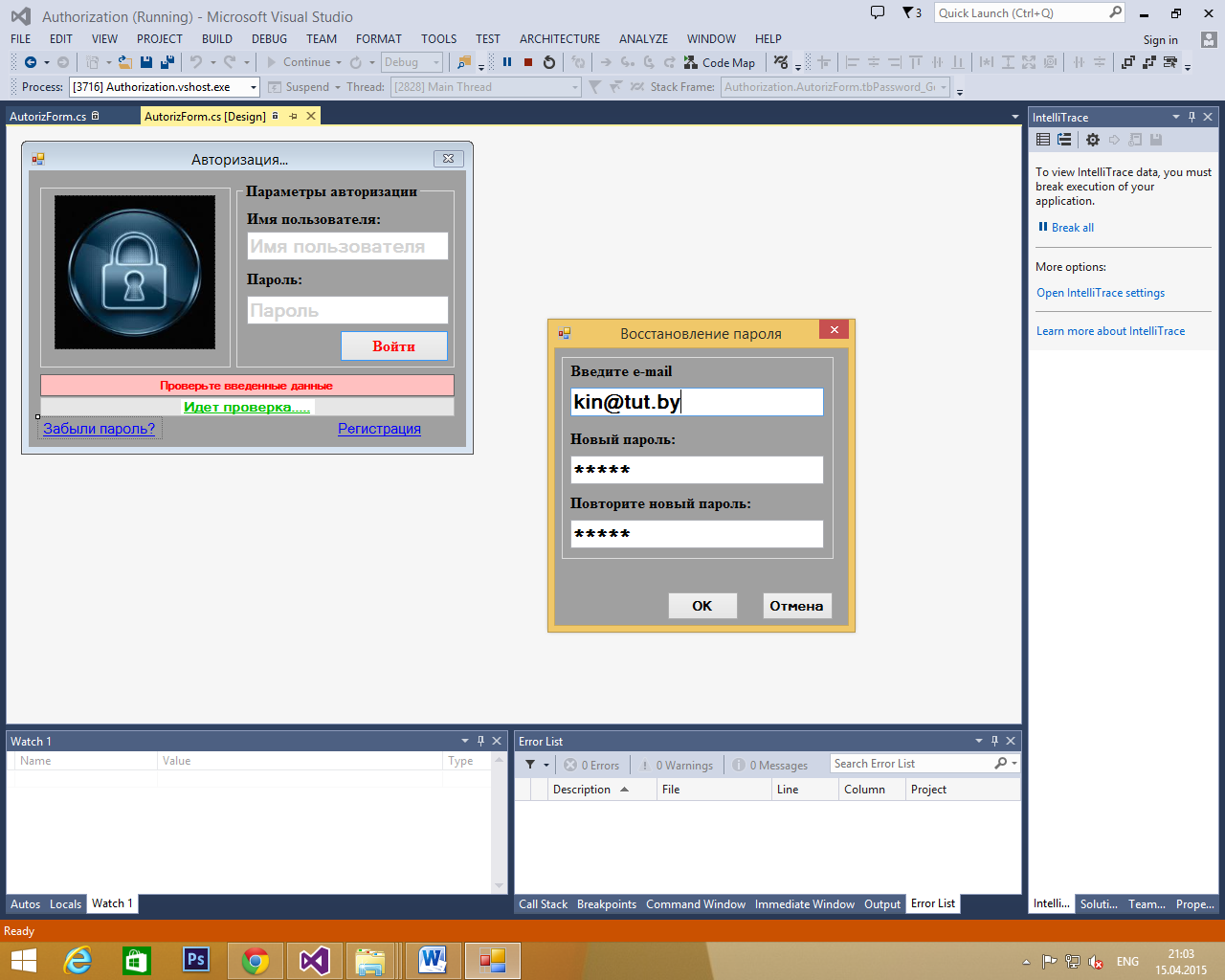
**Рисунок 1 Окно авторизации**



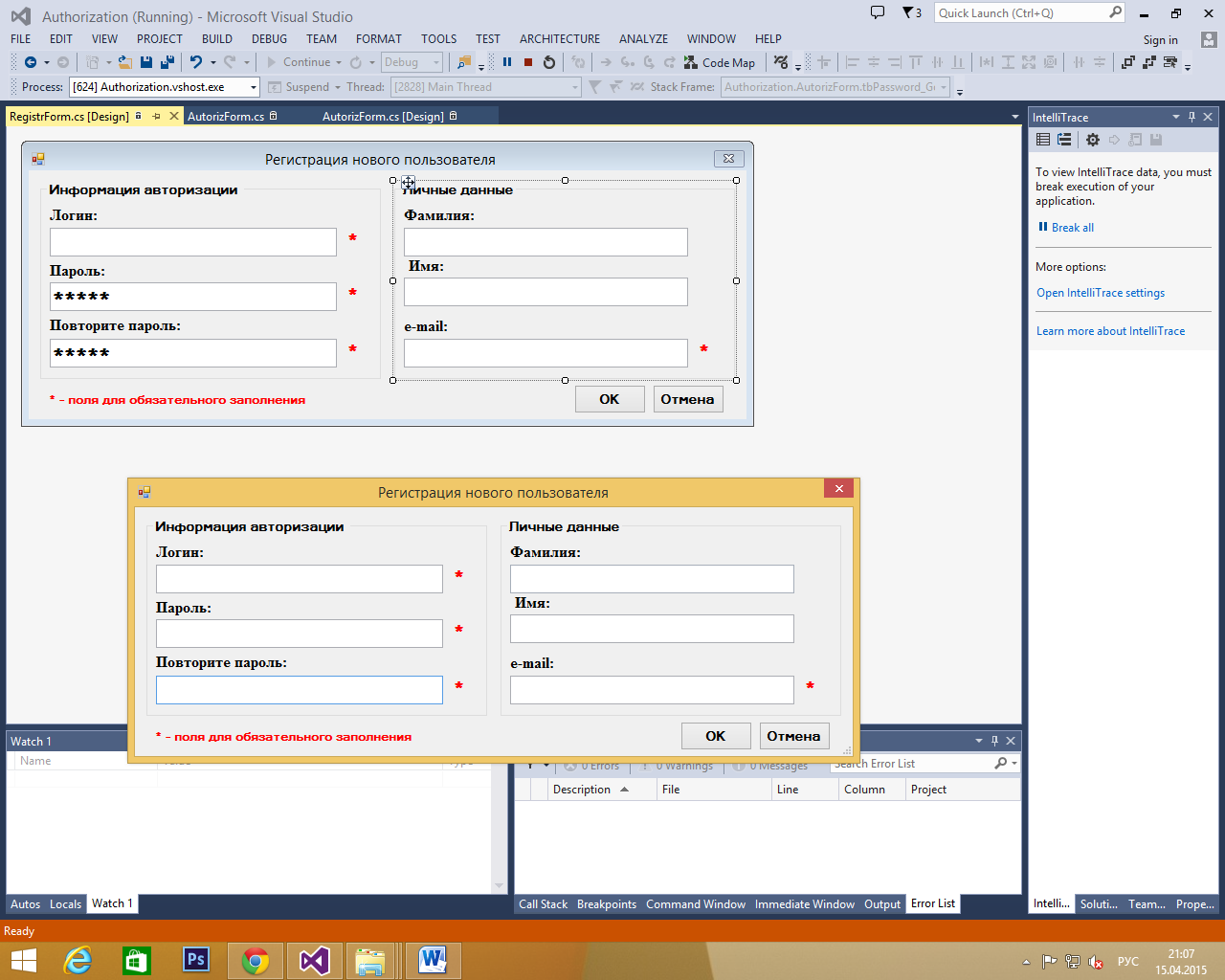
**Рисунок 2 Проверка данных Рисунок 3 Неверные данные**



**Рисунок 4 Проверка данных**



**Рисунок 5**



**Рисунок 6**

**Дополнительные задания:**

- установить Timeout подключения к базе - 5сек.

- настроить подключение к базе по логину и паролю (SQL server Authentification)

- функциональность проверки наличия в системе пользователя с таким же именем реализовать в виде хранимой процедуры;

- придумать и реализовать использование транзакции.

**Приложение 1 Операторы SQl для проекта «Авторизация»:**

**Создание таблицы:**

**CREATE TABLE [dbo].[Users] (**

**[Id] BIGINT IDENTITY (1, 1) NOT NULL,**

**[Name] NCHAR (20) NOT NULL,**

**[Password] NCHAR (15) NOT NULL,**

**[Email] NCHAR (20) NOT NULL,**

**PRIMARY KEY CLUSTERED ([Id] ASC)**

**);**

**CREATE TABLE [dbo].[UsersInfo] (**

**[Id] BIGINT IDENTITY (1, 1) NOT NULL,**

**[LastName] NCHAR (20) NOT NULL,**

**[FirstName] NCHAR (50) NOT NULL,**

**[Adres] NCHAR (150) NULL,**

**[Phone] NCHAR (40) NULL,**

**[Code] INT NOT NULL,**

**PRIMARY KEY CLUSTERED ([Id] ASC)**

**);**

**INSERT INTO [Users] ([Name], [Password], [Email]) VALUES ('Fox','1111',' Fox @mail.by');**

**INSERT INTO [Users] ([Name], [Password], [Email]) VALUES ('SunKot','852632','Sun@mail.by');**

**INSERT INTO [Users] ([Name], [Password], [Email]) VALUES ('Rulon','Run456','rulon@mail.by');**

**INSERT INTO [Users] ([Name], [Password], [Email]) VALUES ('Loput7','qwerty','Loput@mail.by');**

**INSERT INTO [Users] ([Name], [Password], [Email]) VALUES ('Rewa','asd44',' Rewa @mail.by');**

**INSERT INTO [UsersInfo] (LastName, FirstName, Adres,Phone, Code) VALUES ('Liza','Ivanova','Grodno','+37529741235',123565);**

**INSERT INTO [UsersInfo] (LastName, FirstName, Adres,Phone, Code) VALUES ('Alex','Smirnov','Brest','',34532423);**

**INSERT INTO [UsersInfo] (LastName, FirstName, Adres,Phone, Code) VALUES ('Vova','Sidorov','','+37529865421',356321);**

**INSERT INTO [UsersInfo] (LastName, FirstName, Adres,Phone, Code) VALUES ('Inna','Krot','Minsk','',56312365);**

**INSERT INTO [UsersInfo] (LastName, FirstName, Adres,Phone, Code) VALUES ('Nikolaj','Guk','Minsk','',7865333);**

**INSERT INTO [UsersInfo] (LastName, FirstName, Adres,Phone, Code) VALUES ('Ivan','Gromov','Minsk','',457445);**