Разработка библиотеки классов

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

namespace CAPTCHA\_lib

{

public class Class1

{

static public int ClimateTechModelID\_SQL(string name, string modelEq)

{

try

{

SqlConnection MyConnection = new SqlConnection(name);

MyConnection.Open();

string ComDel = $"select ClimateTechModelID from ClimateTechModel where ClimateTechModel=N\'{modelEq}\'";

SqlCommand cmd1 = new SqlCommand(ComDel, MyConnection);

int modelID = Convert.ToInt32(cmd1.ExecuteScalar().ToString());

MyConnection.Close();

return modelID;

}

catch (Exception ex) { return 0; }

}

static public int ProblemDescryptionID\_SQL(string name, string problem)

{

try

{

SqlConnection MyConnection = new SqlConnection(name);

MyConnection.Open();

string ComDel1 = $"select ProblemDescryptionID from ProblemDescryption where ProblemDescryption=N\'{problem}\'";

SqlCommand cmd2 = new SqlCommand(ComDel1, MyConnection);

int problemlID = Convert.ToInt32(cmd2.ExecuteScalar().ToString());

MyConnection.Close();

return problemlID;

}

catch (Exception ex) { return 0; }

}

static public void ADD\_SQL(string name, string modelEq, string problem, int requestsID)

{

try

{

SqlConnection MyConnection = new SqlConnection(name);

MyConnection.Open();

string ComDel2 = $"Insert into Requests ([StartDate],[ClimateTechModelID],[ProblemDescryptionID], [RequestStatusID],[clientID]) values " +

$"(N\'{DateTime.Today}\',N\'{ClimateTechModelID\_SQL(name,modelEq)}\',N\'{ProblemDescryptionID\_SQL(name, problem)}\',N\'{3}\',N\'{requestsID}\')";

SqlCommand cmd3 = new SqlCommand(ComDel2, MyConnection);

cmd3.ExecuteNonQuery();

MyConnection.Close();

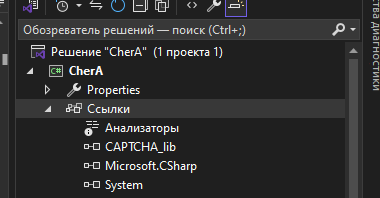
}

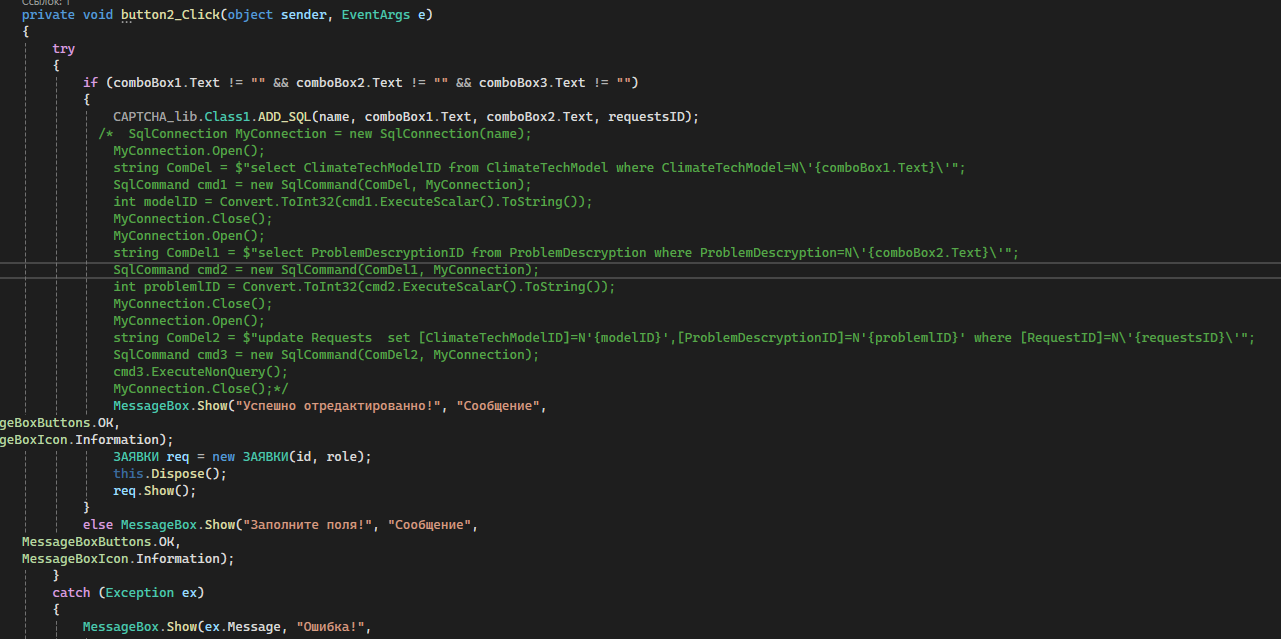
catch (Exception ex) { /\* ignore \*/ }

}

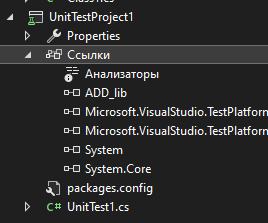
}

}





Тестирование приложения



Аннотация теста:

|  |  |
| --- | --- |
| **Название проекта** | CherA |
| **Рабочая версия** | 1.0 |
| **Имя тестирующего** | Червонцева Анастасия Андреевна |
| **Дата(ы) теста** | 05.11.2024 |

Тестовый пример #1:

|  |  |
| --- | --- |
| **Тестовый пример #** | TC \_1 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Добавление новой заявки |
| **Краткое изложение теста** | Необходимо зайти в приложение и ввести логин и пароль заказчика, после чего зайти в меню заказчика. После этого нужно зайти на форму ЗАЯВКИ, а после перейти на форму добавления заявки. Заполнив все поля, нужно добавить заявку, а затем проверить наличие добавленной заявки в таблице. Проверить наличие добавленной заявки |
| **Этапы теста** | 1. Зайти в заявки 2. Зайти в добавление заявок 3. Заполнить поля 4. Добавить заявку 5. Проверить наличие добавленной заявки |
| **Тестовые данные** | Тип оборудования кондиционер, название модели оборудования Ballu BAHD-1250, проблема Не работает |
| **Ожидаемый результат** | Сообщение «Заявка добавлена» |
| **Фактический результат** | Сообщение «Заявка добавлена» |
| **Статус** | Зачет |
| **Предварительное условие** | войти в систему |
| **Постусловие** | С добавленной заявкой могут работать другие пользователи |
| **Примечания/комментарии** |  |

Тестовый пример #2:

|  |  |
| --- | --- |
| **Тестовый пример #** | TC \_2 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Создание неверной заявки |
| **Краткое изложение теста** | Необходимо зайти в приложение и ввести логин и пароль заказчика, после чего зайти в меню заказчика. После этого нужно зайти на форму ЗАЯВКИ, а после перейти на форму добавления заявки. Заполнив только модель оборудования. Попытаться создать заявку |
| **Этапы теста** | 1. Зайти в заявки 2. Зайти в добавление заявок 3. Заполнить не все поля 4. Добавить заявку |
| **Тестовые данные** | Тип оборудования кондиционер, название модели « », проблема «» |
| **Ожидаемый результат** | Сообщение «Заполните поля!» |
| **Фактический результат** | Сообщение «Заполните поля!» |
| **Статус** | Зачет |
| **Предварительное условие** | войти в систему |
| **Постусловие** | Заявка не добавлена |
| **Примечания/комментарии** |  |

Тестовый пример #3:

|  |  |
| --- | --- |
| **Тестовый пример #** | TC \_3 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Успешный вход в систему |
| **Краткое изложение теста** | Необходимо зайти в приложение и ввести логин и пароль заказчика, после чего зайти в меню заказчика. |
| **Этапы теста** | 1. Войти в приложение 2. Ввести логин и пароль заказчика 3. Войти в меню заказчика |
| **Тестовые данные** | Имя пользователя login8, пароль пользователя pass8 |
| **Ожидаемый результат** | Скрытие меню авторизации и открытие главного меню |
| **Фактический результат** | Скрытие меню авторизации и открытие главного меню |
| **Статус** | Зачет |
| **Предварительное условие** | Наличие пользователя (заказчика) в системе |
| **Постусловие** | Открытие меню заказчика |
| **Примечания/комментарии** |  |

Тестовый пример #4:

|  |  |
| --- | --- |
| **Тестовый пример #** | TC \_4 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Редактирование заявки оператором |
| **Краткое изложение теста** | Необходимо зайти в приложение и ввести логин и пароль оператора, после чего зайти в меню оператора. После этого нужно зайти на форму ЗАЯВКИ, а после перейти на форму редактирования заявки. Заполнив все поля, нужно добавить заявку. Проверить наличие отредактированных данных в заявке |
| **Этапы теста** | 1. Зайти в заявки 2. Зайти в редактирование заявок 3. Заполнить поля 4. Добавить заявку |
| **Тестовые данные** | Номер заявки 4, мастер Гончарова |
| **Ожидаемый результат** | Сообщение: «Успешно отредактировано!» |
| **Фактический результат** | Сообщение: «Успешно отредактировано!» |
| **Статус** | Зачет |
| **Предварительное условие** | войти в систему под операторм |
| **Постусловие** | изменение доступно всем пользователям |
| **Примечания/комментарии** |  |

Тестовый пример #5:

|  |  |
| --- | --- |
| **Тестовый пример #** | TC \_5 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Неверное редактирование заявки оператором |
| **Краткое изложение теста** | Необходимо зайти в приложение и ввести логин и пароль оператора, после чего зайти в меню оператора. После этого нужно зайти на форму ЗАЯВКИ, а после перейти на форму редактирования заявки. НЕ заполнять поля редактирования, попробовать добавить заявку. |
| **Этапы теста** | 1. Зайти в заявки 2. Зайти в редактирование заявок 3. Добавить заявку |
| **Тестовые данные** | Номер заявки 4, мастер « » |
| **Ожидаемый результат** | Сообщение: «Заполните поля!» |
| **Фактический результат** | Сообщение: «Заполните поля!» |
| **Статус** | Зачет |
| **Предварительное условие** | войти в систему под оператором |
| **Постусловие** | Заявка не отредактирована |
| **Примечания/комментарии** |  |

**Модульные тесты**

using Microsoft.VisualStudio.TestTools.UnitTesting;

using System;

using System.Data.SqlClient;

namespace UnitTestProject1

{

[TestClass]

public class UnitTest1

{

private string name = @"Data Source=ADCLG1; Initial catalog=CherA;Integrated Security=True";

[TestMethod]

public void ClimateTechModelID\_SQL\_5()

{

object result = CAPTCHA\_lib.Class1.ClimateTechModelID\_SQL(name, "Ballu BAHD-1250");

Assert.AreEqual(result.ToString(), "5");

}

[TestMethod]

public void ClimateTechModelID\_SQL\_error()

{

object result = CAPTCHA\_lib.Class1.ClimateTechModelID\_SQL(name, "dfsdfsdfsdfsd");

Assert.AreEqual(result.ToString(), "0");

}

[TestMethod]

public void ProblemDescryptionID\_SQL\_5()

{

object result = CAPTCHA\_lib.Class1.ProblemDescryptionID\_SQL(name, "Не работает");

Assert.AreEqual(result.ToString(), "5");

}

[TestMethod]

public void ProblemDescryptionID\_SQL\_error()

{

object result = CAPTCHA\_lib.Class1.ProblemDescryptionID\_SQL(name, "dfsdfsdfsdfsd");

Assert.AreEqual(result.ToString(), "0");

}

[TestMethod]

public void Add\_Columns()

{

SqlConnection MyConnection = new SqlConnection(name);

MyConnection.Open();

string ComDel = $"select count(\*) from Requests";

SqlCommand cmd1 = new SqlCommand(ComDel, MyConnection);

int result = Convert.ToInt32(cmd1.ExecuteScalar().ToString());

MyConnection.Close();

CAPTCHA\_lib.Class1.ADD\_SQL(name, "Ballu BAHD-1250", "Не работает", 8);

MyConnection.Open();

ComDel = $"select count(\*) from Requests";

cmd1 = new SqlCommand(ComDel, MyConnection);

int result1 = Convert.ToInt32(cmd1.ExecuteScalar().ToString());

MyConnection.Close();

Console.WriteLine(result);

Console.WriteLine(result1);

Assert.AreEqual(result1, result+1);

}

}

}

