|  |
| --- |
| Изображение выглядит как текст, внешний, арка  Автоматически созданное описание**МИНОБРНАУКИ РОССИИ**  федеральное государственное бюджетное образовательное  учреждение высшего образования  **«Национальный исследовательский университет «МЭИ»** |

|  |  |
| --- | --- |
| **Институт** | ИВТИ |
| **Кафедра** | ПМИИ |

|  |  |
| --- | --- |
|  |  |
| **Курсовая работа по дисциплине: «Программная инженерия»**  **на тему:** **«Разработка развивающего приложения для детей начальных классов».**  **Выполнил: студент группы А-13а-19**  **Щучкин Н. Ю.**  **Преподаватель: Маран М. М.**  **Руководитель НИР: Бартеньев О. В.**  **Москва, 2021** | |

Оглавление

[**Введение** 3](#_Toc91163598)

[**Анализ задания** 5](#_Toc91163599)

[**Проектирование** 5](#_Toc91163600)

[**Описание реализации** 6](#_Toc91163601)

[**Тестирование** 15](#_Toc91163602)

[Тестирование системы регистрации и входа 15](#_Toc91163603)

[Тестирование отображения результатов ученика на главной форме 19](#_Toc91163604)

[**Заключение** 22](#_Toc91163605)

[**Список источников** 22](#_Toc91163606)

# **Введение**

Наша жизнь очень сильно изменилась с тех пор, как в ней появился Covid-19. Пандемия коронавируса разделила жизнь на «до» и «после». Многие из-за противовирусных ограничений были вынуждены перейти на дистанционный формат работы или обучения. Считаю, что на взрослых людей это повлияло не так сильно, как на детей и подростков. Взрослый человек сам решает и делает выбор, что ему нужно и важно. И если человек правильно спланирует время и найдет правильную мотивацию, то он сможет работать из дома не хуже, чем в офисе. То же самое относится и к студентам. Если студенту важно получить знания, то он может получит их в нужном объеме и на дистанционном обучении, все зависит от него самого.

Со школьниками совсем другая история, особенно в начальных классах. Учителя начальных классов должны давать детям не только знания, но и быть их наставниками, готовить своих учеников к взрослой жизни и всячески пытаться помочь им с проблемами (не только проблемами успеваемости). Очевидно, что при личном общении учителя с учеником это делать гораздо проще.

Ещё одна проблема заключается в том, что на дистанционном обучении школьники могут начать расслабляться. Во время очных уроков учитель видит всех своих учеников и может сделать замечание тем, кто отвлекается или ничего не делает. Во время дистанционных занятий (если они проходят без видеокамер) учитель не может пристально следить за учениками, а они в свою очередь могут этим воспользоваться (заниматься во время уроков совершенно другими делами). Они ещё не до конца понимают, насколько им важна учёба. А родителям сложно их контролировать, потому что они постоянно находятся на работе.

В такой ситуации учитель должен организовать свою работу во время дистанционного обучения так, чтобы он смог заинтересовать своих учеников, привлечь их внимание и направить их активность в нужное русло.

Для этого можно использовать разные способы. Один из таких способов – создание единой платформы, которая поможет, как и учителю отслеживать успеваемость своих учеников и быть в постоянном контакте с ними, так и сможет привлечь внимание детей и заинтересовать их.

Цель моей курсовой работы – изучить способы, при помощи которых можно было бы реализовать такую платформу и создать прототип. Учителя в данной платформе смогут оценивать результаты своих учеников, а ученики смогут больше заинтересоваться учебой из-за интерактивного формата платформы.

# **Анализ задания**

Для достижения поставленной цели необходимо изучить инструменты, которые могут помочь в создании такой комплексной платформы-приложения. Мой выбор пал на технологию Windows Forms и, соответственно, язык программирования C#. В Windows Forms есть все необходимые элементы: многооконные приложения, текстовые поля, поля ввода, кнопки и т. д.

# **Проектирование**

Приложение должно содержать в себе:

1. Окно регистрации.
2. Окно входа.
3. Главное окно, на котором будут представлены результаты прохождения учеником определенных заданий.
4. Отдельные окна для каждого предмета (окно для заданий по русскому языку, окно для заданий по математике, и т. д.)
5. При входе в приложение в статусе “учитель” на главном окне учителя должны отображаться результаты всех его учеников.

# **Описание реализации**

Для начала нужно реализовать систему регистрации новых пользователей и входа в систему. Реализовать эту систему можно при помощи базы данных.

База данных будет содержать следующие поля:

1. Имя пользователя.
2. Пароль пользователя.
3. Класс пользователя.
4. Идентификатор, является ли пользователь учителем.
5. Количество выполненных пользователем заданий по русскому языку.
6. Количество выполненных пользователем заданий по математике.
7. Количество выполненных пользователем заданий по окружающему миру.
8. Максимальное количество шагов, которое потребовалось пользователю для прохождения игры “Пятнашки”.

При регистрации пользователь вводит свой логин, пароль, класс и отмечает галочкой, является ли он учителем. Далее пользователь нажимает кнопку “Регистрация” и после этого в базу данных добавляется новый пользователь. Поля нового пользователя заполняются по следующему принципу: поле “Имя пользователя” заполняется значением текстового поля, в которое пользователь ввел свой логин, “Пароль пользователя” заполняется значением текстового поля, в которое пользователь ввел свой пароль, “Класс пользователя” заполняется значением текстового поля, в которое пользователь ввел свой класс, если во время регистрации пользователь не выбрал пункт “Зарегистрироваться как учитель”, то поле “Идентификатор” заполняется значением 0, в противном случае – значением 1, остальные поля базы данных заполняются значением 0.

При входе в систему пользователь должен ввести свой класс, логин и пароль. Затем пользователь нажимает кнопку “Войти”. В этот момент базе данных отправляется запрос, существует ли пользователь с введенными данными. Если такой пользователь существует, то происходит вход в систему.  
  
  
  
  
Кот, отвечающий за регистрацию пользователя в системе:  
  
private void button2\_Click(object sender, EventArgs e) // Событие нажатия на кнопку "Регистрация"

{

// Если при нажатии на кнопку какие-то поля остались пустыми, выводим сообщение об ошибке

if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" || textBox4.Text == "")

{

MessageBox.Show("Все поля должны быть заполнены", "Ошибка во время регистрации", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

// Если все поля заполнены и два поля с вводом пароля имеют одинаковые значения:

else if (textBox2.Text == textBox3.Text)

{

// Добавляем нового пользователя в базу данных

con.Open();

if (checkBox1.Checked)

{

string register = "INSERT INTO tbl\_users VALUES('" + textBox1.Text + "','" + textBox2.Text + "', 0, 0, 999, 0, '" + textBox4.Text + "', 1)";

cmd = new OleDbCommand(register, con);

cmd.ExecuteNonQuery();

}

else

{

string register = "INSERT INTO tbl\_users VALUES('" + textBox1.Text + "','" + textBox2.Text + "', 0, 0, 999, 0, '" + textBox4.Text + "', 0)";

cmd = new OleDbCommand(register, con);

cmd.ExecuteNonQuery();

}

con.Close();

MessageBox.Show("Вы успешно зарегистрировались", "Регистрация", MessageBoxButtons.OK, MessageBoxIcon.Information);

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

textBox4.Text = "";

}

else

{

MessageBox.Show("Пароли не совпадают", "Ошибка во время регистрация", MessageBoxButtons.OK, MessageBoxIcon.Error);

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

textBox4.Text = "";

}

}  
  
  
  
  
  
Код, отвечающий за вход пользователя в систему:  
  
private void button2\_Click(object sender, EventArgs e) // Событие нажатия на кнопку "Войти как ученик"

{

con.Open();

// Проверяем, существует ли пользователь с введенными данными в базе данных

string login = "SELECT \* FROM tbl\_users WHERE username= '" + textBox1.Text + "' and password = '" + textBox2.Text + "' and class = '" + textBox3.Text + "' ";

cmd = new OleDbCommand(login, con);

OleDbDataReader dr = cmd.ExecuteReader();

if (dr.Read() == true) // Если нашелся пользователь с введенными данными

{

// скрытие этой формы и открытие главной формы

this.Hide();

f1.Enabled = true;

f1.WindowState = FormWindowState.Normal;

f1.label1.Text = textBox1.Text;

Form1.name = textBox1.Text;

// получаем из базы данных результаты данного пользователя и заносим их на главную фомру

string query = "SELECT russ FROM tbl\_users WHERE username = '" + textBox1.Text + "' ";

cmd = new OleDbCommand(query, con);

f1.label6.Text = cmd.ExecuteScalar().ToString() + "/10";

string query2 = "SELECT math FROM tbl\_users WHERE username = '" + textBox1.Text + "' ";

cmd = new OleDbCommand(query2, con);

f1.label5.Text = cmd.ExecuteScalar().ToString() + "/10";

string query3 = "SELECT game15 FROM tbl\_users WHERE username = '" + textBox1.Text + "' ";

cmd = new OleDbCommand(query3, con);

f1.label2.Text = cmd.ExecuteScalar().ToString();

string query4 = "SELECT nature FROM tbl\_users WHERE username = '" + textBox1.Text + "' ";

cmd = new OleDbCommand(query4, con);

f1.label9.Text = cmd.ExecuteScalar().ToString() + "/10";

}

else // Если такого пользователя не существует

{

MessageBox.Show("Неправильно введен логин и/или пароль", "Ошибка при входе", MessageBoxButtons.OK, MessageBoxIcon.Error);

textBox1.Text = "";

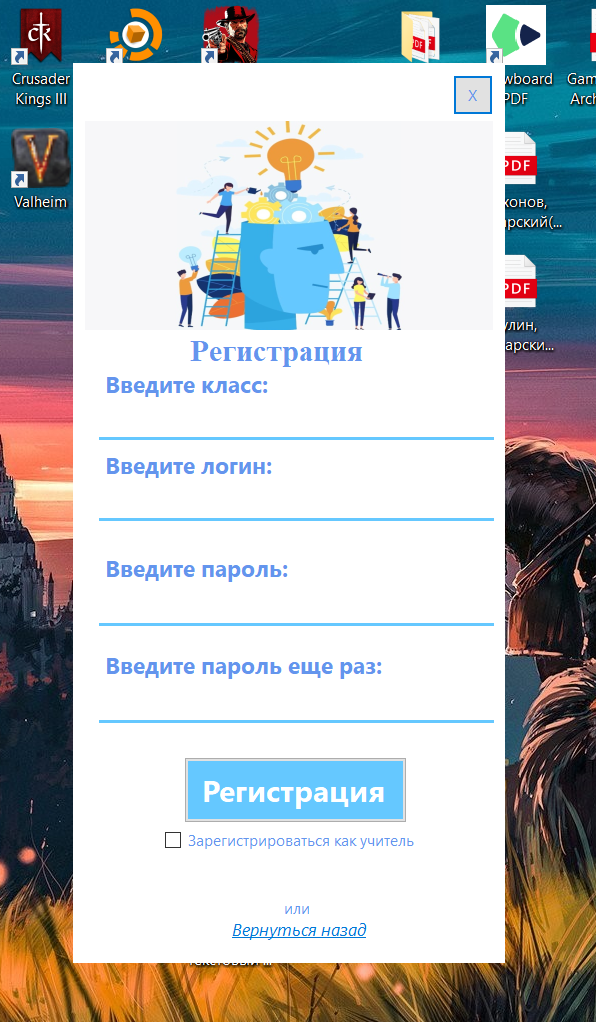
textBox2.Text = "";

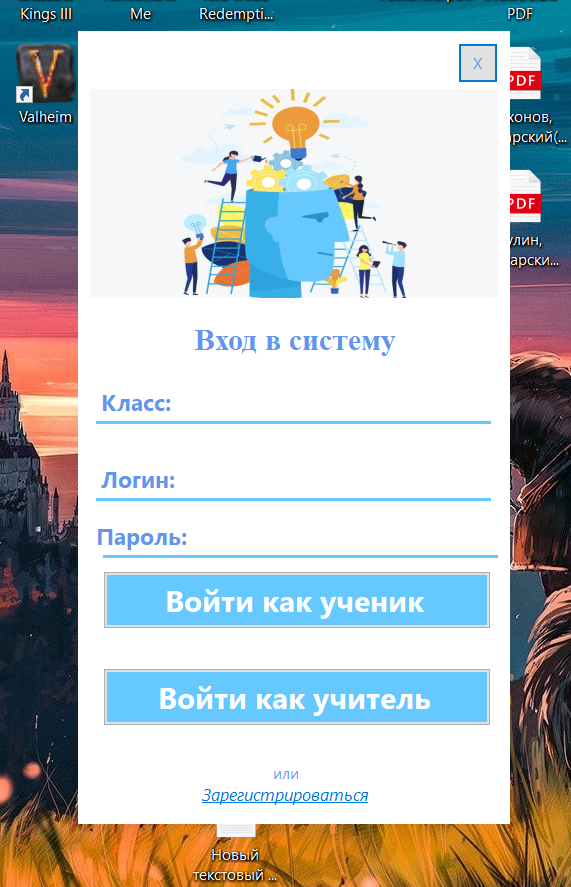
textBox3.Text = "";

}

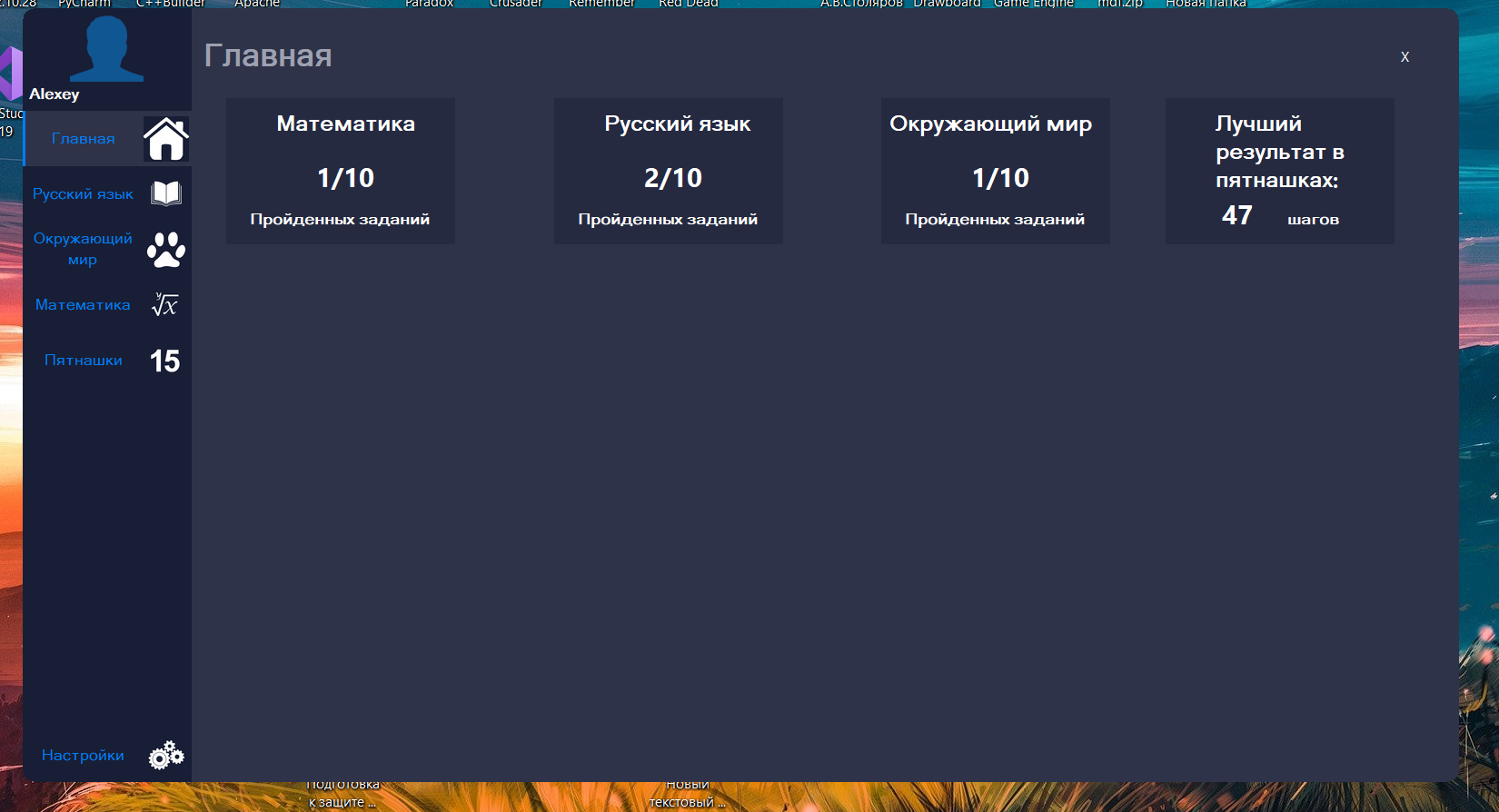
con.Close();

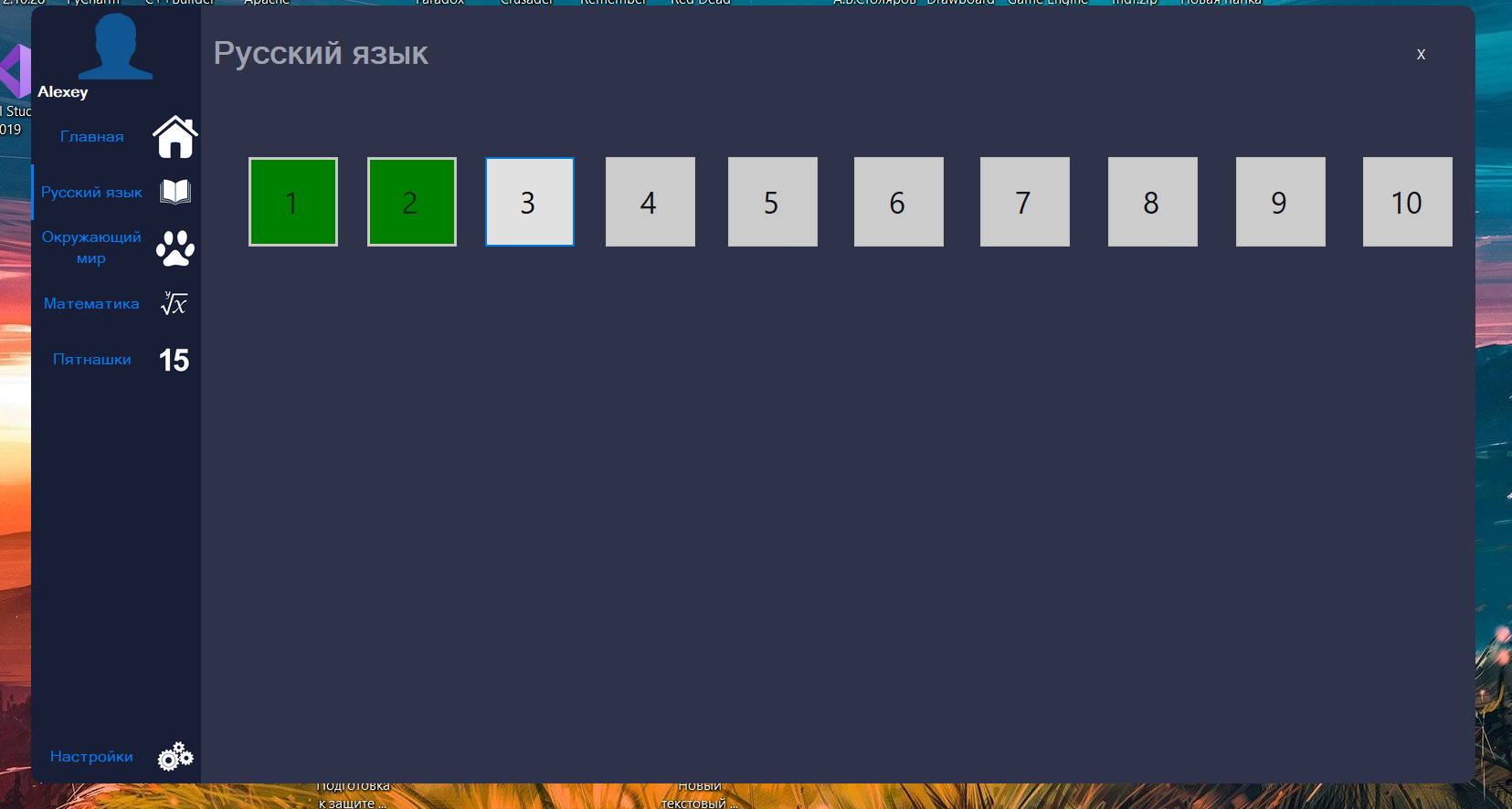
}

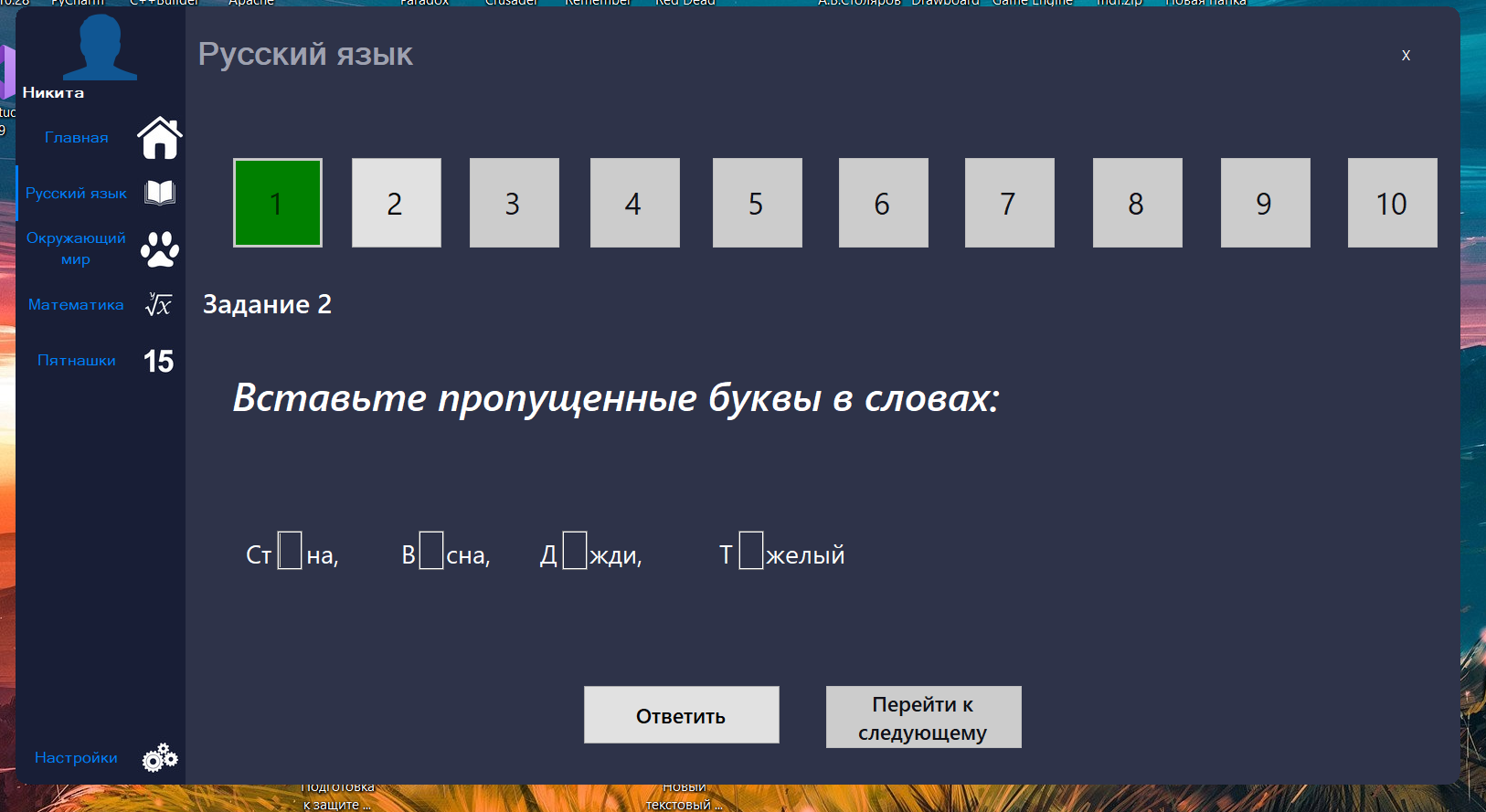
Окно регистрации выглядят следующим образом:  
  
  
Окно входа выглядят следующим образом:



После входа в систему, если пользователь является учеником, открывается главное окно пользователя, в котором ему предоставляется информация о пройденных им заданиях. В левой части главного окна находится меню.

Главное окно выглядит следующим образом:  


При нажатии на пункт меню, пользователь переходит на новое окно, соответствующее данному пункту, которое выглядит следующим образом:  
  
  
   
 Те задания, которые пользователь уже выполнил, отмечаются зеленым цветом. В данном случае (скриншот выше) пользователь выполнил два задания и ему стало доступно следующее - третье задание для выполнения. Задания становятся доступными постепенно, пока пользователь не выполнит задание, следующие будут недоступны. Как только пользователь выполнит задание с номером 3, ему станет доступно задание под номером 4 и т. д.

Второе задание по русскому языку, например, выглядит так:  
  
  
Код для анализа ответа пользователя на данное задание:  
  
private void button1\_Click(object sender, EventArgs e) // Событие нажатия на кнопку "Ответить"

{

// Проверяем, что ученик правильно вставил все буквы в слова

if (textBox1.Text == "е" && textBox2.Text == "е" && textBox3.Text == "о" && textBox4.Text == "я")

{

label10.Text = "Верно!";

button2.Enabled = true;

// Обновляем значение выполненных учеником заданий по русскому языку в БД

con.Open();

string query = "UPDATE tbl\_users SET russ = 2 WHERE username = '" + Form1.name + "' ";

OleDbCommand cmd = new OleDbCommand(query, con);

cmd.ExecuteNonQuery();

con.Close();

// Окрашиваем кнопку выполненного задания в зеленый цвет и делаем доступным следующее задание

R.button2.BackColor = Color.Green;

R.button3.Enabled = true;

// остальные задания остаются недоступными

buttons = new List<Button> { R.button1, R.button2, R.button3, R.button4, R.button5, R.button6, R.button7, R.button8, R.button9, R.button10 };

for (int i = 0; i < buttons.Count; i++)

{

if (i != 2)

{

dis\_able(buttons[i]);

}

}

}

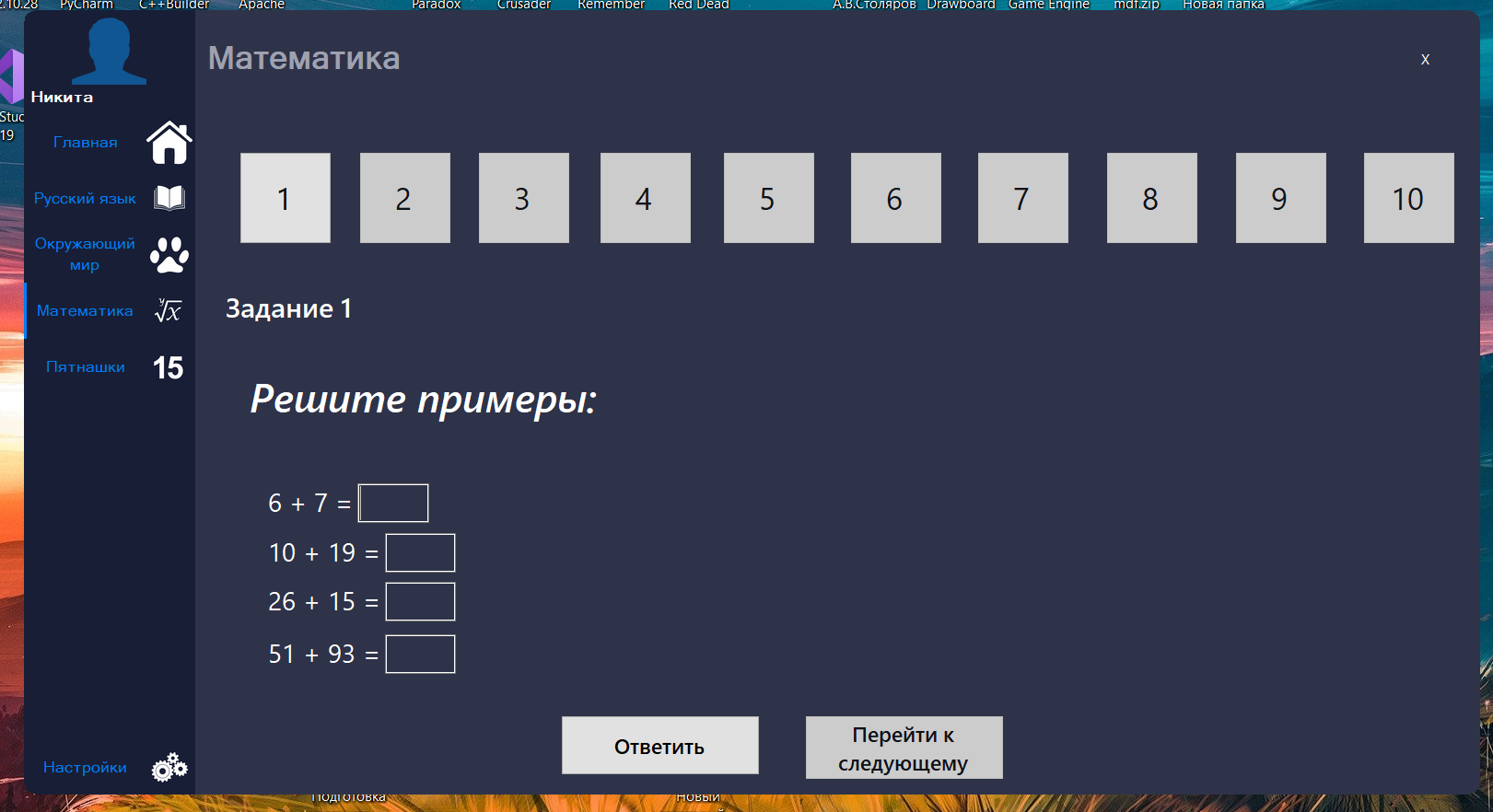
else

{

label10.Text = "Неверно, попробуйте ещё";

}

}

Первое задание по математике, например, выглядит так:  
  
  
Код для анализа ответа пользователя на данное задание:  
  
private void button1\_Click(object sender, EventArgs e) // Событие нажатие на кнопку "Ответить"

{

// Если ученик правильно ответил на все примеры

if (textBox1.Text == "13" && textBox2.Text == "29" && textBox3.Text == "41" && textBox4.Text == "144")

{

label10.Text = "Верно!";

button2.Enabled = true;

// Обновляем значение выполненных учеником заданий по математике в БД

con.Open();

string query = "UPDATE tbl\_users SET math = 1 WHERE username = '" + Form1.name + "' ";

cmd = new OleDbCommand(query, con);

cmd.ExecuteNonQuery();

con.Close();

// Окрашиваем кнопку выполненного задания в зеленый цвет и делаем доступным следующее задание

M.button1.BackColor = Color.Green;

M.button2.Enabled = true;

// остальные задания остаются недоступными

buttons = new List<Button> { M.button1, M.button2, M.button3, M.button4, M.button5, M.button6, M.button7, M.button8, M.button9, M.button10 };

for (int i = 0; i < buttons.Count; i++)

{

if (i != 1)

{

dis\_able(buttons[i]);

}

}

}

else

{

label10.Text = "Неверно, попробуйте ещё";

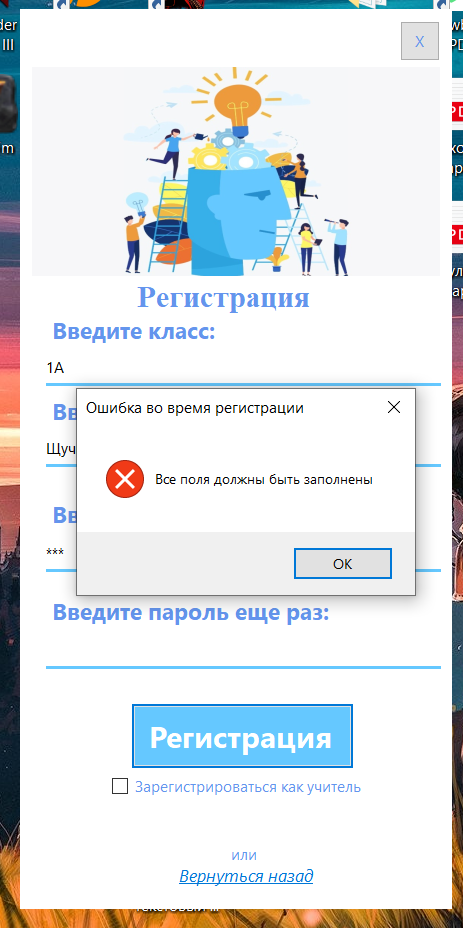
}

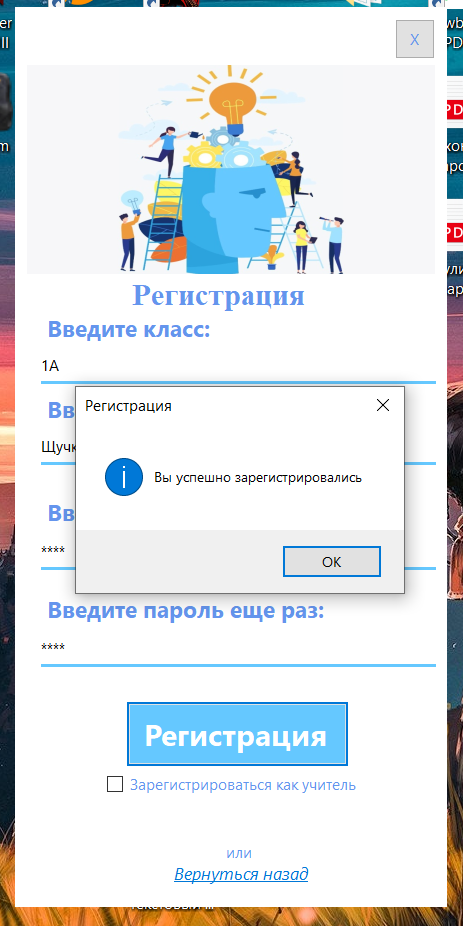
}  
  
Первое задание по окружающему миру, например, выглядит так:  
Изображение выглядит как текст, экран, другой

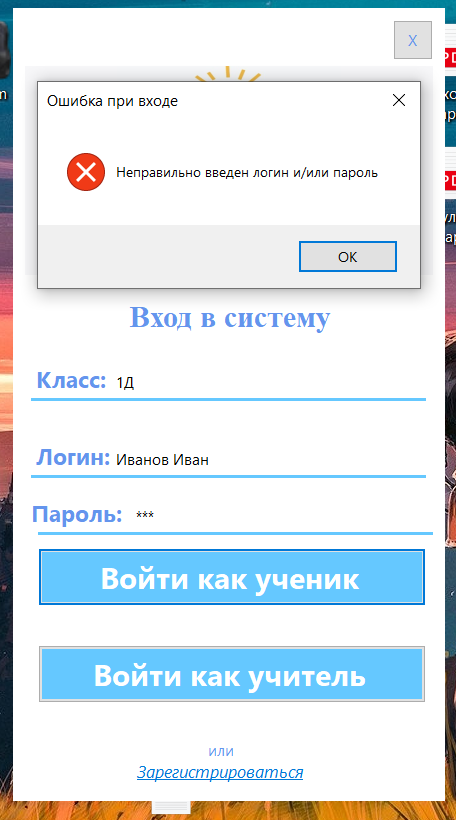
Автоматически созданное описание

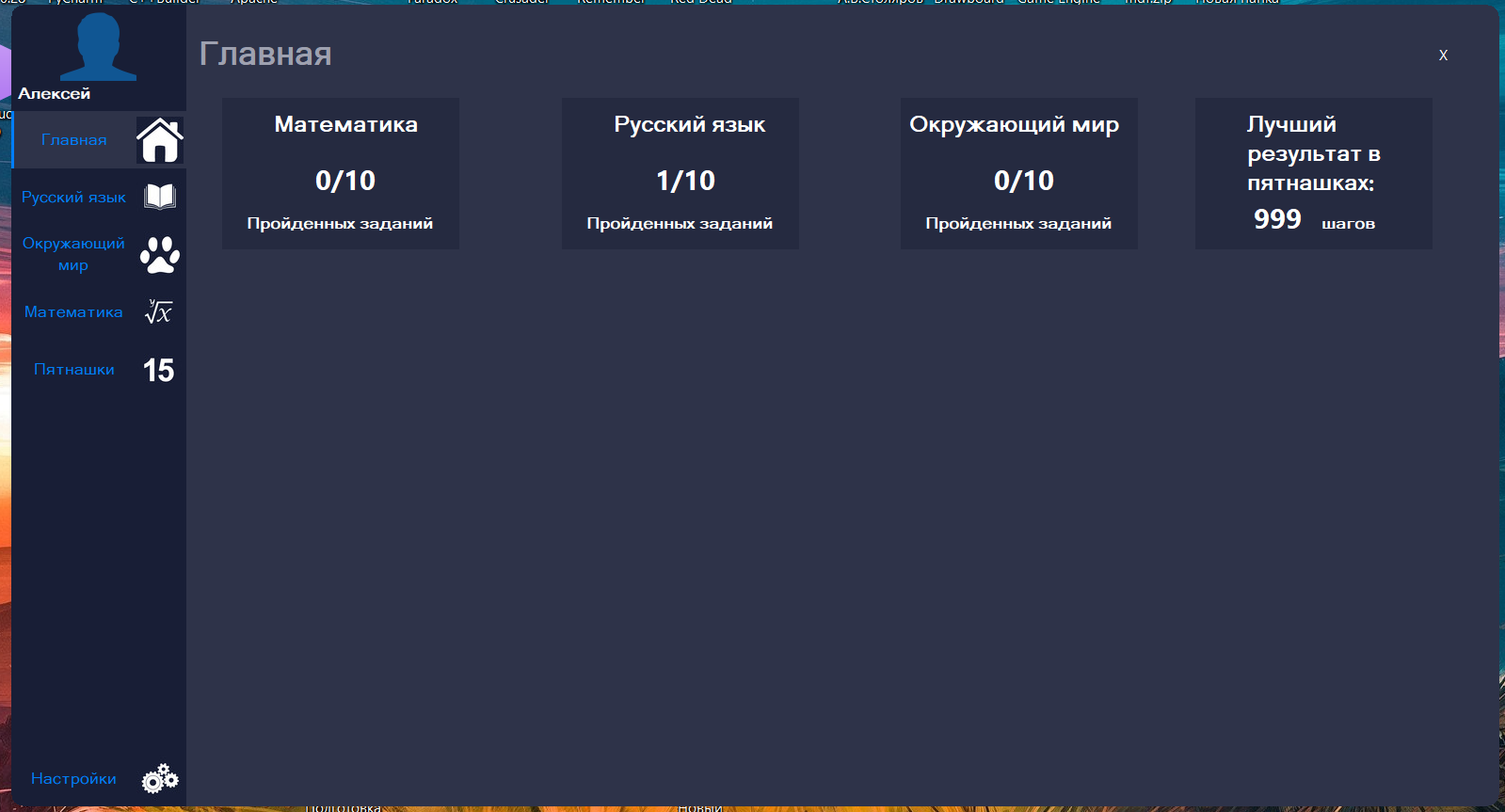
# **Тестирование**

## Тестирование системы регистрации и входа

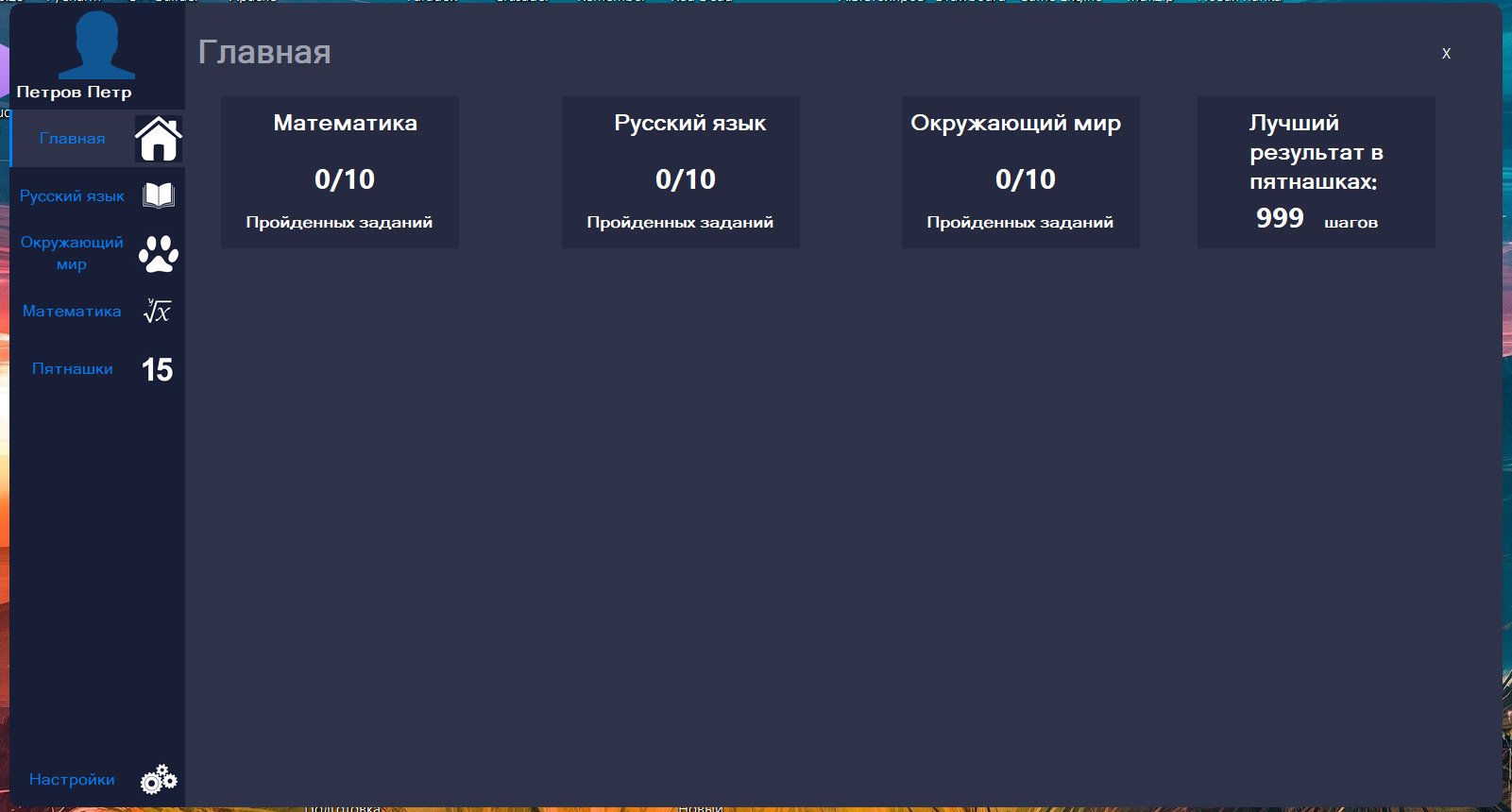
**Тест 1:**  
 Во время регистрации оставить какое-то поле пустым и нажать на кнопку “Регистрация”.  
**Результат теста 1:**  
 **Получаем сообщение об ошибке.**

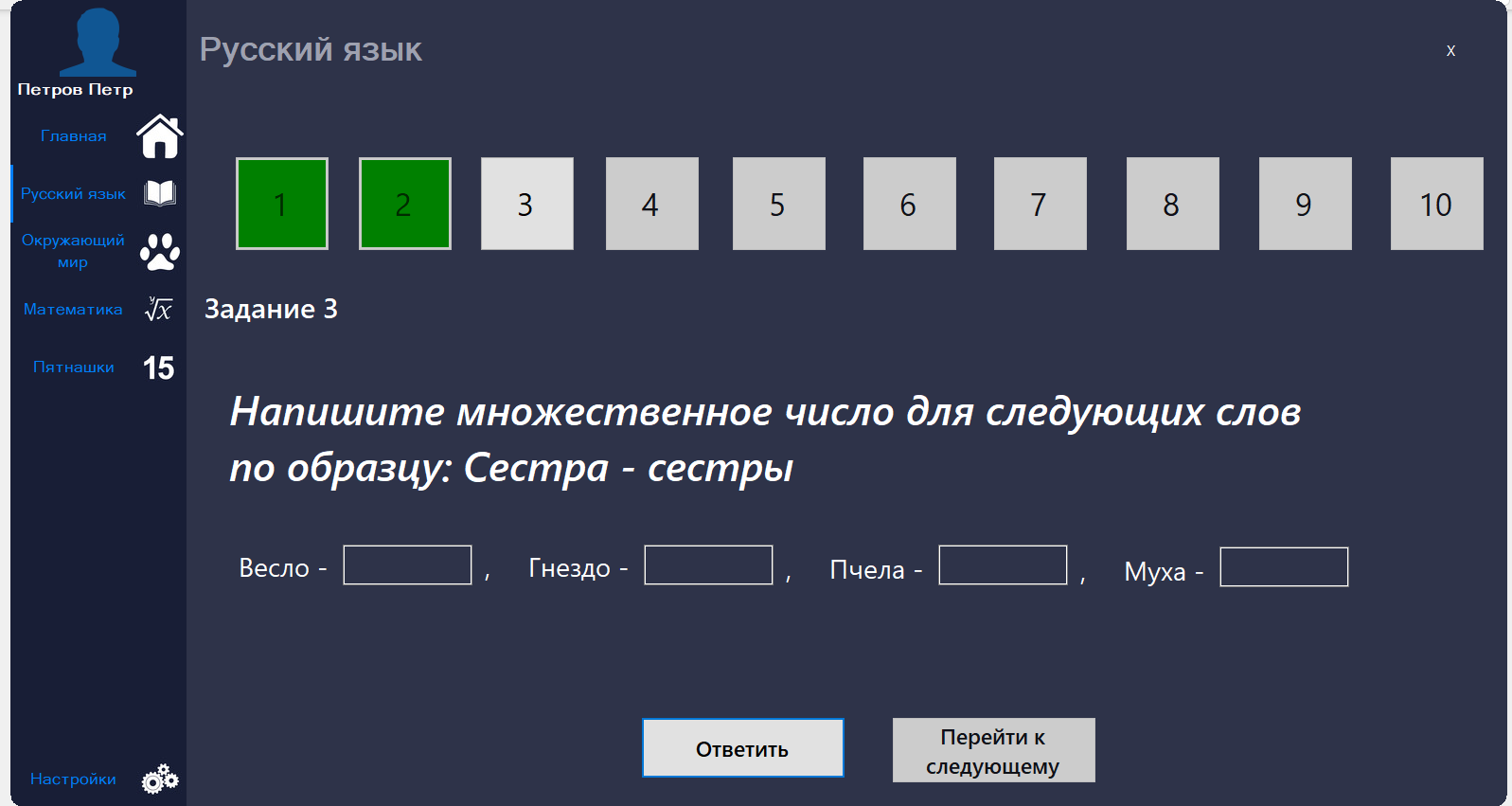
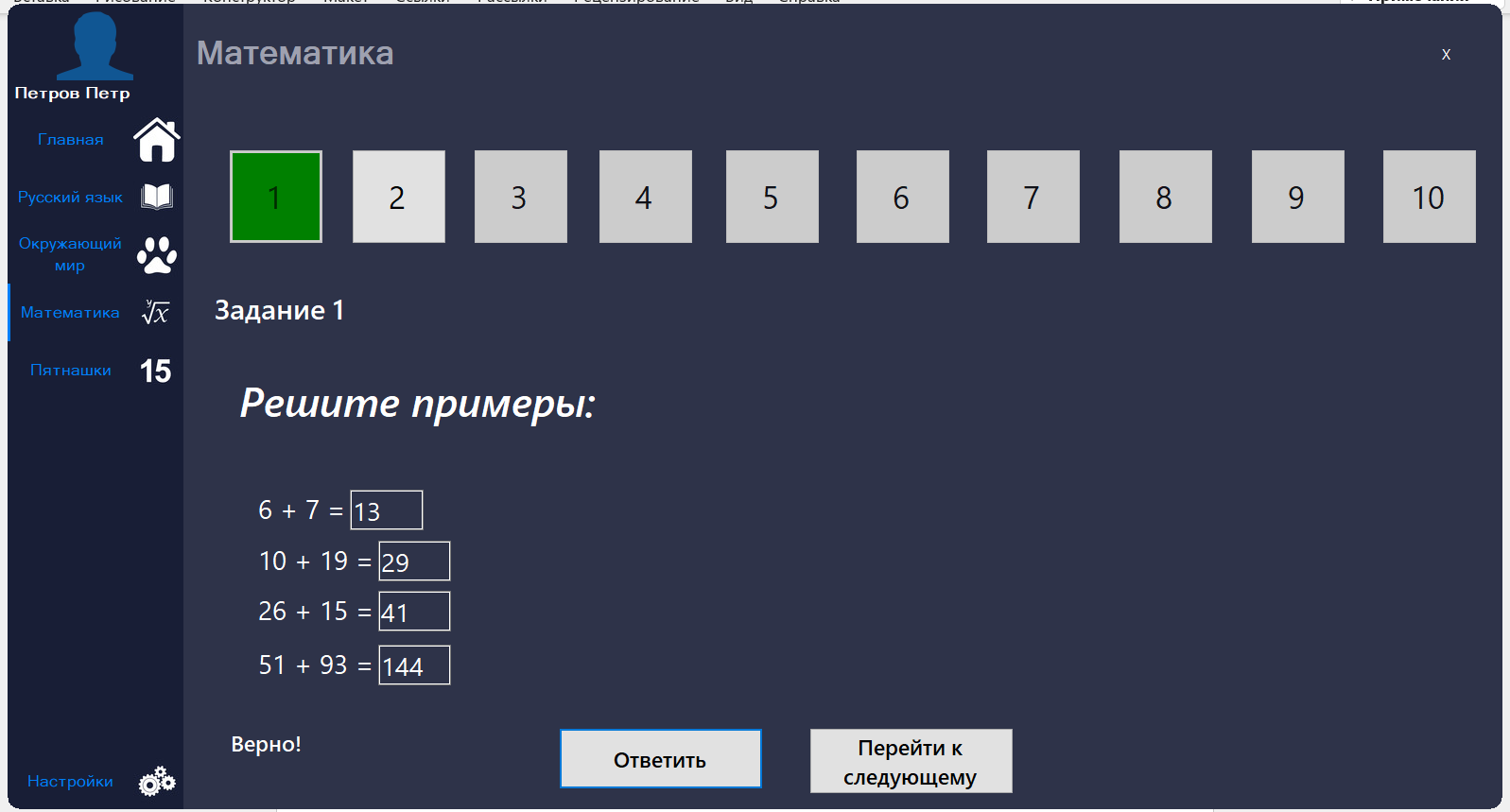
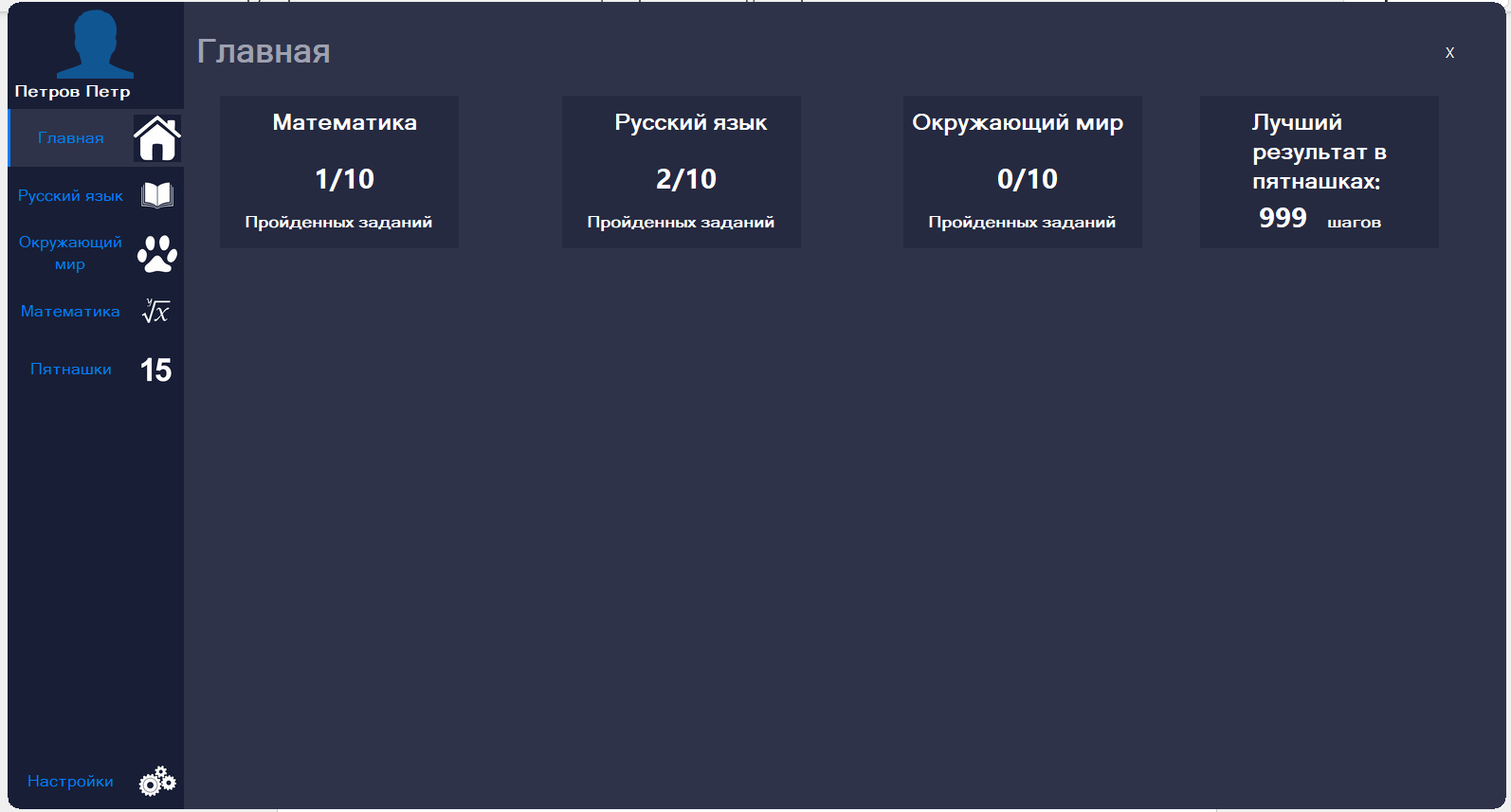
**Тест 2:**  
 Заполним все поля во время регистрации и нажмем на кнопку “Регистрация”.  
**Результат теста 2:** **Получаем сообщение о том, что мы успешно зарегистрировались.**

**Тест 3:**  
 Во время входа в систему введем данные не существующего пользователя.  
**Результат теста 3:  
  
Получаем сообщение об ошибке.**

**Тест 4:**  
 Во время входа в систему введем данные существующего пользователя.  
**Результат теста 4:  
  
Попадаем на главную форму с результатами пользователя, который вошёл в систему.**

## Тестирование отображения результатов ученика на главной форме

**Тест 1:**  
 Зарегистрируем нового пользователя и войдем в систему от его имени.  
**Результат теста 1:** **На главной форме все поля заполнены стандартными значениями, которые даются пользователю во время регистрации.**

**Тест 2:**  
 Выполним несколько заданий за этого пользователя (из теста 1), например, два задания по русскому и одно задание по математике.  
  
  
Затем перейдем на главную форму, выбрав соответствующий пункт меню.  
 **Результат теста 2:  
  
Данные о выполненных пользователем заданиях обновились**

**Тест 3:**  
 Выйдем из системы и зайдем заново под тем же пользователем (из теста 2).  
  
**Результат теста 3:  
Изображение выглядит как текст, монитор, снимок экрана

Автоматически созданное описание  
На главном форме поля обновились с учетом прошлого сеанса данного пользователя.**

# **Заключение**

При выполнении курсовой работы решены следующие задачи:

1. Приобретены навыки создания многооконных приложений на C# с применением Windows Form.  
2. Разработаны модели форм для регистрации, идентификации пользователей, выбора дисциплины, задачи, выполнения задачи и предъявления результата.  
3. Разработаны базы данных и предложены учебные задачи по русскому языку, арифметики и природоведению.  
4. С применением СУБД Microsoft Access реализована база данных приложения.  
5. Реализован прототип развивающего приложения для детей начальных классов.

# **Список источников**

1. Н. В. Каширина М. М. Маран “Программирование на языке c#”.