Комитет образования и науки Волгоградской области

Государственное бюджетное профессиональное образовательное учреждение «Волгоградский технический колледж»

**Курсовой проект**

|  |  |
| --- | --- |
| Профессиональный модуль | ПМ 01 Разработка программных модулей программного обеспечения для компьютерных систем |
| Междисциплинарный курс | МДК 01.02 Прикладное программирование |
| Тема | **Разработка игрового приложения «Угадай число»** |
| Выполнил студент(ка) | Шеховцов Глеб Евгеньевич |
| Группа | ПКС - 1 - 19 |
| Специальность | 09.02.03 Программирование в компьютерных системах |
| Преподаватель | Мочалов Никита Андреевич |

|  |  |
| --- | --- |
| Дата проверки |  |
| Дата защиты |  |
| Оценка |  |

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Н. А. Мочалов

(подпись)

г. Волгоград

2021 г

**Содержание**

[Введение 3](#_Toc88258140)

[1. Анализ требований и определение спецификаций на программный продукт 4](#_Toc88258141)

[1.1. Анализ предметной области 4](#_Toc88258142)

[1.2. Описание программной области 4](#_Toc88258143)

[1.3. Программно-аппаратная платформа реализации 8](#_Toc88258144)

[2. Проектирование структуры и компонентов программного продукта 9](#_Toc88258145)

[2.1. Описание логики функционирования программного продукта 10](#_Toc88258146)

[2.2. Разработка экранных форм и размещение элементов 11](#_Toc88258147)

[2.3. Реализация программного кода в среде Visual Studio 12](#_Toc88258148)

[3. Разработка технической документации для сопровождения программного приложения 19](#_Toc88258149)

[3.1 Системные требования к программному продукту 19](#_Toc88258150)

[3.2 Описание программного модуля. 20](#_Toc88258151)

# Введение

**Актуальность.** Задачи, основанные на угадывании числа, часто рассматривают в качестве примера использования целых чисел и их сравнение. Игровое приложение “угадай число” предназначено для развития логического мышления и умения анализировать ситуацию.

**Объект** - целые числа

**Предмет** - головоломка на скорость, основанная на угадывании.

**Цель** – разработка игрового приложения “угадай число” средствами языка программирования С#.

Для достижения поставленной цели необходимо решение следующих **задач:**

* проанализировать предметную область;
* провести существующих систем аналогов;
* описать логическую структуру игры;
* разработать интерфейс экранных форм;
* провести реализацию программного кода в среде Visual Studio;
* создать инсталлятор для удобства пользователя;
* сформировать основные требования к программному продукту.

# Анализ требований и определение спецификаций на программный продукт

# Анализ предметной области

Головоломка - название жанра компьютерных игр, целью которых является решение логических задач, требующих от игрока задействования логики, стратегии и интуиции.

Разновидности головоломок:

1. Традиционная головоломка;
2. Физическая головоломка.

Головоломки являются очень эффективным средством для развития логического и пространственного мышления, вдобавок к этому головоломки развивают эвристическое мышление.

'Угадай число' это игра, где вы должны угадать число от 1 до 100, выбранное компьютером за определенное количество времени. Это число должно быть угадано методом постепенных попыток. Попытка состоит из предлагаемого числа, выбранного вами, и ответа компьютера. Он должен указать, равно ли ваше введенное число загаданному, если нет – указать больше ли или меньше числа, загаданного компьютером. Число попыток записывается и формирует таблицу рекордов, лидирует тот, кто потратил меньше попыток на отгадку.

# Описание программной области

Для выполнения поставленной задачи могут использоваться различные языки программирования. Выделим среди них:

1. JavaScript;
2. Python;
3. Delphi.

Рассмотрим каждый вариант по отдельности:

1. Вариант, реализованный средствами языка JavaScript, имеет графический интерфейс, который можно гибко настраивать при знании языка гипертекстовой разметки HTML. Игра реагирует на введенные значения и информирует игрока больше или меньше введенное число чем загаданное, так же записываются раннее введенные числа. Игрок проигрывает, если не угадает число за 10 попыток. Недостатки заключаются в том, что данный вариант не записывает количество попыток, не формирует таблицу рекордов и не имеет таймера для создания элемента соревнования между игроками. Данный вариант использует браузер, что довольно неудобно так как оказывает дополнительную нагрузку на систему.

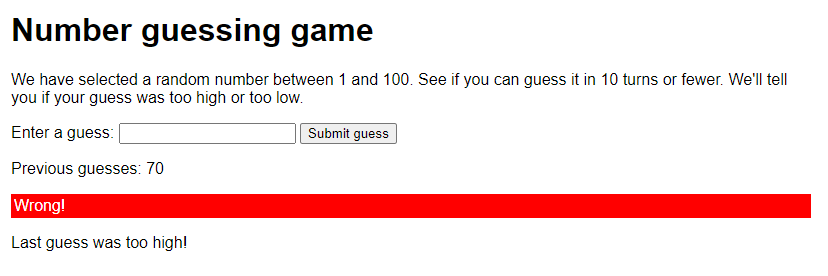


Рисунок 1.1 – Интерфейс JavaScript варианта

1. Вариант, реализованный средствами языка Python, не имеет графического интерфейса, для его создания необходимо использовать дополнительные библиотеки. Игровой процесс проходит в консоли после загрузки скрипта. Игра реагирует на введенные значения и информирует игрока больше или меньше введенное число чем загаданное, раннее введенные числа не записываются. Производится подсчёт попыток, их число записывается при угадывании числа. Игрок проигрывает, если не угадает число за 6 попыток. Данный вариант не имеет таймера, не формирует таблицу рекордов. Не имеет графического интерфейса, его можно реализовать только при использовании сторонних библиотек.
2. Вариант, реализованный средствам языка Delphi, имеет графический интерфейс, но достаточно примитивный. Такие параметры как количество попыток, верхний и нижний числовой предел можно настраивать. Игра реагирует на введенные значения и информирует игрока больше или меньше введенное число чем загаданное, раннее введенные числа не записываются. Игрок проигрывает, если не угадает число за выбранное количество попыток. Недостатки заключаются в том, что данный вариант не записывает количество попыток, не формирует таблицу рекордов и не имеет таймера для создания элемента соревнования между игроками.

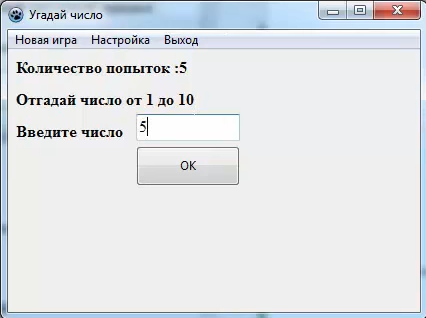


Рисунок 1.2 – Интерфейс Delphi варианта

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Скорость обработки | Читабельность | Простота | GUI (графический интерфейс пользователя) | Графика (2D) | Графика (3D) |
| С | 8 | 6 | 2 | 3 | 5 | 8 |
| С++ | 8 | 6 | 3 | 4 | 6 | 8 |
| C# | 7 | 7 | 5 | 6 | 6 | 7 |
| Java | 6 | 7 | 6 | 7 | 7 | 6 |
| Python | 2 | 5 | 10 | 8 | 10 | 1 |
| [VB.Net](http://VB.net) | 6 | 10 | 8 | 10 | 5 | 2 |

Таблица 1.1 – Сравнительная характеристика языков программирования

Сравнение характеристик языков программирования проводилось по следующим атрибутам:

**Скорость обработки**: как быстро программа будет выполнена.

**Читаемость**: насколько сложно будет читать код.

**Простота**: насколько легко будет работать и использовать библиотеки.

**GUI (графический интерфейс пользователя)**: как легко будет работать с приложением графического интерфейса пользователя.

**Графика (2D)**: простота использования 2D-графики и ее скорости.

**Графика (3D)**: простота использования 3D-графики и ее скорости.

# Программно-аппаратная платформа реализации

Программное решение было реализовано с помощью объектно-ориентированного языка программирования C# с использованием интерфейса программирования приложений Windows Forms.

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, переменные, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Windows Forms - API, отвечающий за графический интерфейс пользователя. Данный интерфейс упрощает доступ к элементам интерфейса Microsoft Windows за счет создания обёртки для существующего Win32 API в управляемом коде. Причём управляемый код — классы, реализующие API для Windows Forms, не зависят от языка разработки. То есть программист одинаково может использовать Windows Forms как при написании ПО на C#, С++, так и на VB.Net, J# и др.

В качестве интегрированной среды разработки (IDE) была выбрана Microsoft Visual Studio Community 2019, так как она является полнофункциональной интегрированной средой разработки для написания, отладки, тестирования и развертывания кода на любой платформе.

Для записи рекордов используется база данных, написанная с помощью декларативного языка программирования SQL. Для администрирования использовалось веб-приложение, написанное на языке PHP, phpMyAdmin, запускаемое на локальном сервере при помощи приложения XAMPP. phpMyAdmin позволяет создать таблицу с помощью графических методов.

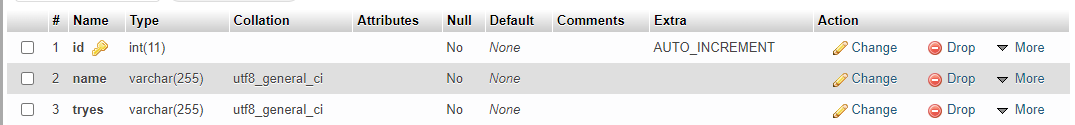


Рисунок 1.3 – Схема Базы данных для рекордов

Разработка программы проводилась на персональном компьютере с использованием операционной системы Windows 10 версии 20H2.

* Модель процессора: Intel Core I7 – 7700T
* Количество ядер процессора: 4
* Количество потоков процессора: 8
* Частота процессора: 2,4 ГГц
* Материнская плата: Gigabyte B250M-DS3H
* Объем оперативной памяти: 8 ГБ
* Видеокарта: AMD RX 570 4GB

**Обобщающий вывод:**

При просмотре приложений для программной разработки были рассмотрены языки JavaScript, C#, Delphi, Python так же их положительные, так и отрицательные свойства.

# 2. Проектирование структуры и компонентов программного продукта

# 2.1. Описание логики функционирования программного продукта

В меню игры пользователь может посмотреть рекорды. Для того что начать игру необходимо ввести имя, пользователь может отслеживать введенные числа. В случае успешного угадывания появляется возможность записать свой рекорд, имеется возможность покинуть игру, не записывая рекорд. Угадать число необходимо за минуту, ведётся подсчёт попыток.

Входной информацией являются числа в пределах от 1 до 100. При вводе числа, программа производит сравнение введенного числа с загаданным компьютером. Если число больше задуманного, программа сообщит нам об этом, схожая ситуация происходит, если число меньше задуманного. Таблица рекордов формируется исходя из количества попыток, затраченных на отгадку числа, чем меньше, тем выше позиция в рейтинге.

Алгоритм угадывания числа представлен на рисунке (рисунок 2.1)

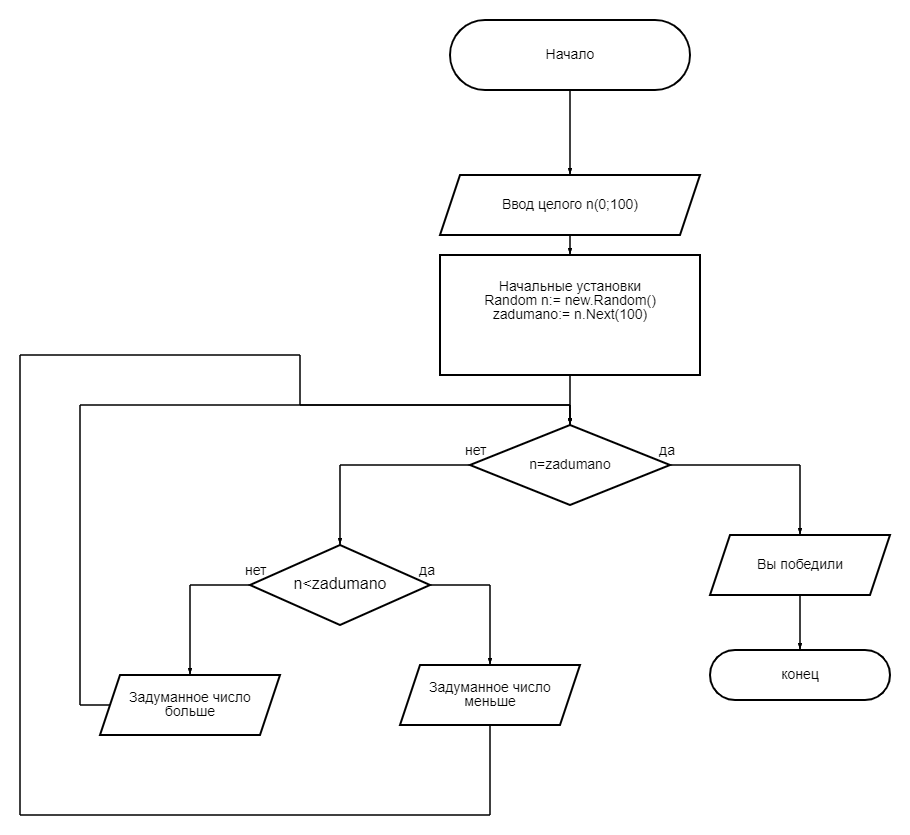


Рисунок 2.1 – Алгоритм угадывания числа

# 2.2. Разработка экранных форм и размещение элементов

**Меню игры**

Размер окна приложения:

Ширина (Width) – 885 px

Высота (Height) – 490 px

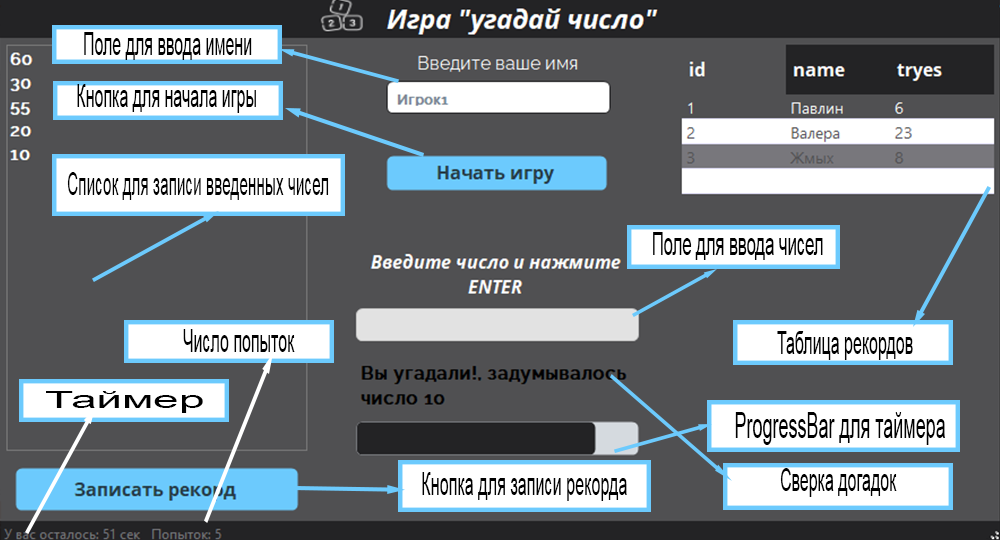
****

Рисунок 2.2 – Меню игры

Меню было разработана с помощью фреймворка Guna UI 2, который позволяет использовать графические элементы WPF на Windows Forms. Guna UI 2 позволила скруглить форму, добавить анимацию запуска.

# 2.3. Реализация программного кода в среде Visual Studio

Листинг программного продукта

|  |
| --- |
| const int t = 60;  //Задаем переменную для задуманного числа  int zadumano = 0;  //Задаем переменную для таймера  int ostalos = 60;  //Задаем переменную для количества попыток  int nomer\_popitki = 0;  //Подключение к БД  DBconnection connect = new DBconnection();  RecordClass record = new RecordClass();  public Form1()  {  InitializeComponent();  //Отображение таблицы рекордов  showRecInfo();  //Фокус на поле ввода имени  guna2TextBox2.Focus();  //Отключение поле ввода числа при старте  guna2TextBox1.Enabled = false;  //Отключение кнопки записать игру  guna2Button1.Enabled = false;  //Отключение кнопки начать игру  guna2Button2.Enabled = false;  label2.Text = "";  //Отображение таймера  toolStripStatusLabel1.Text = "У вас осталось: " + Convert.ToString(t) + " сек";  //Отображение количества попыток  toolStripStatusLabel2.Text = " Попыток: 0 ";  }  private void timer1\_Tick(object sender, EventArgs e)//Событие для работы таймера  {  //Уменьшение таймера на одну секунду каждый тик таймера  ostalos--;  //Уменьшение заполнения progressbar каждый шаг таймера  guna2ProgressBar1.Value--;  //Обновление таймера  toolStripStatusLabel1.Text = "У вас осталось: " + Convert.ToString(ostalos) + " сек";  //Если таймер истёк  if (ostalos == 0)  {  //Отключение таймера  timer1.Enabled = false;  //Отключение поля ввода имени  guna2TextBox1.Enabled = false;  //Отключение progressbar  guna2ProgressBar1.Enabled = false;  label2.Text = "Увы, время истекло...";  }  }  private void guna2TextBox1\_KeyPress(object sender, KeyPressEventArgs e)//Событие при нажатии кнопок в поле ввода чисел  {  //Если введено нажата клавиша Enter и введено число  if (e.KeyChar.Equals((char)13))  {  //Введено число  try  {  //Добавление введенного числа в список  listBox1.Items.Add(guna2TextBox1.Text);  //Если введенное число равно задуманному  if (Convert.ToInt16(guna2TextBox1.Text) == zadumano)  {  //Отключение таймера  timer1.Enabled = false;  //Отключение поля ввода чисел  guna2TextBox1.Enabled = false;  //Отображение информации о победе  label2.Text = "Вы угадали!, задумывалось число " + Convert.ToString(zadumano);  //Включение кнопки записи рекорда  guna2Button1.Enabled = true;  };  //Если введенное число меньше задуманного  if (Convert.ToInt16(guna2TextBox1.Text) > zadumano) label2.Text = "Задуманное число меньше";  //Если введенное число больше задуманного  if (Convert.ToInt16(guna2TextBox1.Text) < zadumano) label2.Text = "Задуманное число больше";  }  //Если введено не число  catch { label2.Text = "Некорректные входные данные!"; } //Сообщение о вводе некорректных данных  //Увеличение счетчика попыток  nomer\_popitki++;  //Изменение информации в счетчика попыток  toolStripStatusLabel2.Text = " Попыток: " + Convert.ToString(nomer\_popitki);  //Очистка поля при нажатии Enter  guna2TextBox1.Text = "";  //Фокус на поле ввода текста  guna2TextBox1.Focus();  }  }  bool verify()  {  if ((guna2TextBox2.Text == ""))  {  return false;  }  else  {  return true;  }  }  public void showRecInfo()//Вывод информации из БД  {  guna2DataGridView1.DataSource = record.getreclist();  DataGridViewColumn Column = new DataGridViewColumn();  Column = guna2DataGridView1.Columns[2];  }  private void guna2Button1\_Click(object sender, EventArgs e)//Добавление рекорд  {  //Переменная для имени  string name = guna2TextBox2.Text;  //Переменная для номера попытки  string tryes = Convert.ToString(nomer\_popitki);  if (verify())  {  //Добавление имени и числа попыток в БД  try  {  if (record.insertrec(name, tryes))  {  //Сообщение об успехе  MessageBox.Show("Новые данные успешно добавлены", "Сообщение", MessageBoxButtons.OK, MessageBoxIcon.Information);  //Начало новой игры  Application.Restart();  }  }  catch (Exception ex)  {  MessageBox.Show(ex.Message, "Введите своё имя для записи рекорда", MessageBoxButtons.OK, MessageBoxIcon.Error);  }  }  }  private void guna2TextBox2\_KeyPress(object sender, KeyPressEventArgs e)//Событие при нажатии в поле ввода имени  {  //Если количество символов в поле имя меньше или равно двум  if (guna2TextBox2.TextLength <= 2)  {  //Отключение кнопки начать игру  guna2Button2.Enabled = false;  }  //Если количество символов в поле имя больше двух  else  {  //Включение кнопки начать игру  guna2Button2.Enabled = true;  }  }  private void guna2Button2\_Click(object sender, EventArgs e)//Нажатие на кнопку начать игру  {  //Включение поля ввода чисел  guna2TextBox1.Enabled = true;  //Включение таймера  timer1.Enabled = true;  //Интервал таймера в одну секунду  timer1.Interval = 1000;  //Инициализация progressbar  guna2ProgressBar1.Maximum = t;  guna2ProgressBar1.Value = t;  //Рандом  Random n = new Random();  //Рандомизация для переменной задумано  zadumano = n.Next(100);  }  } |

# 3. Разработка технической документации для сопровождения программного приложения

# 3.1 Системные требования к программному продукту

* Также поддерживается установка в Windows 10 на ARM, Windows 10 Корпоративная LTSC, Windows Server 2012 и Windows Server 2008 R2 с пакетом обновления 1 (SP1)
* При использовании архитектуры x86 или AMD64/x64 требуется процессор с тактовой частотой не ниже 1,6 ГГц
* Требуется 1 ГБ ОЗУ (1,5 ГБ при запуске на виртуальной машине)
* Требуется 1 ГБ доступного пространства на жестком диске
* Требуется разрешение экрана не ниже 1024 на 768
* Для получения наилучших результатов используйте средства диагностики с самым последним обновлением для вашей версии Visual Studio.

# 3.2 Описание программного модуля.

**Общие сведения**

Название программы: «Ugaday 4islo»

Используемый файл: Ugaday4islo.sln/ Ugaday4islo.exe

Программный модуль написан на языке C# NET Framework

**Функциональное назначение**

Программа предназначена для демонстрации игрового процесса игры «Угадай число».

**Вызов и загрузка**

Вызов разработанной программы осуществляется запуском компиляции в программе Visual Studio 2019 или запуска .exe файла

**Входные данные**

Не используются.

**Выходные данные**

Не используются

**ЗАКЛЮЧЕНИЕ**

В представленной курсовой работе было разработано приложение игры головоломки, являющейся разновидностью игры «Угадай число». Программа написана на языке программирования С# с использованием среды программной разработки Visual Studio NET.Framework.

Разработанная программа соответствует требованиям, указанным в задании на курсовую работу.

**СПИСОК ЛИТЕРАТУРЫ**

1. Методическое пособие по курсовой работе.

2. Правила оформления; ГОСТ Р 7.0.100-2018.

3. Комплекс стандартов и руководящих документов на автоматизированные системы РД 50-34.698-90.

4. Межгосударственный стандарт. Единая система программной документации ГОСТ 19.402-78 ЕСПД.

5. Программа Visual Studio 2019.

6. Программа MSWord 2016.

7. Сайт «METANIT» https://metanit.com/sharp/