**Содержание**

[**Введение** 2](#_Toc183592329)

[**1.** **Основание для разработки** 4](#_Toc183592330)

[**2.** **Назначение разработки** 5](#_Toc183592331)

[**3.** **Требования к программе** 6](#_Toc183592332)

[**3.1** **Функциональные требования** 6](#_Toc183592333)

[**3.2** **Требования к надёжности** 6](#_Toc183592334)

[**3.3** **Требования к условиям эксплуатации** 6](#_Toc183592335)

[**3.5** **Требования к информационной и программной совместимости** 7](#_Toc183592336)

[**3.6** **Требования к маркировке и упаковке** 7](#_Toc183592337)

[**3.7 Требования к транспортировке и хранению** 7](#_Toc183592338)

[**4.** **Требования к программной документации** 8](#_Toc183592339)

[**5.** **Стадии и этапы разработки** 9](#_Toc183592340)

[**6.** **Порядок контроля и приёмки** 10](#_Toc183592341)

# **Введение**

Настоящее техническое задание составлено для разработки компьютерной игры в жанре «Змейка», реализованной с использованием технологии Windows Presentation Foundation (WPF). Разработка направлена на создание увлекательной и интуитивно понятной игры, которая обеспечит пользователям приятное времяпрепровождение и познакомит их с классическим игровым процессом в современной реализации.

Цель проекта — создание интерактивной игры, которая позволит пользователям управлять змейкой, собирать бонусы, избегать столкновений и соревноваться за наилучший результат. Проект будет включать базовые элементы игрового процесса, такие как управление персонажем, нарастающая сложность и система подсчёта очков. Интерфейс игры будет простым и удобным, что сделает её доступной как для новичков, так и для опытных.

Для достижения цели поставлены следующие задачи:

1. Проанализировать требования пользователей к классическим играм и их предпочтения.

2. Разработать спецификации для игры с описанием механик и интерфейса.

3. Изучить современные подходы к созданию игр с использованием WPF.

4. Спроектировать игровой интерфейс с элементами минималистичного дизайна.

5. Реализовать основные игровые механики, включая управление змейкой и генерацию препятствий.

6. Подготовить сценарии тестирования и наборы тестов для проверки корректности работы игры.

Актуальность проекта:

1. Классическая игра «Znake Game» остаётся популярной благодаря своей простой, но увлекательной игровой механике.

2. Реализация игры с использованием WPF позволит использовать современные технологии.

3. Проект может быть полезен для начинающих разработчиков, изучающих основы создания интерфейсов и игровых приложений.

Создание игры «Znake Game» с использованием WPF отвечает требованиям современных пользователей и предоставляет возможности для творческой реализации. Это делает данный проект значимым вкладом в развитие небольших, но качественных игровых приложений.

# **Основание для разработки**

Разработка игры «Znake Game» обоснована её популярностью среди пользователей, а также необходимостью создания увлекательного игрового опыта с использованием современных технологий. Такие проекты позволяют привлечь широкую аудиторию, включая тех, кто ищет простые и увлекательные игры для кратковременного отдыха, так называемые «игры-таймкиллеры». Уникальные игровые возможности, такие как плавное управление, постепенное усложнение игрового процесса и система подсчёта очков, создадут интересную и интуитивно понятную механику.

Разнообразие функций, например динамическая генерация препятствий, обеспечит пользователям интерес к долгосрочному использованию приложения. Реализация игры с использованием технологии WPF позволит оптимизировать её работу для настольных устройств, предоставляя комфортный игровой процесс в любой операционной среде.

Социальные элементы, такие как возможность соревноваться за рекорды или делиться своими достижениями, добавят соревновательности и укрепят связь между пользователями. Эти аспекты позволят игре выделиться среди других решений, ориентированных на минималистичные игры, и привлечь внимание целевой аудитории.

Игра будет доступна как для опытных пользователей, так и для новичков, что значительно расширит её привлекательность. Это обеспечит интерес как у тех, кто ценит классические игры, так и у разработчиков, желающих изучить технологии WPF. Такой подход повысит образовательную значимость проекта, способствуя его долгосрочному успеху на рынке приложений.

Таким образом, создание игры «Znake Game» с использованием WPF полностью отвечает современным запросам пользователей и разработчиков, представляя собой значимый проект в области разработки небольших, но качественных приложений.

# **Назначение разработки**

Проект направлен на создание увлекательной и доступной игры «Znake Game», которая привлечёт разнообразную аудиторию пользователей. Основной целью является предоставление интуитивно понятного игрового процесса, который позволит игрокам сосредоточиться на достижении рекордов, избегая препятствий и поедая яблоки.

Игра будет включать уникальные игровые элементы, такие как динамическое увеличение сложности и система подсчёта очков. Это обеспечит захватывающий и эффективный игровой процесс, позволяя пользователям с лёгкостью наслаждаться игрой, независимо от уровня их опыта. Разнообразие функций будет поддерживать интерес пользователей и стимулировать их возвращаться к игре снова и снова.

Цель разработки — создать уникальную, высококачественную версию классической игры «Znake Game», которая будет интересна как любителям ретро-игр, так и новым пользователям. Эти аспекты делают проект перспективным, обеспечивая ему долгосрочный успех и выделяя среди других решений в данной категории.

# **Требования к программе**

## **Функциональные требования**

Программа должна выполнять следующие функции:

1. Управление змейкой с помощью клавиш «WASD»;
2. Генерация яблок и препятствий;
3. Увеличение счёта и размера змейки при подборе яблок;
4. Конец игры при столкновении змейки с краем игрового поля/препятствия;
5. Подсчёт очков и вывод их на экран

## **Требования к надёжности**

1. Стабильность - программа должна работать без сбоев;
2. Неправильные действия пользователей - программа должна корректно обрабатывать неправильные действия пользователей, не приводя к краху;

## **Требования к условиям эксплуатации**

Программа должна функционировать в климатических условиях, соответствующих санитарным нормам.

Пользователи данного приложения должны владеть начальными навыками пользователя персонального компьютера.

* 1. **Требования к составу и параметрам технических средств**

Минимальные требования:

* Windows IQ/11

## **Требования к информационной и программной совместимости**

Программа должна работать под управлением операционной системы с поддержкой платформы не ниже Win32

## **Требования к маркировке и упаковке**

Дополнительных требований к маркировке и упаковке не предъявляется.

## **3.7 Требования к транспортировке и хранению**

Данное приложение размещается на сервере, поэтому не требует наличия какого-либо носителя.

# **Требования к программной документации**

Должна быть разработана программная документация:

1. Хз что писать Субхан всё равно не посмотрит
2. Техническое задание

# **Стадии и этапы разработки**

Стадии и этапы, представленные в таблице 1.

Таблица 1 – Стадии и этапы разработки шаблона для взаимодействия с веб-сервисом.

|  |  |  |  |
| --- | --- | --- | --- |
| **Стадия разработки** | **Этап работ** | **Сроки исполнения** | **Ответственный** |
| Рабочий проект | Разработка программы | 27.11.2024 |  |
| Разработка программной документации | 27.11.2024 |  |
| Тестирование программы | 27.11.2024 |  |

# **Порядок контроля и приёмки**

Об окончании выполнения каждого этапа работ исполнитель обязан уведомить руководителя. По окончании каждого этапа производится приёмка данного этапа руководителем. При отсутствии претензий, производится подписание акта приёмки-сдачи. Претензии к уже принятым этапам не принимаются.

По окончании разработки программа тестируется в присутствии руководителя. В случае мотивированного отказа руководителя принять работу, составляется акт с перечнем необходимых доработок и сроков их выполнения.