

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

«Учебная игра Snake 3D»

Код: SNAKE-3D-001

Разработчики:

Мурехин Ярослав Андреевич

Воронин Артём Тимофеевич

Виноградова Дарья Сергеевна

Нурпиисов Никита Игоревич

СОДЕРЖАНИЕ

1 ВВЕДЕНИЕ	3
1.1 Наименование программы.....	3
1.2 Краткая характеристика области применения	3
2 ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ.....	4
2.1 Основание для проведения разработки.....	4
2.2 Наименование и условное обозначение темы разработки	4
3 НАЗНАЧЕНИЕ РАЗРАБОТКИ	5
3.1 Функциональное назначение	5
3.2 Эксплуатационное назначение	5
3.3 Задачи, решаемые в ходе разработки.....	5
4. ТРЕБОВАНИЯ К ПРОГРАММЕ	6
4.1 Требования к функциональным характеристикам.....	6
4.1.1 Требования к составу выполняемых функций.....	6
4.1.2 Требования к организации входных данных	6
4.1.3 Требования к организации выходных данных.....	6
4.2 Требования к надежности.....	6
4.2.1 Требования к обеспечению надежного функционирования программы	6
4.2.2 Время восстановления после отказа	7
4.2.3 Отказы из-за некорректных действий пользователя	7
4.3 Условия эксплуатации	7
4.4 Требования к составу и параметрам технических средств	7
4.5 Требования к информационной и программной совместимости	7
5 ТРЕБОВАНИЯ К ИНТЕРФЕЙСУ	8
6 ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ	9
7 ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ	10
8 ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ	11
9 СТАДИИ И ЭТАПЫ РАЗРАБОТКИ	13
9.1 План-график разработки.....	13

1 ВВЕДЕНИЕ

1.1 Наименование программы

Snake 3D — учебное игровое приложение на pygame, реализующее классическую змейку в псевдо-3D графике.

1.2 Краткая характеристика области применения

Программа предназначена для использования в учебном процессе по дисциплине «Тестирование и верификация программного обеспечения». Игра содержит преднамеренные ошибки, что позволяет студентам тренироваться в поиске багов и написании тестовой документации.

2 ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ

2.1 Основание для проведения разработки

Необходимость создания учебного примера для дисциплины «Тестирование и верификация ПО».

2.2 Наименование и условное обозначение темы разработки

Наименование: «Учебная игра Snake 3D»

Шифр темы: SNAKE-3D-001

3 НАЗНАЧЕНИЕ РАЗРАБОТКИ

3.1 Функциональное назначение

Программа должна обеспечивать возможность игрового процесса «Змейка» с базовыми функциями управления, начисления очков, постановки паузы и отображения интерфейса.

3.2 Эксплуатационное назначение

Использование в качестве учебного проекта: для студентов, преподавателей и тестировщиков при изучении процессов тестирования, верификации и оформления документации.

3.3 Задачи, решаемые в ходе разработки

- Реализация игрового цикла и управления.
- Обработка событий клавиатуры и кнопок меню.
- Реализация механики поедания яблок, увеличения длины и счёта.
- Отображение текущего и лучшего результата.
- Создание преднамеренных ошибок для учебных целей.

4. ТРЕБОВАНИЯ К ПРОГРАММЕ

4.1 Требования к функциональным характеристикам

4.1.1 Требования к составу выполняемых функций

- Запуск главного меню с кнопками Start, How to Play, Quit.
- Запуск игрового процесса со змейкой.
- Управление стрелками или WASD.
- Постановка паузы (P), возобновление игры, рестарт (R или Enter).
- Выход в меню (Esc).
- Начисление очков при съедании яблок.
- Хранение лучшего результата.
- Слайдер изменения скорости игры.

4.1.2 Требования к организации входных данных

- Управляющие команды пользователя с клавиатуры.
- События мыши (для кнопок меню и слайдера).

4.1.3 Требования к организации выходных данных

- Отображение текущего счёта и лучшего результата.
- Визуализация игрового поля, змейки и яблок.
- Текстовые сообщения (Game Over, Help).

4.2 Требования к надежности

4.2.1 Требования к обеспечению надежного функционирования программы

Программа должна устойчиво работать при стандартных действиях пользователя. Предусмотреть обработку невалидных действий (например, нажатие противоположного направления).

4.2.2 Время восстановления после отказа

Перезапуск программы должен занимать не более 30 секунд.

4.2.3 Отказы из-за некорректных действий пользователя

Некорректные команды не должны приводить к краху приложения.

4.3 Условия эксплуатации

- ОС: Windows 10/11, Linux, macOS (Python 3.10+).
- Не требуется интернет-соединение для работы.

4.4 Требования к составу и параметрам технических средств

- CPU: 1 ГГц или выше.
- RAM: от 512 МБ.
- HDD/SSD: не менее 100 МБ свободного места.

4.5 Требования к информационной и программной совместимости

- Язык программирования: Python 3.10+.
- Библиотека: pygame.

5 ТРЕБОВАНИЯ К ИНТЕРФЕЙСУ

- Главное меню с кнопками Start, How to Play, Quit.
- Экран паузы с Resume, Restart, Menu, Quit, а также слайдер скорости.
- Экран Help с текстовой справкой по управлению.
- Экран Game Over с кнопкой возврата в меню.

6 ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

- Экономический эффект: повышение качества обучения тестированию ПО.
- Затраты: минимальные, так как разработка ведётся студентами.

7 ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

В состав входят:

- Техническое задание.
- Руководство пользователя.
- Техническая документация.
- Тестовая документация.

8 ПОРЯДОК КРИТЕРИЕВ И ПРИЕМКИ

Критерии приёмки и порядок контроля строятся на методе «чёрного ящика», то есть тестирующая команда не имеет доступа к исходному коду, а проверка ведётся исключительно через взаимодействие с интерфейсом и наблюдение за поведением программы.

Основные метрики и пороги успешности включают в себя стабильный запуск приложения без ошибок и сбоев, доступность всех игровых сцен и элементов интерфейса, корректность отображения игрового поля и счётчиков. В качестве количественных критериев устанавливается успешное прохождение всех критических сценариев и не менее 95% тестов в целом, отсутствие падений или зависаний во время игры, стабильная реакция на управление с задержкой не более 100 мс, а также корректная работа программы на разных операционных системах с заявленной конфигурацией.

Процедура приёмки предполагает последовательное тестирование. Сначала проверяется сборка, то есть целостность файлов и возможность запуска игры на целевых системах. Затем проводится проверка игровых сцен: стартовое меню с кнопками «Start», «Help», «Quit», переход в игровую сцену по нажатию кнопки или клавиши Enter, отображение змейки из трёх сегментов и одного яблока при начале игры, корректный возврат в меню из справки. Особое внимание уделяется тому, что все переходы происходят без задержек, кнопки реагируют на клики, а визуальное оформление совпадает с заявленным.

Запуск и сцены рассматриваются как основные точки приёмки. Игра должна открываться в отдельном окне с корректным заголовком, в меню должны отображаться все элементы управления, кнопки должны быть кликабельны, а переход в игровой процесс — мгновенным и предсказуемым. Сцена помощи должна содержать описание управления и целей игры, из неё должен выполняться возврат в главное меню. Экран «Game Over» обязан

корректно появляться при проигрыше и содержать кнопки для рестарта, выхода в меню и завершения работы приложения.

Все проверки осуществляются методом «чёрного ящика» и фиксируются в приёмочном протоколе. Успешной приёмка считается в случае прохождения всех критических сценариев и достижения установленных пороговых значений метрик.

9 СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

9.1 План-график разработки

Таблица 9.1.1 – Этапы разработки

№	Этап	Срок исполнения	Исполнитель
1	Анализ требований и проектирование	3 дня	Разработчик
2	Написание кода и модульное тестирование	7 дней	Разработчик
3	Наполнение контентом (графика, меню)	2 дня	Разработчик
4	Пилотная эксплуатация и тестирование	3 дня	Совместно
5	Сдача и приёмка	1 день	Разработчик
ИТОГО		16 дней	