

**АНАЛИТИЧЕСКИЙ ОТЧЁТ**  
о разработке программного продукта  
*«Эхо Манёвра»*  
с использованием технологий генеративного  
искусственного интеллекта

28 ноября 2025 г.

# Содержание

<b>1</b>	<b>ВВЕДЕНИЕ</b>	<b>3</b>
1.1	Наименование проекта . . . . .	3
1.2	Цель работы . . . . .	3
1.3	Задачи . . . . .	3
<b>2</b>	<b>МЕТОДОЛОГИЯ РАЗРАБОТКИ</b>	<b>3</b>
2.1	Используемые инструменты . . . . .	3
2.2	Метод «Вайбкодинга» . . . . .	3
2.3	Процесс разработки . . . . .	3
2.3.1	Итерация 1: Прототипирование (MVP) . . . . .	3
2.3.2	Итерация 2: Архитектурная оптимизация . . . . .	4
2.3.3	Итерация 3: Визуальная полировка . . . . .	4
<b>3</b>	<b>ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ</b>	<b>4</b>
3.1	Архитектура . . . . .	4
3.2	Решение проблемы «Эха» . . . . .	4
3.3	Физические параметры . . . . .	4
<b>4</b>	<b>РЕЗУЛЬТАТЫ И АНАЛИЗ</b>	<b>5</b>
4.1	Сравнение подходов . . . . .	5
4.2	Выявленные ограничения . . . . .	5
<b>5</b>	<b>ЗАКЛЮЧЕНИЕ</b>	<b>5</b>

# 1 ВВЕДЕНИЕ

## 1.1 Наименование проекта

Браузерная игра-платформер «Эхо Манёвра» (англ.: *Echo Maneuver*).

## 1.2 Цель работы

Разработка полнофункционального игрового приложения с использованием методов генеративного искусственного интеллекта (GenAI) и технологии «вайбкодинга» (англ.: *Vibecoding*) на платформе WebSim. Исследование эффективности применения больших языковых моделей (LLM) для ускоренной разработки программного обеспечения.

## 1.3 Задачи

1. Сформировать концепцию и техническое задание с помощью ИИ-мозгового штурма.
2. Реализовать программный код игры, используя естественный язык (пром프트-инжиниринг).
3. Обеспечить корректную работу сложной механики «темпорального эха» (запись и воспроизведение действий игрока).
4. Провести тестирование и отладку полученного продукта.
5. Документировать процесс разработки.

# 2 МЕТОДОЛОГИЯ РАЗРАБОТКИ

## 2.1 Используемые инструменты

- **Платформа:** WebSim.ai (облачная среда).
- **Модель:** GPT-5.1 / Claude 3.5 Sonnet.
- **Стек:** HTML5, CSS3, JavaScript (Vanilla JS), Canvas API.

## 2.2 Метод «Вайбкодинга»

Разработка велась методом **декларативного описания функционала**. Вместо императивного кодирования применялся подход, при котором:

- Разработчик выступает в роли архитектора, описывая требования на естественном языке.
- Нейросеть генерирует исполняемый код.
- Процесс сводится к итеративному уточнению промптов, а не отладке синтаксиса.

## 2.3 Процесс разработки

### 2.3.1 Итерация 1: Прототипирование (MVP)

Была сгенерирована базовая версия. Выявлена проблема **непроходимости уровней** из-за несоответствия физики прыжка и высоты платформ.

### 2.3.2 Итерация 2: Архитектурная оптимизация

Внедрен метод **ASCII-карт** для жесткого контроля геометрии. Уровни стали описываться символьной сеткой, что гарантировало проходимость.

```
1 #####
2 #P.....#
3 #####.....#
4 #.....#
```

### 2.3.3 Итерация 3: Визуальная полировка

Примитивы заменены на процедурные текстуры (кирпич, неон) средствами Canvas API.

## 3 ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ

### 3.1 Архитектура

Приложение — Single Page Application (SPA). Основные классы:

- **Game:** Главный цикл (`requestAnimationFrame`).
- **Player:** Физика и коллизии (AABB).
- **ShadowSystem:** Кольцевой буфер для механики эха.
- **TileMap:** Парсер уровней.

### 3.2 Решение проблемы «Эха»

Алгоритм синхронизации:

1. **Запись:** Кадр за кадром сохраняются координаты  $\{x, y\}$ .
2. **Воспроизведение:** Тень берет координаты из буфера с индексом  $T - 180$ .
3. **Коллизия:** Пересечение хитбоксов вызывает `GAME_OVER`.

### 3.3 Физические параметры

Параметр	Значение
GRAVITY	0.6
JUMP_FORCE	-14
TILE_SIZE	40px
SHADOW_DELAY	180 кадров (3 сек)

Таблица 1: Константы физического движка

Критерий	Ручной код	WebSim (AI)
Время на MVP	4-6 часов	15 минут
Отладка	Сложная	Быстрая
Навыки	Deep JS	Prompt Engineering

## 4 РЕЗУЛЬТАТЫ И АНАЛИЗ

### 4.1 Сравнение подходов

### 4.2 Выявленные ограничения

1. **Галлюцинации физики:** ИИ может создать непроходимый уровень. Решение: использование жестких ограничений (ASCII-сетка).
2. **Контекстное окно:** При генерации большого кода он может обрезаться. Решение: модульная генерация.

## 5 ЗАКЛЮЧЕНИЕ

Разработанный продукт «Эхо Манёвра» полностью соответствует ТЗ. Применение вайбкодинга сократило время разработки в 8-10 раз. Проект демонстрирует высокий потенциал LLM в геймдеве.