

Ду Гомин, магистрант,  
Амурский государственный университет

## ПРОЕКТИРОВАНИЕ МОДУЛЕЙ ИГРЫ И СТРАТЕГИИ УЛУЧШЕНИЯ ИГРОВОГО ПРОЦЕССА В ИГРЕ «ТАНКОВЫЙ БОЙ»

**Аннотация.** В статье рассматривается дизайн функциональных модулей и стратегии улучшения игрового процесса в игре «Танковый бой» на Java. Основное внимание уделено объектно-ориентированному подходу к проектированию ключевых модулей: управления танками, обработки столкновений и визуальных эффектов. Проанализированы методы повышения вовлеченности пользователей через систему стратегических бонусов и редактор пользовательских карт.

**Ключевые слова:** Java, игровой дизайн, модульность, игровой процесс, обработка столкновений.

Современная разработка игр требует сочетания технического совершенства и увлекательного игрового опыта, и классическая игра «Танковый бой» становится удобной основой для демонстрации преимуществ модульной архитектуры и способов углубления геймплея. Описание структуры функциональных модулей и анализ методов повышения вариативности позволяют обеспечить не только стабильность работы, но и увеличить реиграбельность, продлевая жизненный цикл приложения.

Принцип модульности особенно ясно проявляется в системе управления. Обработка ввода с клавиатуры разделена на два разных режима: непрерывное движение, обеспечивающее плавность перемещения танка, и дискретная стрельба, задающая ритм боя. Такое поведение реализовано через механизм событий Java, что гарантирует точную и своевременную реакцию на действия игрока.

Далее логический модуль принимает на себя ключевую задачу расчёта столкновений. Снаряды взаимодействуют с танками и объектами окружения: повреждают технику, уничтожают её при нулевом уровне здоровья, разрушают кирпичные блоки или отскакивают от металлических поверхностей. Все вычисления производятся на стороне сервера, что исключает возможность нечестной игры и поддерживает единые условия для всех участников.

В визуальном модуле эти данные превращаются в графические элементы. Его работа включает отображение анимаций, в частности эффектов разрушения через класс Explode, который формирует визуальную обратную связь и усиливает эмоциональный отклик игрока на происходящее в бою.

Для увеличения глубины игрового процесса применяется система стратегических бонусов. Усиления появляются на карте случайным образом и могут значительно изменить ход сражения: повышение здоровья, увеличение огневой мощи, временная неуязвимость, мгновенное уничтожение противника или заморозка вражеских танков. Появление таких объектов формирует динамичные зоны контроля и побуждает игроков менять тактику в реальном времени.



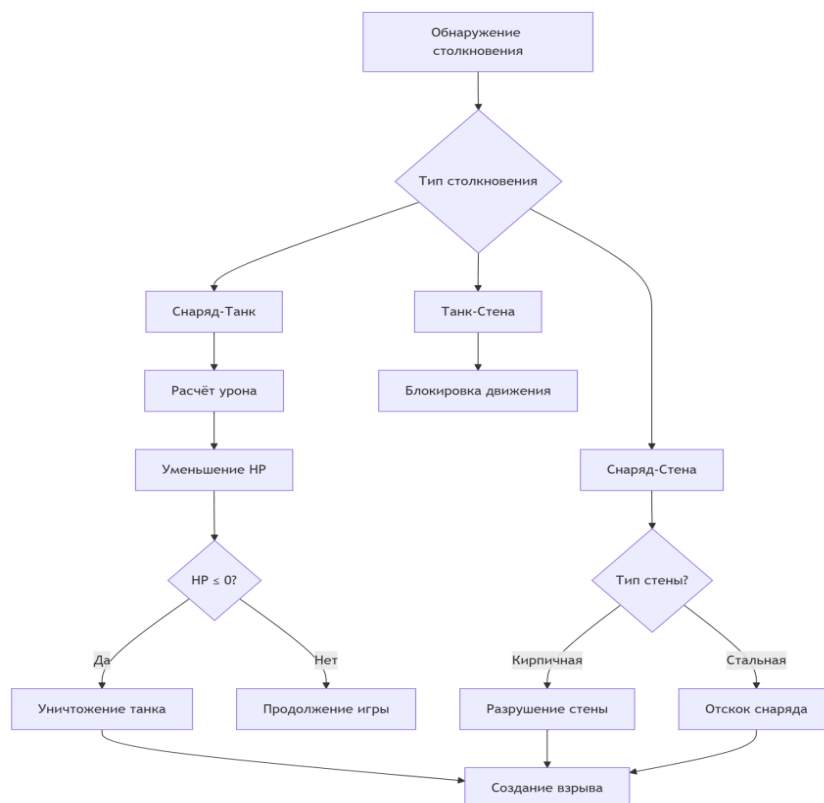


Рисунок 1. Система обработки столкновений

Для увеличения глубины игрового процесса применяется система стратегических бонусов. Усиления появляются на карте случайным образом и могут значительно изменить ход сражения: повышение здоровья, увеличение огневой мощи, временная неуязвимость, мгновенное уничтожение противника или заморозка вражеских танков. Появление таких объектов формирует динамичные зоны контроля и побуждает игроков менять тактику в реальном времени.

Дополняет общую концепцию редактор пользовательских карт. Он основан на сетке  $13 \times 13$  и позволяет игрокам самостоятельно размещать элементы окружения, формируя уникальные конфигурации уровней. Использование тех же объектов, что и в режиме игры, демонстрирует гибкость архитектуры и позволяет легко расширять функциональность.

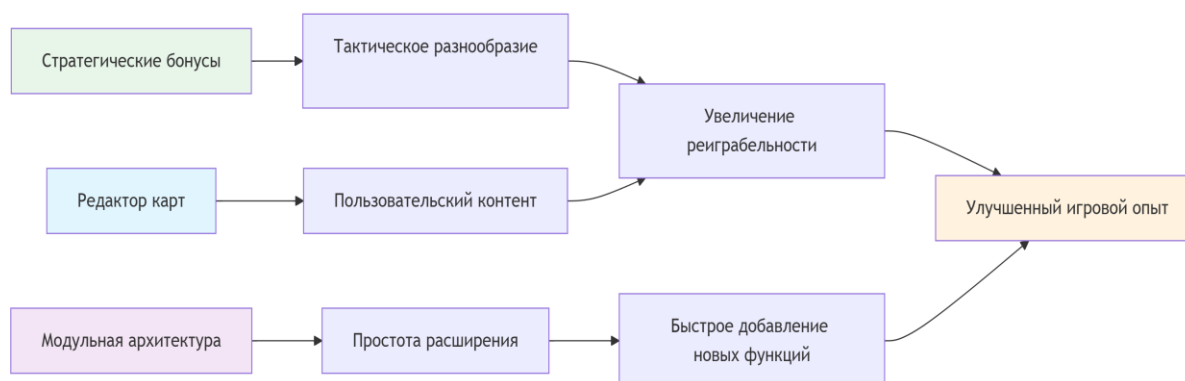


Рисунок 2. Система улучшения игрового процесса



После реализации ключевых модулей и игровых механик проведено комплексное тестирование. Проверялась корректность обработки столкновений, функционирование бонусов и работа редактора карт. Испытания подтвердили устойчивость всех элементов взаимодействия, а игроки отметили повышение динамичности, вариативности и общего качества игрового опыта.

Финальные результаты показывают, что выбранная модульная архитектура полностью оправдала себя. Она обеспечивает стабильность, расширяемость и удобство внедрения новых возможностей. Внедрение бонусов и редактора уровней значительно усилило реиграбельность проекта. В дальнейшем развитие может продолжиться через улучшение поведения вражеского ИИ и расширение возможностей пользовательского редактирования карт.

**Список литературы:**

1. Радченко, С. В. Программирование компьютерных игр. – СПб.: БХВ-Петербург, 2017. – 512 с.
2. Шварцман, М. Разработка игр на Java. Практическое руководство. – М.: Вильямс, 2014. – 382 с.
3. Нистром, Р. Паттерны проектирования в разработке игр. – М.: Питер, 2021. – 354 с.
4. Поляков, Н. В. “Алгоритмы обнаружения столкновений в двумерных играх.” // Информационные технологии, 2020, №7, с. 54–62.

