

Один рыцарь, сто монет и немного кода: как я изучала игровые движки на примере собственного 2D платформера

Автор: Рязанова Вероника Андреевна, 10 класс, Санкт-Петербургская гимназия «АЛЬМА-МАТЕР»

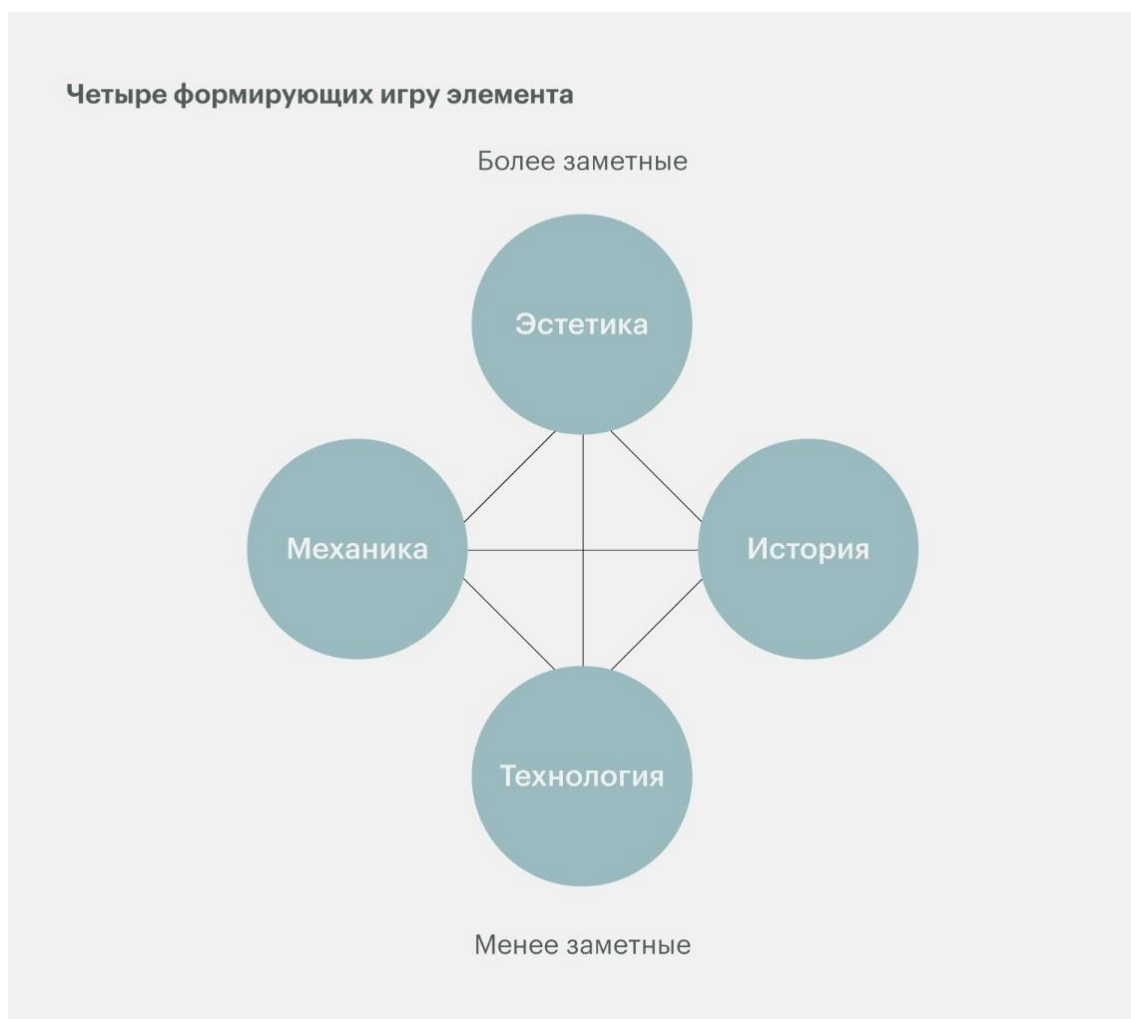
Вместо того чтобы просто играть, я решила создать свою игру

Каждый, кто хоть раз засиживался допоздна за Minecraft, GTA V или Genshin Impact, наверняка ловил себя на мысли: «А как они это сделали? И смогу ли я сам?». Обычно кажется, что разработка игр — это что-то невероятно сложное, доступное только взрослым программистам в больших компаниях. Но так ли это на самом деле?

Я решила проверить гипотезу: даже школьник с ограниченным опытом в программировании способен создать полноценную, красивую и увлекательную игру, если правильно выбрать инструменты и соблюдать основные законы гейм-дизайна. Чтобы доказать это, я не просто изучила теорию, а разработала собственный 2D-платформер — с рыцарем, монетами, скелетами, лабиринтом и принцессой в замке.

А потом я дала опробовать эту игру одноклассникам и спросила их мнение, проведя опрос. Результаты оказались даже лучше, чем я ожидала.

Гейм-дизайн — это не просто «придумать идею»



Первое, что я поняла в ходе работы: хорошая игра рождается не из случайной идеи, а из продуманной системы. В своей книге «Гейм-дизайн» Джесси Шелл сравнивает игру с четырьмя ножками стула. Если убрать хотя бы одну, конструкция развалится.

Первая ножка — механика. Это правила: как далеко прыгает герой, сколько урона наносит меч, сколько монет нужно для победы.

Вторая — история. Даже в простом платформере должен быть смысл: рыцарь спасает принцессу, а не просто бежит по экрану.

Третья — эстетика. Графика, звук, анимация — всё то, что создаёт настроение.

И четвёртая — технология, то есть игровой движок и код, которые приводят всё в движение.

Я убедилась в этом на собственном опыте. Когда я меняла всего один параметр — например, количество монет, необходимых для открытия замка, — игра начинала ощущаться совершенно по-другому. Если монет слишком мало, игроку неинтересно исследовать уровень; если слишком много — прохождение превращается в скучную рутину. Гейм-дизайн — это постоянный поиск баланса.

Три ведущих игровых движка, которые сегодня доминируют на рынке

Следующим шагом стал выбор инструмента. Сегодня существует десятки игровых движков, но три из них используют чаще всего: Unity, Unreal Engine и Godot. Каждый обладает своими сильными и слабыми сторонами.

Unreal Engine — это выбор профессионалов, которые гонятся за фотореалистичной графикой. На нём делают такие хиты, как Fortnite и Hellblade II. Однако освоить Unreal Engine новичку очень непросто, к тому же он требует мощного компьютера, а в случае коммерческого успеха нужно отчислять 5% дохода разработчикам.

Godot Engine — полная противоположность. Он полностью бесплатный, с открытым исходным кодом и очень простой в изучении. Godot идеально подходит для 2D-игр, но его трёхмерные возможности пока заметно скромнее, чем у конкурентов, а найти готовые уроки на русском языке бывает непросто.

Unity оказался золотой серединой. Он достаточно прост для начинающего разработчика, но при этом позволяет создавать и 2D, и 3D-проекты высокого уровня. Огромный магазин готовых ассетов и тысячи бесплатных уроков сделали своё дело: я оценила Unity в 4,75 балла из 5 по всем ключевым критериям. Именно на нём я и решила делать свою игру.

Сравнение популярных игровых движков по ключевым критериям

Критерии	Игровые движки		
	Unity	Unreal Engine	Godot Engine
Бесплатное использование	5	5	5
Средний порог вхождения	4	5	3
Количество бесплатных и платных дополнительных ресурсов	5	2	4
Уроки, курсы, документация	5	2	4
Среднее значение	4,75	3,5	4

Взгляд изнутри игровой механики: герой, монстр и награда

Когда движок был выбран, я села за дизайн-документ — это такой «чертёж» будущей игры, где прописано всё до мелочей. Моя история звучала так: злые силы заточили принцессу в замке, и только отважный рыцарь может её спасти. Но просто так войти в замок нельзя: сначала нужно собрать сто монет, которые снимают магическую защиту с ворот.

Главный герой управляется с помощью клавиш WASD и мыши. У него есть пять сердец здоровья, и при столкновении с врагом он теряет половину сердца. Я запрограммировала двух противников: медленного зелёного слайма, который погибает от одного удара, и более опасного скелета, способного атаковать издалека и выдерживающего два попадания. За победу над каждым врагом выпадают монеты — это связывает боевую систему с главной целью игры.

Особую гордость у меня вызывает механика тайника. Рыцарь может найти на уровне специальный сундук и временно положить туда часть монет. Зачем это нужно? Если герой погибнет, монеты из инвентаря теряются, а те, что лежат в тайнике, — сохраняются. Это добавляет игре стратегическую глубину: можно рискнуть и носить все монеты с собой, а можно перестраховаться и спрятать их.

Тысячи пикселей, один кадр: процесс создания игры

Всех персонажей и объекты я рисовала сама в программе Aseprite. Это специальный редактор для пиксель-арта, и он оказался невероятно удобным. Я сознательно ограничила цветовую палитру 64 цветами — так спрайты занимают меньше памяти и выглядят стильно, в едином ключе.

Самой сложной оказалась анимация рыцаря. Только цикл бега состоит из десяти кадров, а ведь есть ещё атака, получение урона и состояние покоя. Для слайма я сделала всего три кадра анимации — это сразу даёт понять игроку, что перед ним простой и медленный противник, имеющий дальнюю атаку огненными шарами. Скелет же получил более сложное поведение и анимацию ближнего удара.





Когда я впервые соединила код и спрайты и рыцарь побежал по экрану, управляемый моими пальцами, это было невероятное чувство. Все эти часы в Aseprite и десятки строчек кода вдруг превратились в живого персонажа.

Тестирование на реальных игроках: урок информатики как игровая лаборатория

Теория теорией, но настоящая проверка началась, когда я принесла игру в школу. На уроке информатики одноклассники сели за компьютеры, и я попросила их просто поиграть — без подсказок и инструкций. Мне было важно понять, разберётся ли человек в управлении с первого взгляда, интересно ли ему и что именно он захочет изменить.

Забегая вперёд: игра запустилась на всех компьютерах без единого сбоя, и никто не спрашивал, какие кнопки нажимать. Это был первый хороший знак.

После тестирования я провела анонимный опрос среди 63 школьников в возрасте от 11 до 18 лет. Результаты превзошли все ожидания. На вопрос «Понятно ли, что делать в игре?» утвердительно ответили 95,2% респондентов. Спрашиваю дальше: «Нравится ли графика?» — и снова 96,8% выбирают вариант «очень круто, хочется рассматривать». «Было ли скучно?» — 95,2% сказали, что им было интересно и они хотели бы поиграть ещё.

И под конец я попросила поставить игре школьную оценку от двойки до пятёрки. 95,2% участников поставили пятёрку со словами «Класс! Советую всем». Остальные — четвёрку. Двоек и троек не было вообще.

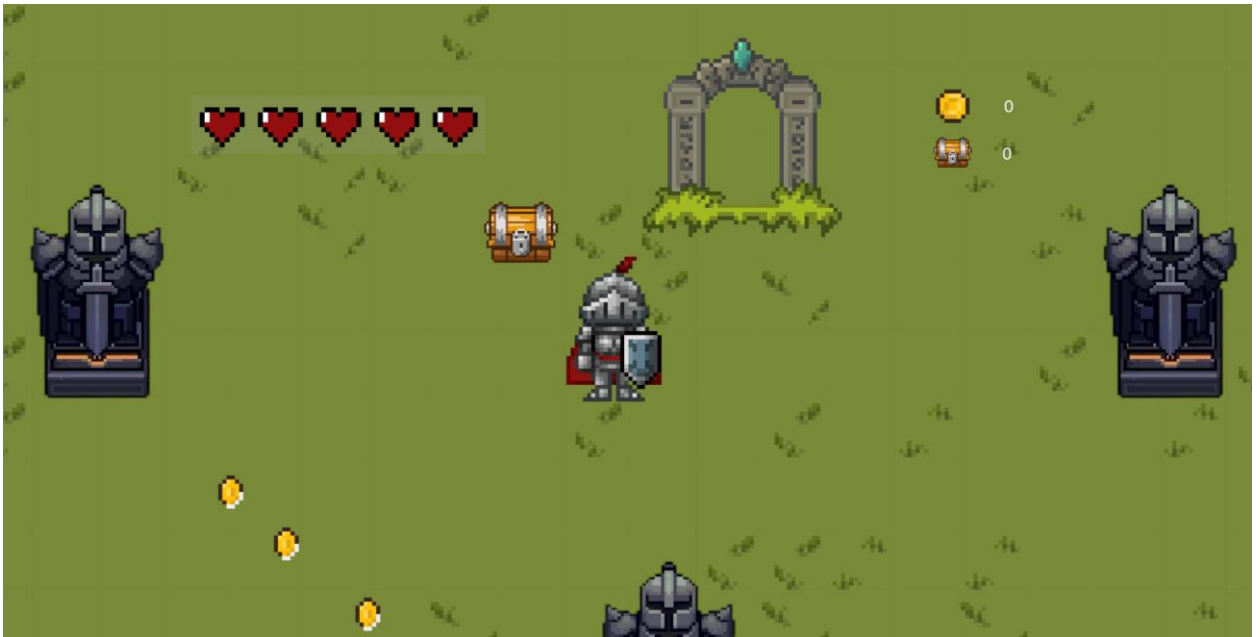


Особенно приятно было слышать, когда ребята удивлялись, узнав, что все спрайты нарисованы мной вручную. Для них это стало отдельным открытием.

Что в итоге и почему это важно

Моя гипотеза полностью подтвердилась. Современные игровые движки, в частности Unity, действительно позволяют начинающему разработчику создать работающий, красивый и увлекательный продукт. Для этого не нужно быть гением программирования — достаточно освоить базовые принципы гейм-дизайна, запастись терпением и не бояться экспериментировать.

Конечно, моя игра — это не GTA V и не Genshin Impact. Но она выполняет свою главную задачу: в неё интересно играть, она понятна с первого взгляда и дарит игроку ощущение победы, когда рыцарь наконец открывает замок и спасает принцессу.



У этого проекта есть будущее. Я планирую добавить новые уровни, записать звуковое сопровождение и, возможно, портировать игру на мобильные устройства. Но самое главное — теперь я точно знаю, что создание игр доступно каждому, кто готов потратить время и подойти к делу с умом.

А вы попробуйте. Возможно, ваша первая игра окажется лучше, чем вы думаете.

Коротко о том, что я прочитала и изучила

В работе я опиралась на книгу Джесси Шелла «Гейм-дизайн», официальную документацию Unity, обзоры движков Unreal Engine и Godot, на статьи с Habr, Skillbox и DTF, а также на свою исследовательскую работу, где вы можете более подробно ознакомиться с процессом создания игры. Отдельное спасибо одноклассникам, которые честно тестировали игру и помогли сделать её лучше.

Ссылка на исследовательскую работу: <https://docs.google.com/document/d/1xx-252p4fNDxzNPKpq3AAR2HXtqGDVP-/edit?usp=drivesdk&oid=101248204463932429683&rtpof=true&sd=true>

Список литературы

1. Джесси Шелл Геймдизайн [Электронный ресурс]. Режим доступа: <https://flibusta.su/book/8199-geymdizayn-kak-sozdat-igru-v-kotoruyu-budut-igrat-vse/read/> (дата обращения: 18.12.2024)
2. Значение гейм-дизайна для успеха игры [Электронный ресурс]. Режим доступа: <https://itanddigital.ru/bloghrconsulting/tpost/18uaa9xms1-geimdizainer-sozdanie-igr-kotorie-porazh>(дата обращения: 18.01.2025)
3. Игровой движок [Электронный ресурс]. Режим доступа: <https://bigenc.ru/c/igrovoi-dvizhok-6bбес7> (дата обращения:29.12.2024)

4. Игровой движок Unity: почему его выбирают [Электронный ресурс]. Режим доступа: <https://itstan.ru/programmirovanie/igrovoy-dvizhok-unity-pochemu-ego-vybirayut.html> (дата обращения: 29.12.2024)
5. Как создавали GTA 5. Внедрили работающую биржу, беседовали с агентами ФБР под прикрытием [Электронный ресурс]. Режим доступа: <https://cyber.sports.ru/games/blogs/3170999.html> (обращения: 04.03.2025)
6. Семь этапов создания игры: от концепта до релиза [Электронный ресурс]. Режим доступа: <https://habr.com/ru/companies/miip/articles/308286/> (обращения: 11.01.2025)
7. Скриптинг анимации (Legacy) // Unity Documentation [Электронный ресурс]. Режим доступа: <https://docs.unity3d.com/ru/530/Manual/AnimationScripting.html> (дата обращения: 25.01.2025)
8. Чем хорош Unity? [Электронный ресурс]. Режим доступа: <https://cubiq.ru/dvizhok-unity/> (дата обращения: 24.12.2024)
9. Что такое геймдизайн? [Электронный ресурс]. Режим доступа: <https://sky.pro/wiki/gamedev/chto-takoe-gejmdizajn/> (дата обращения: 15.12.2024)
10. Что такое геймдизайн? [Электронный ресурс]. Режим доступа: <https://skillbox.ru/media/gamedev/chto-takoe-geymdizayn/> (дата обращения: 15.12.2024)
11. Что такое Unity3D? [Электронный ресурс]. Режим доступа: <https://habr.com/ru/articles/161463/> (дата обращения: 24.12.2024)
12. Этапы разработки компьютерных игр [Электронный ресурс]. Режим доступа: <https://gdjob.pro/stati/soiskatelyam/etapy-razrabotki-kompyuternykh-igr/> (обращения: 11.01.2025)
13. Этапы создания компьютерной игры [Электронный ресурс]. Режим доступа: <https://dzen.ru/a/XU-nUT9UhwCtDtD7> (обращения: 17.01.2025)
14. Что такое Godot Engine и как работает этот движок [Электронный ресурс]. Режим доступа: <https://skillbox.ru/media/gamedev/chto-takoe-godot-engine-i-kak-rabotaet-etot-dvizhok/> (дата обращения: 26.02.2025)
15. Made with Godot [Электронный ресурс]. Режим доступа: <https://godotengine.org/showcase/> (обращения: 26.02.2025)
16. Ibodullayev Sardor Nasriddin O'g'li, Sodiqjonov Yaxyobek Sobirjon O'g'li Stages and methods of fps game development based on virtual reality technology // ymj. 2023. №6. URL: <https://cyberleninka.ru/article/n/stages-and-methods-of-fps-game-development-based-on-virtual-reality-technology> (дата обращения: 17.02.2025)

17. The evolution of video gaming and content consumption [Электронный ресурс]. Режим доступа: <https://www.pwc.com/us/en/industry/entertainmentmedia/publications/assets/pwcvideo-gaming-and-content-consumption.pdf> (дата обращения: 20.01.2025)
18. Unreal Engine: полный обзор движка, версий и технологий [Электронный ресурс]. Режим доступа: <https://education.yandex.ru/journal/unreal-engine> (дата обращения: 20.01.2025)
19. Unreal Engine // Contented. Media [Электронный ресурс]. Режим доступа: <https://media.contented.ru/glossary/unreal-engine/> (дата обращения: 20.01.2025)