***Аналитический отчёт: Выбор движка Unity для образовательной игры в рамках Проектного Практикума.***

При разработке образовательной игры ключевыми критериями выбора движка являлись:

* **Простота разработки** – удобные инструменты для быстрого прототипирования.
* **Основной язык программирования движка C#** – в нашем учебном плане предусмотрено изучение языка программирования C#, который является основным языком программирования в Unity. Благодаря использованию уже знакомых конструкций и опыту работы в C# в других проектах можно не тратить дополнительные ресурсы на освоение и понимание языка, что замедлило бы процесс.
* **Поддержка 2D и 3D** – гибкость в выборе формата контента.
* **Наличие образовательного контента** – встроенные решения для геймификации обучения.
* **Сообщество и документация** – доступность учебных материалов и готовых решений.
* **Возможная Кроссплатформенность** – возможность в будущем реализовать запуск образовательной игры не только на ПК, но и на мобильных устройствах, и в браузере.

Преимущества Unity для образовательных игр:

* **Удобство разработки** – в Unity есть визуальные редактор, который обеспечивает простое создание сцен и интерфейсов.
* **Asset Store -** готовые 3D-модели, анимации, плагины для обучения (например, системы диалогов, квестов).
* **Большое количество обучающей документации -** официальные туториалы по созданию игр разных жанров, готовые решения на форумах (Unity Answers, Reddit).

Альтернативы и почему они не подошли:

|  |  |  |  |
| --- | --- | --- | --- |
| **Движок** | **Плюсы** | **Минусы** | **Почему не выбрал** |
| Unreal Engine | Высокое качество графики, Blueprints | Сложность для 2D, высокий порог входа | Избыточен для образовательного проекта |
| Godot | Лёгкий, открытый исходный код | Меньше готовых ассетов, слабая 3D-поддержка | Меньше документации под образовательные задачи |
| Construct 3 | Простота, работа в браузере | Ограниченная функциональность | Не подходит для сложных механик |