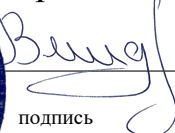


**ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«ИННОПРОГ»**

УТВЕРЖДАЮ



Генеральный директор ООО «Иннопрог»



/ Венедиктов Р.В.

подпись

инициалы, фамилия

30 декабря 2025 г.

Приказ об утверждении программы
от 30 декабря 2025 г. № ОБР-5

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

«Unreal Engine»

Направленность программы:	техническая
Уровень освоения:	разноуровневый
Возраст обучающихся:	дети от 12 лет и взрослые
Трудоемкость:	800 академических часов
Форма обучения:	очная, с применением электронного обучения и дистанционных образовательных технологий
Режим реализации:	10 месяцев (40 учебных недель), рекомендуемая нагрузка – 20 академических часов в неделю
Язык обучения:	русский

Иннополис, 2026

Содержание

1	Пояснительная записка	3
1.1	Наименование программы	3
1.2	Нормативно-правовые основания разработки программы . .	3
1.3	Структура образовательной программы	4
1.4	Направленность и уровень программы	5
1.5	Актуальность программы	5
1.6	Цель реализации программы	5
1.7	Задачи программы	6
1.8	Адресат программы	7
1.9	Порядок приема, зачисления, перевода и прекращения обучения	7
1.10	Форма обучения, формы организации занятий и язык обучения	7
1.11	Объем, срок освоения и режим занятий	8
1.12	Документ об обучении	8
1.13	Профориентационная и практическая направленность	8
1.14	Планируемые результаты освоения программы	9
1.14.1	Перечень формируемых предметных результатов . .	9
1.14.2	Планируемые результаты обучения	10
1.14.3	Метапредметные и личностные результаты	12
2	Содержание программы	13
2.1	Учебный план	13
2.2	Календарный учебный график	16
2.3	Рабочие программы модулей	17
2.3.1	Модуль 1. Базовые принципы создания игр	18
2.3.2	Модуль 2. Знакомство с Unreal Engine 5	20
2.3.3	Модуль 3. Визуальное программирование Blueprint .	22
2.3.4	Модуль 4. Разработка базовых механик игры	25
2.3.5	Модуль 5. Инструменты моделирования в редакторе, источники света, базовый шейдер	27
2.3.6	Модуль 6. Создание кастомного персонажа, настройка управления и камеры	30

2.3.7	Модуль 7. Взаимодействие объектов через систему коллизий	32
2.3.8	Модуль 8. Основы создания ИИ с использованием State Tree	34
2.3.9	Модуль 9. Основы работы с UMG и User Widget . . .	36
2.3.10	Модуль 10. Профилирование и оптимизация в Unreal Engine	39
2.3.11	Модуль 11. Проект «Collect the coin»	41
2.3.12	Модуль 12. Проект «Шутер с летающим дроном» . .	43
2.3.13	Модуль 13. Проектная практика и портфолио. Загрузка игр на itch и другие площадки	46
2.3.14	Модуль 14. Промежуточная аттестация по завершении Программы	49
2.4	Проектная практика и консультации	51
3	Организационно-педагогические условия реализации программы	52
3.1	Общие условия реализации	52
3.2	Материально-технические условия	52
3.3	Электронная информационно-образовательная среда	53
3.4	Кадровые условия	53
3.5	Условия для несовершеннолетних обучающихся	54
3.6	Условия для обучающихся с ограниченными возможностями здоровья	54
3.7	Учебно-методическое обеспечение	54
3.8	Применение электронного обучения и дистанционных образовательных технологий	55
3.9	Информационная безопасность, персональные данные и этика	55
3.10	Локальные процедуры обучения	56
3.11	Обновление программы	56
4	Оценка качества освоения программы	57
4.1	Общие положения	57
4.2	Текущий контроль успеваемости	57
4.3	Промежуточная аттестация	57
4.4	Пересдача, апелляция и подтверждение самостоятельности .	58

4.5	Допуск к промежуточной аттестации по завершении Программы	59
4.6	Промежуточная аттестация по завершении Программы . . .	59
4.6.1	Примерная тематика проектных работ	59
4.6.2	Обязательные требования к итоговому проекту . . .	60
4.6.3	Порядок проведения промежуточной аттестации по завершении Программы	61
4.6.4	Критерии оценивания итогового проекта	61
4.7	Внутренняя и внешняя оценка качества реализации программы	62
5	Методические материалы	64
5.1	Методические указания для обучающихся	64
5.2	Методические указания для преподавателей и проверяющих	64
5.3	Рекомендуемая литература и ресурсы	64
	Приложение 1. Матрица формирования компетенций	65
	Приложение 2. Оценочные материалы промежуточной аттестации .	66
	Приложение 3. Оценочные материалы промежуточной аттестации по завершении Программы	67
	Приложение 4. Перечень локальных нормативных актов, на которые должна опираться реализация программы	68
	Приложение 5. Лист актуализации программы	69
	Приложение 6. Требования к репозиторию, README и демонстрации	70
	Приложение 7. Дорожная карта подготовки итогового проекта	71
	Приложение 8. Контрольный лист соответствия требованиям программы	72

1. Пояснительная записка

1.1. Наименование программы

Дополнительная общеобразовательная общеразвивающая программа технической направленности «Unreal Engine» (далее – Программа) разработана для реализации обществом с ограниченной ответственностью «Иннопрог» в сфере дополнительного образования детей и взрослых.

Программа утверждена и введена в действие с 30 декабря 2025 года приказом ООО «ИННОПРОГ» от 30.12.2025 № ОБР-5 «О дополнении перечня образовательных программ».

1.2. Нормативно-правовые основания разработки программы

Программа разработана в соответствии с:

- Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- приказом Министерства просвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- постановлением Правительства Российской Федерации от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- постановлением Правительства Российской Федерации от 18.09.2020 № 1490 «О лицензировании образовательной деятельности»;
- постановлением Правительства Российской Федерации от 15.09.2020 № 1441 «Об утверждении Правил оказания платных образовательных услуг»;
- Федеральным законом от 27.07.2006 № 152-ФЗ «О персональных данных»;
- постановлением Главного государственного санитарного врача Российской Федерации от 02.06.2026 № 19, которым утверждены

санитарные правила СП 2.4.2.4283-26 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (применяются с 01.09.2026);

- ГОСТ Р 7.0.97-2025 «Система стандартов по информации, библиотечному и издательскому делу. Организационно-распорядительная документация. Требования к оформлению документов» в части оформления организационно-распорядительных реквизитов;
- локальными нормативными актами ООО «Иннопрог», регуливающими разработку и утверждение дополнительных общеобразовательных программ, прием и зачисление, организацию образовательного процесса, обучение по индивидуальному учебному плану, применение электронного обучения и дистанционных образовательных технологий, текущий контроль и промежуточную аттестацию, выдачу документов об обучении, защиту персональных данных, охрану здоровья обучающихся и оказание платных образовательных услуг.

Программа реализуется за пределами федеральных государственных образовательных стандартов и федеральных государственных требований. Она не относится к дополнительным профессиональным программам, не является программой профессионального обучения и не предусматривает присвоение профессии, специальности, квалификации, квалификационного разряда или права на ведение нового вида профессиональной деятельности.

1.3. Структура образовательной программы

Программа представляет собой комплекс основных характеристик образования, организационно-педагогических условий, оценочных и методических материалов. В состав Программы включены:

- пояснительная записка, цель, задачи, адресат, условия приема, объем, срок и режим освоения;
- планируемые предметные, метапредметные и личностные результаты;
- учебный план, календарный учебный график и рабочие программы модулей;

- организационно-педагогические условия, формы контроля, оценочные и методические материалы;
- описание проектной практики и промежуточной аттестации по завершении Программы в форме защиты итогового проекта.

1.4. Направленность и уровень программы

Программа имеет техническую направленность и разноуровневый характер. Стартовый уровень обеспечивает освоение терминов и базовых операций; базовый – самостоятельное решение типовых задач; углубленный – интеграцию нескольких технологий в проекте. Индивидуальная траектория определяется входной диагностикой, темпом освоения, консультациями и уровнем проектных заданий.

1.5. Актуальность программы

Актуальность Программы обусловлена устойчивым интересом к разработке игр, интерактивных 3D-приложений, симуляторов, образовательных визуализаций и мультимедийных продуктов. Unreal Engine 5 используется как среда, в которой можно быстро перейти от идеи к работающему прототипу, проверить механику, подготовить демонстрацию, собрать портфолио и продолжить развитие проекта в индивидуальной или командной разработке.

Для начинающего разработчика особенно важна связь между игровой идеей и ее технической реализацией. Игровой прототип требует понимания правил, состояния игры, поведения объектов, пользовательского интерфейса, обратной связи, производительности и подготовки сборки. Программа отвечает этой потребности через последовательное освоение инструментов Unreal Engine 5 и выполнение проектных заданий, которые накапливаются в итоговый портфолио-результат.

1.6. Цель реализации программы

Цель Программы – формирование у детей от 12 лет и взрослых устойчивого интереса, предметных результатов и опыта проектной деятельности в области проектирования игровых механик, визуального

программирования Blueprint, создания интерактивных 3D-сцен, интерфейсов и игровых прототипов, развитие цифровой грамотности, алгоритмического мышления, самостоятельности, культуры безопасной и ответственной работы с информационными технологиями.

1.7. Задачи программы

Задачами Программы являются:

- сформировать представление о базовых принципах создания игр, игровых жанрах, core loop, состояниях игры и критериях качества прототипа;
- научить работать с интерфейсом Unreal Engine 5, структурой проекта, уровнями, actors, components и Content Browser;
- освоить визуальное программирование Blueprint, события, переменные, функции, компоненты, интерфейсы, dispatchers и отладку логики;
- научить разрабатывать базовые игровые механики, включая сбор предметов, счет, таймеры, здоровье, урон, стрельбу, победу и поражение;
- освоить инструменты моделирования в редакторе, blockout, источники света, базовые материалы и простой шейдер;
- научить создавать кастомного персонажа или Pawn, настраивать управление, движение, камеру и ограничения;
- сформировать навыки настройки коллизий, overlap, hit-событий, trace-проверок и правил взаимодействия объектов;
- познакомить с базовыми принципами создания AI с использованием State Tree, состояний, условий и переходов;
- научить создавать пользовательский интерфейс средствами UMG и User Widget;
- сформировать базовые навыки профилирования, оптимизации, подготовки сборки и проверки производительности проекта;
- организовать разработку учебных проектов «Collect the coin» и «Шутер с летающим дроном»;
- сформировать навыки подготовки портфолио, README, скриншотов, видео, страницы проекта и публикации игры на itch.io или другой площадке;

- обеспечить подготовку и проведение промежуточной аттестации по завершении Программы в формате защиты проекта.

1.8. Адресат программы

Программа предназначена для детей от 12 лет и взрослых, заинтересованных в освоении направления «разработка игр в Unreal Engine 5». Наличие среднего профессионального или высшего образования не требуется. При приеме несовершеннолетнего договор заключается с его законным представителем либо с участием законного представителя в порядке, установленном законодательством и локальными актами организации.

Рекомендуемые начальные условия: уверенная работа с компьютером и браузером, умение создавать и сохранять файлы, базовая математическая и информационная грамотность, готовность регулярно выполнять практические задания. Входная диагностика используется для подбора стартового уровня и не является конкурсным испытанием.

1.9. Порядок приема, зачисления, перевода и прекращения обучения

Прием осуществляется на основании заявления, договора об образовании и документов, предусмотренных локальными актами организации. Зачисление оформляется приказом. Перевод на индивидуальный учебный план, изменение срока, восстановление и прекращение образовательных отношений осуществляются в порядке, установленном законодательством, договором и локальными нормативными актами ООО «Иннопрог».

1.10. Форма обучения, формы организации занятий и язык обучения

Форма обучения – очная, с применением электронного обучения и дистанционных образовательных технологий. Занятия проводятся в синхронном и асинхронном форматах и включают объяснение нового материала, демонстрации, практикумы, самостоятельную работу, консультации, текущий контроль, промежуточную аттестацию по модулям и проектную практику. Язык обучения – русский. Академический час равен 45 минутам.

1.11. Объем, срок освоения и режим занятий

Трудоемкость Программы составляет 800 академических часов, включая теоретические и практические занятия, самостоятельную работу, текущий контроль, промежуточную аттестацию по модулям, проектную практику и защиту итогового проекта по завершении Программы.

Рекомендуемый срок освоения составляет 10 месяцев (40 учебных недель) при средней нагрузке 20 академических часов в неделю. Конкретное расписание может предусматривать различное соотношение занятий с преподавателем и самостоятельной работы. Допускается обучение по индивидуальному учебному плану, в том числе ускоренное, при сохранении общего объема и достижении планируемых результатов.

1.12. Документ об обучении

Обучающемуся, освоившему Программу, выполнившему обязательные задания и успешно прошедшему промежуточную аттестацию по завершении Программы в форме защиты итогового проекта, выдается сертификат об обучении по образцу, самостоятельно установленному ООО «Иннопрог».

Сертификат является документом об обучении, подтверждает факт и объем освоения дополнительной общеобразовательной общеразвивающей программы, но не является документом об образовании и (или) о квалификации, не присваивает квалификацию и не заменяет диплом о профессиональной переподготовке.

Лицу, освоившему часть Программы, отчисленному до ее завершения или не прошедшему аттестацию, по его заявлению может быть выдана справка об обучении или о периоде обучения по образцу, установленному организацией.

1.13. Профориентационная и практическая направленность

Содержание Программы соотнесено с типовыми современными задачами по направлению «разработка игр в Unreal Engine 5» исключительно как профориентационный и содержательный ориентир. Профессиональные стандарты могут использоваться при актуализации тем и критериев качества проектов, однако Программа не заявляет подготовку к выполнению трудовых

функций в полном объеме и не устанавливает уровень квалификации выпускника.

1.14. Планируемые результаты освоения программы

1.14.1. Перечень формируемых предметных результатов

Таблица 1 – Формируемые предметные результаты

Код	Наименование предметного результата
ПР-1	Проектировать игровой прототип, формулировать игровую цель, core loop, правила, условия победы и поражения, ограничения и критерии приемки.
ПР-2	Создавать и сопровождать проекты Unreal Engine 5, уровни, actors, components, assets, материалы, источники света и структуру Content Browser.
ПР-3	Разрабатывать игровую логику средствами Blueprint, использовать события, переменные, функции, компоненты, интерфейсы, dispatchers и средства отладки.
ПР-4	Реализовывать базовые игровые механики, включая сбор предметов, счет, таймер, здоровье, урон, стрельбу, победу, поражение и перезапуск уровня.
ПР-5	Создавать читаемую 3D-сцену и визуальную основу игрового уровня с использованием blockout, Modeling Tools, освещения и базовых материалов.
ПР-6	Создавать кастомного персонажа или Pawn, настраивать управление, движение, камеру и взаимодействие игрока с уровнем.
ПР-7	Настраивать взаимодействие объектов через collision presets, overlap, hit-события, trace-проверки и правила реакции игровых объектов.
ПР-8	Разрабатывать простое поведение игровых объектов через State Tree и создавать пользовательский интерфейс средствами UMG и User Widget.

Код	Наименование предметного результата
ПР-9	Проводить базовое профилирование и оптимизацию проекта, готовить сборку, проверять производительность и воспроизводимость демонстрации.
ПР-10	Оформлять игровой проект как портфолио-результат, готовить README, скриншоты, видео, страницу на itch.io или другой площадке и защиту проекта.

1.14.2. Планируемые результаты обучения

В результате освоения Программы обучающийся должен:

знать:

- базовые принципы создания игр, структуру игрового цикла, понятия core loop, цели игрока, победы, поражения и обратной связи;
- назначение основных элементов Unreal Engine 5: проекта, уровня, actor, component, pawn, character, player controller, game mode, content browser;
- принципы визуального программирования Blueprint, событийной модели, переменных, функций, компонентов, интерфейсов и отладки;
- подходы к проектированию базовых игровых механик, включая сбор предметов, счет, таймер, здоровье, урон, стрельбу и завершение уровня;
- назначение blockout, инструментов моделирования редактора, источников света, базовых материалов и простого шейдера;
- принципы настройки персонажа, управления, камеры и пользовательского опыта игрока;
- основы настройки коллизий, overlap, hit-событий, collision channels и trace-проверок;
- назначение State Tree для описания поведения объектов через состояния, переходы и условия;
- основы создания пользовательского интерфейса в UMG и связи виджетов с игровым состоянием;
- базовые подходы к профилированию, оптимизации, подготовке сборки, README и публикации проекта.

уметь:

- создавать и настраивать проект Unreal Engine 5, уровни, структуру папок, assets и простые сцены;
- описывать игровую идею в виде правил, сценария игрока, критериев приемки и чек-листа проверки;
- создавать Blueprint-классы, использовать события, переменные, функции, интерфейсы, dispatchers и компоненты;
- реализовывать базовые игровые механики и связывать их с UI, коллизиями, персонажем и состояниями игры;
- создавать простую 3D-сцену с blockout, светом, материалами и визуальными подсказками для игрока;
- создавать кастомного персонажа или Pawn, настраивать управление, движение и камеру;
- настраивать правила взаимодействия объектов через collision presets, overlap, hit events и trace;
- создавать простое поведение объекта или противника через State Tree;
- создавать User Widget, HUD, стартовый экран, экран победы, поражения и перезапуска;
- профилировать учебный проект, исправлять типовые проблемы производительности, готовить сборку, README, скриншоты, видео и страницу проекта.

владеть навыками:

- практической разработки игровых прототипов в Unreal Engine 5;
- проектирования и документирования игровых механик;
- визуального программирования и отладки Blueprint-логики;
- организации структуры проекта, assets, уровней и демонстрационных материалов;
- настройки персонажа, управления, камеры, коллизий, UI и простого AI;
- подготовки проекта к проверке, сборке, публикации и защите;
- выполнения итогового проекта от постановки задачи до портфолио-публикации и защиты результата.

1.14.3. Метапредметные и личностные результаты

По завершении Программы обучающийся способен:

- ставить учебную задачу, разбивать ее на этапы, планировать сроки и контролировать продвижение;
- искать, сопоставлять и критически оценивать техническую информацию, документацию и примеры;
- объяснять выбранное решение, воспринимать обратную связь, аргументированно вносить изменения и представлять результат;
- соблюдать правила командного взаимодействия, академической добросовестности и корректного указания источников;
- ответственно работать с персональными данными, учетными записями, секретами и пользовательской информацией;
- использовать генеративные инструменты и иные средства искусственного интеллекта как вспомогательный ресурс, проверяя результат и сохраняя личную ответственность за выполненную работу;
- проявлять самостоятельность, аккуратность, настойчивость при поиске ошибок и уважение к результатам чужого труда.

2. Содержание программы

2.1. Учебный план

Таблица 2 – Учебный план

№	Наименование модуля	Всего, час.	Лекц., час.	Практ., час.	Сам. раб., час.	ПА, час.	ИА, час.	Форма контроля
1	Базовые принципы создания игр	12	6	0	6	0	0	входной контроль
2	Знакомство с Unreal Engine 5	56	17	22	11	6	0	зачет
3	Визуальное программирование Blueprint	92	26	43	17	6	0	зачет
4	Разработка базовых механик игры	68	20	31	11	6	0	зачет
5	Инструменты моделирования в редакторе, источники света, базовый шейдер	68	20	31	11	6	0	зачет
6	Создание кастомного персонажа, настройка управления и камеры	80	23	40	11	6	0	зачет
7	Взаимодействие объектов через систему коллизий	56	17	25	8	6	0	зачет
8	Основы создания ИИ с использованием State Tree	56	17	25	8	6	0	зачет
9	Основы работы с UMG и User Widget	48	12	21	9	6	0	зачет
10	Профилирование и оптимизация в Unreal Engine	48	12	21	9	6	0	зачет
11	Проект «Collect the coin»	56	6	39	11	0	0	зачет по проекту
12	Проект «Шутер с летающим дроном»	80	6	57	17	0	0	зачет по проекту

Продолжение таблицы 2

№	Наименование модуля	Всего, час.	Лекц., час.	Практ., час.	Сам. раб., час.	ПА, час.	ИА, час.	Форма контроля
13	Проектная практика и портфолио. Загрузка игр на itch и другие площадки	44	0	33	11	0	0	допуск к ИА
14	Промежуточная аттестация по завершении Программы	36	0	0	0	0	36	защита проекта
Итого		800	182	388	140	54	36	–

Промежуточная аттестация осуществляется по завершении модулей, определенных учебным планом. Промежуточная аттестация по завершении Программы проводится отдельно от промежуточной аттестации в форме защиты итогового аттестационного проекта. Трудоемкость промежуточной аттестации по завершении Программы составляет 36 академических часа.

2.2. Календарный учебный график

Начало обучения определяется по мере комплектования учебной группы. Учебные группы могут формироваться в течение всего календарного года. Период освоения Программы определяется календарным учебным графиком и расписанием. Ниже представлен рекомендуемый график при сроке обучения 10 месяцев (40 недель).

Таблица 3 – Календарный учебный график

№	Наименование модуля	Период обучения	Часы
1	Базовые принципы создания игр	1 неделя	12
2	Знакомство с Unreal Engine 5	2-4 недели	56
3	Визуальное программирование Blueprint	5-9 недели	92
4	Разработка базовых механик игры	10-12 недели	68
5	Инструменты моделирования в редакторе, источники света, базовый шейдер	13-15 недели	68
6	Создание кастомного персонажа, настройка управления и камеры	16-19 недели	80
7	Взаимодействие объектов через систему коллизий	20-22 недели	56
8	Основы создания ИИ с использованием State Tree	23-25 недели	56
9	Основы работы с UMG и User Widget	26-27 недели	48

Продолжение таблицы 3

№	Наименование модуля	Период обучения	Часы
10	Профилирование и оптимизация в Unreal Engine	28-29 недели	48
11	Проект «Collect the coin»	30-32 недели	56
12	Проект «Шутер с летающим дроном»	33-36 недели	80
13	Проектная практика и портфолио. Загрузка игр на itch и другие площадки	37-38 недели	44
14	Промежуточная аттестация по завершении Программы	39-40 недели	36
Итого			800

2.3. Рабочие программы модулей

В рабочих программах модулей определяются цели, результаты, тематическое содержание, формы организации занятий, самостоятельной работы и контроля. Перечень тем соотносится с фактической структурой учебных материалов и внутреннего контента направления «Unreal Engine-разработчик».

Проектная траектория программы. Программа построена так, чтобы каждый модуль добавлял новый слой к будущему портфолио. Обучающийся постепенно проходит путь от понимания игры как системы до защиты опубликованного проекта.

Таблица 4 – Проектная траектория программы

№	Этап	Модули	Результат этапа
1	Вход в профессию	модули 1–2	игровая идея, структура проекта, первый уровень в Unreal Engine 5.

Продолжение таблицы 4

№	Этап	Модули	Результат этапа
2	Базовая разработка	модули 3–7	Blueprint-логика, механики, персонаж, управление, камера, коллизии и визуальная среда.
3	Системы игры	модули 8–10	State Tree, UMG, профилирование и исправление типовых проблем проекта.
4	Проектный блок	модули 11–13	игра Collect the coin, шутер с летающим дроном, портфолио и публикация.
5	Защита результата	модуль 14	итоговый проект, сборка, демонстрация, объяснение архитектуры и ответы на вопросы.

Обучающийся может выполнять указанные проекты последовательно либо выбрать индивидуальную тему после согласования с преподавателем. Индивидуальная тема должна соответствовать результатам Программы, включать разработку в Unreal Engine 5, Blueprint-логику, игровой сценарий, пользовательское взаимодействие, демонстрационный уровень, проверяемый результат, README и материалы для защиты.

2.3.1. Модуль 1. Базовые принципы создания игр

Цель модуля: сформировать понимание игры как интерактивной системы, базовых понятий геймдизайна, структуры игрового цикла, роли прототипирования и требований к учебным проектам.

Объем модуля: 12 ч., в том числе лекционные занятия — 6 ч., практические занятия — 0 ч., самостоятельная работа — 6 ч., контроль — 0 ч.

Планируемые результаты по модулю:

- понимать назначение жанра, целевого действия игрока, core loop и правил игры;

- выделять игровую цель, условия победы и поражения, ограничения уровня и критерии приемки;
- описывать простую игровую идею в формате технического задания и чек-листа;
- понимать структуру программы, формы контроля, требования к проектам и портфолио.

Таблица 5 – Тематический план модуля 1. Базовые принципы создания игр

№	Тема	Часы	Содержание
1	Игра как интерактивная система	3	цель игрока, правила, обратная связь, состояние игры, победа и поражение.
2	Жанры, механики и core loop	3	жанровые ожидания, основной игровой цикл, мотивация игрока, повторяемое действие.
3	Игровая документация и критерии качества	3	краткое техническое задание, user story, критерии приемки, риски реализации.
4	Организация обучения и проектной траектории	3	модули, контрольные точки, репозиторий, демонстрационные материалы и требования к итоговой защите.

Содержание модуля: Модуль задает методическую основу всей программы. Обучающийся разбирает, из каких элементов состоит игра, как формулируется идея прототипа, чем механика отличается от визуального оформления, как фиксируются требования и как учебные задания постепенно складываются в портфолио Unreal Engine-разработчика.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление

кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- описать идею простого игрового прототипа с целью, правилами и условиями завершения;
- подготовить индивидуальный план проектной траектории и структуру папки курса.

Типовые ошибки, подлежащие разбору:

- игровая идея описана слишком общо и не имеет проверяемой цели;
- не определены условия победы, поражения или завершения уровня;
- не создана структура проекта и материалов обучения.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.2. Модуль 2. Знакомство с Unreal Engine 5

Цель модуля: сформировать базовые навыки работы с Unreal Engine 5, интерфейсом редактора, шаблонами проектов, уровнями, объектами, компонентами, assets и первичной организацией проекта.

Объем модуля: 56 ч., в том числе лекционные занятия — 17 ч., практические занятия — 22 ч., самостоятельная работа — 11 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- создавать, открывать и сохранять Unreal Engine 5-проект;
- ориентироваться в основных панелях редактора, viewport, Outliner, Details Panel и Content Browser;
- размещать actors, настраивать transform, components, камеры, источники света и простые материалы;

- оформлять учебную сцену с понятной структурой папок, README и демонстрацией запуска.

Таблица 6 – Тематический план модуля 2. Знакомство с Unreal Engine 5

№	Тема	Часы	Содержание
1	Установка и первый запуск Unreal Engine 5	6	Epic Games Launcher, выбор версии движка, создание первого проекта, проверка системных требований.
2	Интерфейс редактора	9	viewport, toolbar, Outliner, Details Panel, Content Browser, режимы просмотра и навигации.
3	Шаблоны проектов и структура Content	9	templates, maps, blueprints, materials, meshes, textures, правила именования и организации assets.
4	Actors, components и transform	9	размещение объектов, перемещение, поворот, масштабирование, mobility, базовые components.
5	Работа с уровнями	9	создание карты, сохранение уровня, basic blockout, расположение объектов и камер.
6	Импорт и использование assets	8	импорт моделей и текстур, проверка путей, настройка простых материалов, порядок хранения ресурсов.
7	Project Settings, Play In Editor и зачетная сцена	6	настройки проекта, play modes, проверка сцены, подготовка README, скриншотов и короткого видео.

Содержание модуля: Модуль переводит обучающегося от общей идеи игры к работе в редакторе Unreal Engine 5. Основным результатом модуля состоит в создании первой учебной сцены, где есть карта, объекты, камера, свет, простые материалы и воспроизводимый порядок запуска проекта.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;

- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- создать учебную сцену знакомства с Unreal Engine 5;
- оформить структуру Content, README, скриншоты и короткую демонстрацию запуска.

Типовые ошибки, подлежащие разбору:

- проект открывается только на компьютере автора из-за нарушенной структуры assets;
- карта не сохранена или не указана как рабочая сцена;
- отсутствует README и проверяемая инструкция запуска.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.3. Модуль 3. Визуальное программирование Blueprint

Цель модуля: сформировать навыки визуального программирования в Blueprint, включая события, переменные, функции, макросы, интерфейсы, компоненты, коммуникацию объектов и отладку логики.

Объем модуля: 92 ч., в том числе лекционные занятия — 26 ч., практические занятия — 43 ч., самостоятельная работа — 17 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- создавать Blueprint-классы и настраивать их свойства;

- использовать события, переменные, функции, таймеры, ветвления и циклы;
- организовывать взаимодействие объектов через ссылки, интерфейсы, dispatchers и components;
- отлаживать Blueprint-графы, структурировать логику и объяснять работу ключевых узлов.

Таблица 7 – Тематический план модуля 3. Визуальное программирование Blueprint

№	Тема	Часы	Содержание
1	Blueprint Class и Event Graph	9	создание класса, BeginPlay, Tick, Custom Event, exec flow, структура графа.
2	Переменные, типы и функции	9	boolean, integer, float, text, arrays, enums, structs, функции, входные и выходные параметры.
3	Flow Control и таймеры	9	Branch, Sequence, Delay, Timer, Gate, Do Once, loops и правила безопасного применения.
4	Функции, макросы и свернутые графы	9	декомпозиция логики, повторное использование, читаемость и ограничения макросов.
5	Ссылки, Casting и Blueprint Interfaces	9	получение references, проверка валидности, взаимодействие объектов без хрупких связей.
6	Event Dispatchers и Actor Components	9	событийная модель, вынесение повторяемой логики, настройка компонентов через Details Panel.
7	Gameplay Framework на уровне Blueprint	12	GameMode, Pawn, PlayerController, HUD, GameInstance и ответственность классов на обзорном уровне.
8	Blueprint Debugging	9	breakpoints, watch values, Print String, визуальная проверка потока выполнения и поиск ошибок.
9	Архитектура Blueprint-проекта	11	именование, комментарии, функции, компоненты, ограничение сложности Event Graph и подготовка к ревью.

№	Тема	Часы	Содержание
10	Промежуточная аттестация	6	интерактивный Blueprint-прототип с несколькими объектами, состояниями, счетом и завершением сценария.

Содержание модуля: Модуль является основой практической разработки в программе. Обучающийся осваивает визуальную логику Blueprint как главный инструмент создания механик без перехода к текстовому программированию, учится связывать объекты, выносить повторяемые действия в функции и компоненты, находить ошибки и поддерживать читаемость графов.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- создать интерактивную комнату с несколькими Blueprint-объектами;
- реализовать учебную механику со счетом, событиями и проверяемым завершением;
- подготовить схему связи Blueprint-классов для ревью.

Типовые ошибки, подлежащие разбору:

- вся логика размещена в Tick без необходимости;
- графы перегружены и не разделены на функции или компоненты;
- объекты связаны через случайные ссылки без проверки валидности;

- нет отладки и демонстрационного сценария проверки.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.4. Модуль 4. Разработка базовых механик игры

Цель модуля: сформировать навыки проектирования и реализации базовых игровых механик, включая сбор предметов, счет, таймер, интерактивные объекты, здоровье, условия победы, поражения и перезапуска уровня.

Объем модуля: 68 ч., в том числе лекционные занятия — 20 ч., практические занятия — 31 ч., самостоятельная работа — 11 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- проектировать простую игровую механику от цели игрока до проверки результата;
- создавать collectables, триггеры, кнопки, двери, препятствия и зоны завершения;
- реализовывать счет, таймер, здоровье, урон, feedback и перезапуск уровня;
- проверять механику по чек-листу и дорабатывать баланс учебного прототипа.

Таблица 8 – Тематический план модуля 4. Разработка базовых механик игры

№	Тема	Часы	Содержание
1	Игровая идея и core loop механики	6	цель игрока, повторяемое действие, награда, риск, критерии завершения.
2	Collectables и pickup-объекты	9	предметы для сбора, overlap, изменение состояния, удаление объекта, начисление очков.

№	Тема	Часы	Содержание
3	Счет, таймер и прогресс	9	переменные состояния, отображение прогресса, ограничения времени, проверка выполнения цели.
4	Кнопки, двери, рычаги и переключатели	9	интерактивные объекты, события, открытие прохода, включение объектов и обратная связь.
5	Здоровье, урон и препятствия	9	health, damage, опасные зоны, столкновения, восстановление и базовая реакция игрока.
6	Победа, поражение и restart	8	win condition, lose condition, restart level, переход между состояниями игры.
7	Feedback для игрока	6	звук, свет, простые VFX, текстовые подсказки и визуальная читаемость действия.
8	Баланс и проверка механик	6	сложность, темп, крайние случаи, тестовый маршрут и список багов.
9	Промежуточная аттестация	6	мини-прототип с базовой механикой, понятной целью, счетом и завершением уровня.

Содержание модуля: Модуль закрепляет Blueprint-навыки через создание конкретных игровых действий. Обучающийся собирает механику как связанный сценарий, в котором игрок понимает цель, получает обратную связь, сталкивается с ограничениями и может завершить уровень понятным способом.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление

кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- создать мини-прототип со сбором предметов, счетом и условием победы;
- реализовать интерактивный объект, который меняет состояние уровня;
- подготовить чек-лист проверки игровой механики.

Типовые ошибки, подлежащие разбору:

- механика существует отдельно от цели игрока;
- счет или состояние уровня обновляются некорректно;
- нет понятного завершения сценария;
- feedback не объясняет игроку результат действия.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.5. Модуль 5. Инструменты моделирования в редакторе, источники света, базовый шейдер

Цель модуля: сформировать навыки создания простого уровня средствами редактора Unreal Engine, работы с инструментами моделирования, источниками света, базовым шейдером, material instances и визуальной читаемостью сцены.

Объем модуля: 68 ч., в том числе лекционные занятия — 20 ч., практические занятия — 31 ч., самостоятельная работа — 11 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- создавать blockout и простые формы средствами Modeling Mode;
- настраивать Static Mesh, material slots, collisions и организацию assets;

- создавать базовый материал, использовать параметры шейдера и material instances;
- настраивать источники света, базовую атмосферу сцены и проверять визуальную читаемость уровня.

Таблица 9 – Тематический план модуля 5. Инструменты моделирования в редакторе, источники света, базовый шейдер

№	Тема	Часы	Содержание
1	Modeling Mode и blockout	9	инструменты моделирования в редакторе, primitive shapes, планировка пространства и масштаб.
2	Создание и правка простых meshes	9	extrude, bevel, merge, duplicate, pivot, базовая подготовка игровых объектов.
3	Static Mesh settings, UV и material slots	8	настройки mesh, простые UV, material slots, collision для объектов уровня.
4	Источники света	8	Directional Light, Point Light, Spot Light, Sky Light, интенсивность, цвет и читаемость сцены.
5	Базовый шейдер и материал	11	Base Color, Roughness, Metallic, Normal, emissive, параметры и базовые узлы материала.
6	Material Instance и вариативность	8	instances, параметризация цвета и свойств, повторное использование материалов.
7	Post Process и визуальный стиль	6	exposure, bloom, color grading, fog и ограничения визуальных эффектов.
8	Техническая чистка сцены	3	проверка именования, лишних объектов, collision complexity и размера assets.
9	Промежуточная аттестация	6	уровень с blockout, простыми моделями, светом, базовым шейдером и описанием визуальных решений.

Содержание модуля: Модуль связывает геймдизайн уровня с визуальной реализацией. Обучающийся учится быстро собирать игровое пространство, создавать базовые объекты прямо в редакторе, настраивать свет и материалы

так, чтобы механика была читаемой, проект оставался управляемым и не превращался в набор случайных assets.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- собрать уровень для будущей игры Collect the coin;
- создать набор простых объектов средствами Modeling Mode;
- подготовить базовый материал и несколько material instances для сцены.

Типовые ошибки, подлежащие разбору:

- уровень не имеет понятного маршрута игрока;
- модели и материалы названы хаотично;
- свет пересвечивает сцену или скрывает цель игрока;
- один материал дублируется вместо использования instances.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.6. Модуль 6. Создание кастомного персонажа, настройка управления и камеры

Цель модуля: сформировать навыки создания кастомного персонажа, настройки Pawn или Character, управления, камеры, базовой анимации и связи персонажа с игровыми механиками.

Объем модуля: 80 ч., в том числе лекционные занятия — 23 ч., практические занятия — 40 ч., самостоятельная работа — 11 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- создавать управляемого персонажа или Pawn под задачу проекта;
- настраивать Enhanced Input, движение, поворот, ограничения и чувствительность управления;
- создавать камеру через Camera Component и Spring Arm, выбирать подходящий ракурс;
- подключать mesh, базовые анимационные состояния и взаимодействие персонажа с объектами уровня.

Таблица 10 – Тематический план модуля 6. Создание кастомного персонажа, настройка управления и камеры

№	Тема	Часы	Содержание
1	Кастомный персонаж и требования к управлению	6	роль персонажа в проекте, вид камеры, скорость, ограничения и пользовательский сценарий.
2	Character, Pawn и основные components	10	capsule, mesh, movement component, camera component, possession и PlayerController.
3	Enhanced Input	13	Input Actions, Mapping Context, движение, поворот, прыжок, действие взаимодействия и проверка ввода.
4	Движение и ограничения	9	скорость, ускорение, поворот, гравитация, границы уровня и поведение на препятствиях.
5	Камера и Spring Arm	9	ракурс, дистанция, lag, collision test, first-person и third-person на обзорном уровне.

№	Тема	Часы	Содержание
6	Mesh персонажа и визуальная настройка	9	Skeletal Mesh или Static Mesh, материалы, sockets, простая кастомизация внешнего вида.
7	Базовая анимация персонажа	9	Animation Blueprint на обзорном уровне, idle, walk, jump, переходы и связь со скоростью.
8	Персонаж и игровые механики	9	подбор предметов, взаимодействие, урон, зоны, feedback и проверка состояния игрока.
9	Промежуточная аттестация	6	кастомный персонаж с управлением, камерой, базовой анимацией и взаимодействием с объектами уровня.

Содержание модуля: Модуль переводит учебные механики в управляемый игровой сценарий. Обучающийся создает собственного персонажа, задает схему управления, выбирает камеру под жанр и связывает поведение персонажа с объектами, коллизиями и состояниями уровня.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- создать кастомного персонажа для учебного игрового прототипа;
- настроить управление, камеру и действие взаимодействия;

- подготовить демонстрационный уровень для проверки персонажа.

Типовые ошибки, подлежащие разбору:

- управление работает только в отдельном шаблоне и не перенесено в учебный проект;
- камера мешает видеть цель игрока;
- персонаж не взаимодействует с объектами уровня;
- отсутствует описание input settings и управления.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.7. Модуль 7. Взаимодействие объектов через систему коллизий

Цель модуля: сформировать навыки настройки системы коллизий в Unreal Engine, обработки overlap и hit events, использования collision channels, trace-проверок, физики и отладки взаимодействий между объектами.

Объем модуля: 56 ч., в том числе лекционные занятия — 17 ч., практические занятия — 25 ч., самостоятельная работа — 8 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- настраивать collision presets, object channels, trace channels и responses;
- обрабатывать overlap events, hit events, триггеры, зоны и предметы;
- использовать line trace для выбора, проверки и взаимодействия с объектами;
- отлаживать collision, устранять ложные срабатывания и документировать правила взаимодействия.

Таблица 11 – Тематический план модуля 7. Взаимодействие объектов через систему коллизий

№	Тема	Часы	Содержание
1	Система коллизий Unreal Engine	8	blocking, overlap, ignore, collision enabled, complex/simple collision и типовые сценарии.

№	Тема	Часы	Содержание
2	Collision presets и channels	8	object type, trace channel, response matrix, пользовательские каналы и правила именования.
3	Overlap Events	7	триггерные зоны, collectables, checkpoints, опасные зоны и проверка входа объекта.
4	Hit Events	7	столкновения, физические реакции, снаряды, impact feedback и ограничения применения.
5	Line Trace и взаимодействие	7	лучевые проверки, interact prompt, pickup, выбор объекта, дистанция действия.
6	Физика и constraints на базовом уровне	7	simulate physics, mass, impulse, physics material, ограничения стабильности.
7	Отладка коллизий	6	визуализация collision, debug draw, проверка каналов, список типовых ошибок.
8	Промежуточная аттестация	6	сцена с несколькими типами взаимодействий, collision channels, trace и проверяемыми событиями.

Содержание модуля: Модуль посвящен устойчивой логике взаимодействий. Обучающийся учится проектировать правила столкновений до реализации механики, проверять, какие объекты должны блокировать, пересекаться или игнорировать друг друга, и использовать коллизии как основу подбора предметов, снарядов, триггеров и опасных зон.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление

кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- создать тренажер коллизий с overlap, hit events и line trace;
- настроить правила взаимодействия для монет, препятствий, персонажа и снарядов;
- подготовить таблицу collision channels для учебного проекта.

Типовые ошибки, подлежащие разбору:

- collision настроена случайно и мешает gameplay;
- события overlap срабатывают несколько раз без контроля состояния;
- line trace не учитывает дистанцию и тип объекта;
- отсутствует debug-сценарий проверки коллизий.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.8. Модуль 8. Основы создания ИИ с использованием State Tree

Цель модуля: сформировать базовые навыки создания поведения NPC с использованием State Tree, навигации, AI Controller, условий перехода между состояниями, задач, сенсоров и отладки поведения.

Объем модуля: 56 ч., в том числе лекционные занятия — 17 ч., практические занятия — 25 ч., самостоятельная работа — 8 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- создавать простого NPC и связывать его с AI Controller;
- настраивать NavMesh и проверять движение NPC по уровню;
- проектировать State Tree с состояниями, переходами, evaluators и tasks;
- реализовывать патрулирование, преследование, поиск цели и возврат к базовому состоянию.

Таблица 12 – Тематический план модуля 8. Основы создания ИИ с использованием State Tree

№	Тема	Часы	Содержание
1	Роль ИИ в игровом прототипе	6	NPC, цели поведения, состояние мира, ограничения простого ИИ и критерии проверки.
2	NPC, Pawn и AI Controller	7	создание управляемого NPC, possession, движение, базовая структура объекта.
3	Навигация и NavMesh	7	NavMeshBoundsVolume, reachable paths, debug navigation, ограничения уровня.
4	Структура State Tree	8	states, tasks, transitions, evaluators, schema и логика выбора состояния.
5	Tasks, Evaluators и переходы	8	проверка условий, обновление данных, выполнение действий, cooldown и возврат в состояние.
6	Обнаружение игрока и события мира	6	зоны видимости, distance checks, сигналы уровня, переход к преследованию.
7	Патруль, преследование и поиск	8	маршрут NPC, случайные точки, chase, потеря цели, return to patrol.
8	Промежуточная аттестация	6	NPC на State Tree с патрулем, обнаружением игрока и проверяемой реакцией в уровне.

Содержание модуля: Модуль вводит обучающегося в разработку управляемого поведения NPC через State Tree. Акцент сделан на ясной структуре состояний, переходов и задач, чтобы поведение можно было проверить, объяснить и встроить в игровые проекты Collect the coin или шутер с дроном.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;

- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- создать NPC с State Tree, патрулем и реакцией на игрока;
- настроить NavMesh и debug-сценарий проверки маршрута;
- добавить состояние преследования или поиска цели для игрового прототипа.

Типовые ошибки, подлежащие разбору:

- State Tree не имеет понятной структуры состояний;
- NavMesh не покрывает рабочую область уровня;
- NPC застревает, не теряет цель или не возвращается к патрулю;
- условия переходов не документированы и трудно проверяются.

Минимальный уровень освоения: выполнены обязательные практические задания модуля, результат запускается и может быть проверен по заданным критериям.

Повышенный уровень освоения: результат дополнен аккуратной структурой, пояснением решений, улучшенной визуальной подачей и материалами для портфолио.

2.3.9. Модуль 9. Основы работы с UMG и User Widget

Цель модуля: сформировать навыки создания пользовательского интерфейса игры средствами UMG и User Widget, настройки экранных элементов, меню, HUD и связи интерфейса с игровой логикой на Blueprint.

Объем модуля: 48 ч., в том числе лекционные занятия — 12 ч., практические занятия — 21 ч., самостоятельная работа — 9 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- создавать Widget Blueprint и размещать элементы пользовательского интерфейса;
- настраивать компоновку, anchors, масштабирование и визуальную иерархию интерфейса;
- выводить счет, здоровье, таймер, подсказки, состояние победы и поражения;
- связывать UMG с Blueprint-логикой проекта и проверять интерфейс в разных режимах запуска.

Таблица 13 – Тематический план модуля 9. Основы работы с UMG и User Widget

№	Тема	Часы	Содержание
1	UMG и роль интерфейса в игре	6	HUD, меню, обратная связь, состояние игры, отличие интерфейса прототипа от интерфейса готового продукта.
2	User Widget и Widget Blueprint	8	Создание виджета, панель Designer, Graph, Canvas Panel, hierarchy, порядок отображения элементов.
3	Layout, anchors и scaling	8	Адаптация интерфейса под разные разрешения, привязки, отступы, выравнивание, читаемость элементов.
4	Text, Button, Progress Bar и Image	7	Базовые элементы UMG, настройка внешнего вида, обработка нажатий, визуальные индикаторы.
5	Обновление данных интерфейса	6	Передача счета, здоровья, таймера и статуса уровня из Blueprint-логики в User Widget.
6	HUD, меню паузы и финальные экраны	7	Добавление виджета во viewport, переключение ввода, пауза, экран победы, экран поражения, перезапуск.
7	Промежуточная аттестация	6	Создание набора интерфейсных экранов и демонстрация связи UI с игровыми событиями.

Содержание модуля: Модуль раскрывает интерфейс как самостоятельную часть игрового продукта. Обучающиеся проектируют

HUD, меню паузы и финальные экраны, подключают пользовательские виджеты к игровым событиям и проверяют, что игрок получает понятную обратную связь о целях, прогрессе, ошибках и завершении уровня.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- собрать HUD для проекта Collect the coin со счетчиком монет и подсказками;
- создать меню паузы, экран победы и экран поражения;
- подключить User Widget к событиям сбора предметов, урона, таймера или завершения уровня.

Типовые ошибки, подлежащие разбору:

- не настроены anchors, поэтому интерфейс ломается на другом разрешении;
- обновление виджета привязано к хрупкой логике и работает только в одном сценарии;
- экраны победы и поражения визуально созданы, но не связаны с состоянием игры;
- кнопки меню не проверены на перезапуске, выходе и повторном запуске уровня.

Минимальный уровень освоения: создан HUD с корректным выводом хотя бы двух игровых показателей и одним экраном завершения уровня.

Повышенный уровень освоения: реализован полный набор интерфейсных экранов с адаптивной компоновкой, понятной навигацией и проверенной связью с игровой логикой.

2.3.10. Модуль 10. Профилирование и оптимизация в Unreal Engine

Цель модуля: сформировать базовые навыки профилирования, поиска узких мест и оптимизации игрового прототипа в Unreal Engine с опорой на измеримые показатели производительности.

Объем модуля: 48 ч., в том числе лекционные занятия — 12 ч., практические занятия — 21 ч., самостоятельная работа — 9 ч., контроль — 6 ч.

Планируемые результаты по модулю:

- использовать базовые команды и инструменты анализа FPS, CPU и GPU-нагрузки;
- находить типовые проблемы Blueprint-логики, Tick, таймеров и лишних обновлений;
- оптимизировать уровни, ресурсы, материалы, освещение, коллизии и настройки качества на базовом уровне;
- подготавливать отчет по профилированию и подтверждать улучшения в проекте.

Таблица 14 – Тематический план модуля 10. Профилирование и оптимизация в Unreal Engine

№	Тема	Часы	Содержание
1	Цели оптимизации и метрики	7	FPS, frame time, стабильность, отличие субъективного ощущения от измеряемого результата.
2	Stat-команды и первичная диагностика	4	stat fps, stat unit, stat game, stat gpu, интерпретация показателей на базовом уровне.
3	Unreal Insights на обзорном уровне	8	Назначение инструмента, запись сессии, просмотр событий, общий подход к поиску узких мест.
4	Оптимизация Blueprint-логики	8	Сокращение лишнего Tick, использование событий и таймеров, уменьшение количества повторных вычислений.

№	Тема	Часы	Содержание
5	Ресурсы, материалы и свет	8	Размеры ассетов, количество объектов, простые материалы, источники света, влияние теней и постобработки.
6	Scalability и настройки проекта	6	Настройки качества, проверка прототипа на целевой машине, базовая подготовка к сборке.
7	Отчет по оптимизации	5	Фиксация показателей до и после, описание изменений, вывод о влиянии правок.
8	Промежуточная аттестация	2	Демонстрация найденной проблемы и подтвержденного улучшения производительности.

Содержание модуля: Модуль формирует представление о производительности как проверяемом свойстве проекта. Обучающиеся измеряют состояние прототипа до оптимизации, выявляют одну или несколько причин просадок, вносят правки и оформляют короткий отчет с подтверждением результата.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- провести профилирование учебного уровня до и после доработки;

- исправить одну проблему Blueprint-логики или нагрузки от сцены;
- подготовить отчет с показателями, скриншотами и кратким выводом.

Типовые ошибки, подлежащие разбору:

- оценка производительности выполняется только субъективно, без измерений;
- все действия помещаются в Event Tick, хотя подходят события или таймеры;
- в сцену добавляются тяжелые ассеты без проверки влияния на FPS;
- после оптимизации не проводится повторная проверка на тех же условиях.

Минимальный уровень освоения: проведено базовое профилирование проекта, выявлена и исправлена хотя бы одна причина избыточной нагрузки.

Повышенный уровень освоения: оформлен отчет с измерениями до и после, обоснованием выбранных правок и подтвержденным улучшением стабильности прототипа.

2.3.11. Модуль 11. Проект «Collect the coin»

Цель модуля: организовать разработку завершеного учебного проекта Collect the coin, в котором объединяются уровень, персонаж, предметы, коллизии, счет, пользовательский интерфейс, условия победы и базовое тестирование.

Объем модуля: 56 ч., в том числе лекционные занятия — 6 ч., практические занятия — 39 ч., самостоятельная работа — 11 ч., контроль — 0 ч.

Планируемые результаты по модулю:

- создавать игровый прототип Collect the coin в Unreal Engine;
- реализовывать монеты, сбор предметов, счетчик, условие победы и перезапуск;
- оформлять уровень, простой визуальный стиль, User Widget и проверяемые коллизии;
- подготавливать демонстрацию проекта, скриншоты, описание управления и чек-лист тестирования.

Таблица 15 – Тематический план модуля 11. Проект «Collect the coin»

№	Тема	Часы	Содержание
1	Концепция и правила Collect the coin	6	Цель игрока, состав уровня, набор монет, условия победы и поражения, требования к прототипу.
2	Blockout уровня	8	Планировка площадки, точки старта, зоны движения, расположение монет, препятствия и читаемость маршрута.
3	Coin Actor и логика сбора	8	Blueprint монеты, коллизия, исчезновение, звук или визуальная обратная связь, защита от повторного сбора.
4	Счет, прогресс и состояние игры	8	Подсчет собранных предметов, общее количество монет, Game Mode или управляющий Blueprint, состояние победы.
5	Препятствия, таймер и перезапуск	8	Дополнительная сложность, таймер уровня, зоны опасности, рестарт и возврат игрока к началу.
6	UMG для проекта	6	HUD со счетчиком монет, подсказки, экран победы и кнопка повторной игры.
7	Тестирование и полировка	6	Проверка всех монет, маршрута, коллизий, интерфейса, перезапуска, финального состояния.
8	Представление проекта	6	Демонстрация проекта, краткое описание решений, фиксация результата для портфолио.

Содержание модуля: Проектный модуль используется для первого полного цикла разработки игры от замысла до демонстрации. Обучающиеся собирают небольшой, но заверченный продукт, в котором видна цель игрока, есть интерактивные объекты, интерфейс, результат прохождения и проверяемая структура Blueprint-логики.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;

- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- разработать игру Collect the coin с уровнем, монетами, счетом, интерфейсом и условием победы;
- подготовить краткое описание игры, управление, скриншоты и чек-лист тестирования;
- провести демонстрацию проекта и получить рекомендации по доработке.

Типовые ошибки, подлежащие разбору:

- монеты считаются некорректно или могут быть собраны повторно;
- игрок не понимает, сколько предметов осталось собрать;
- условие победы не связано с реальным прогрессом уровня;
- проект невозможно проверить без устного объяснения автора.

Минимальный уровень освоения: игра запускается, позволяет собрать монеты, отображает счет и завершает уровень после выполнения цели.

Повышенный уровень освоения: проект имеет читаемый уровень, устойчивую Blueprint-структуру, понятный интерфейс, полировку и материалы для портфолио.

2.3.12. Модуль 12. Проект «Шутер с летающим дроном»

Цель модуля: организовать разработку более сложного проектного прототипа в жанре шутера с летающим дроном, объединяющего управление, камеру, стрельбу, цели, урон, интерфейс, поведение объектов и подготовку к публикации.

Объем модуля: 80 ч., в том числе лекционные занятия — 6 ч., практические занятия — 57 ч., самостоятельная работа — 17 ч., контроль — 0 ч.

Планируемые результаты по модулю:

- создавать управляемый летающий Drone Pawn с камерой и ограничениями движения;
- реализовывать стрельбу, попадания, цели, здоровье, урон, счет и игровые состояния;
- добавлять AI-цели или противников с использованием State Tree на базовом уровне;
- подготавливать сборку, отчет по тестированию, материалы демонстрации и портфолио.

Таблица 16 – Тематический план модуля 12. Проект «Шутер с летающим дроном»

№	Тема	Часы	Содержание
1	Техническое задание и core loop	6	Игровая цель, управление дроном, противники или цели, условия победы и поражения, критерии готовности.
2	Drone Pawn и управление	11	Создание Pawn, настройка движения по осям, скорость, ускорение, ограничения, ощущение управления.
3	Камера и ограничения движения	11	Камера от третьего лица или свободная камера, поворот, дистанция, зона полета, границы уровня.
4	Стрельба и проверка попаданий	11	Projectile или line trace, спавн выстрела, направление, попадание, эффекты, защита от некорректных столкновений.
5	Здоровье, цели и игровые состояния	9	Health, damage, уничтожение целей, подсчет очков, win/lose state, перезапуск.
6	AI-цели и State Tree	11	Простое поведение противника или цели, переходы состояний, патруль, преследование, реакция на игрока.

№	Тема	Часы	Содержание
7	UMG и игровой цикл	9	HUD с очками, здоровьем и боезапасом, экран победы, экран поражения, меню паузы.
8	Оптимизация, сборка и документация	6	Профилирование проекта, исправление узких мест, подготовка build, README и инструкции запуска.
9	Представление проекта	6	Демонстрация игрового цикла, архитектуры, основных Blueprint и результата тестирования.

Содержание модуля: Модуль закрепляет навыки разработки через проект с более высокой технической сложностью. Обучающиеся проектируют игровой цикл, создают управляемый дрон, реализуют стрельбу, реакции целей, интерфейс и базовое поведение противников, после чего приводят проект к демонстрационному состоянию.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- разработать прототип шутера с летающим дроном, целями, стрельбой, счетом и интерфейсом;
- подготовить сценарий демонстрации проекта в редакторе и в сборке;
- оформить краткий отчет по тестированию и оптимизации.

Типовые ошибки, подлежащие разбору:

- управление дроном нестабильно и не ограничено границами уровня;
- попадания работают случайно из-за неверной настройки трассировки или коллизий;
- AI-цели существуют отдельно от игрового цикла и не влияют на результат;
- проект не имеет сборки, описания управления и материалов демонстрации.

Минимальный уровень освоения: создан играбельный прототип с управляемым дроном, стрельбой, целями, счетом и завершением уровня.

Повышенный уровень освоения: проект содержит устойчивое управление, понятную архитектуру Blueprint, AI-элементы, интерфейс, сборку и материалы для портфолио.

2.3.13. Модуль 13. Проектная практика и портфолио. Загрузка игр на itch и другие площадки

Цель модуля: подготовить итоговый проект к демонстрации и публикации, оформить портфолио-раздел, собрать материалы для страницы проекта и разместить игру на itch.io или другой подходящей площадке.

Объем модуля: 44 ч., в том числе лекционные занятия — 0 ч., практические занятия — 33 ч., самостоятельная работа — 11 ч., контроль — 0 ч.

Планируемые результаты по модулю:

- выбирать и согласовывать тему итогового проекта с измеримыми критериями готовности;
- подготавливать build, README, видео, скриншоты, описание управления и перечень использованных ресурсов;
- оформлять страницу проекта на itch.io или другой площадке с понятной инструкцией запуска;
- проходить предзащиту, фиксировать замечания и вносить финальные улучшения.

Таблица 17 – Тематический план модуля 13. Проектная практика и портфолио. Загрузка игр на itch и другие площадки

№	Тема	Часы	Содержание
1	Выбор темы итогового проекта	7	Формулировка идеи, ограничение объема, критерии готовности, риски, план доработки.
2	План доработки и чек-лист качества	7	Backlog, приоритеты, обязательный минимум, расширенный уровень, проверка запуска и прохождения.
3	Packaging и проверка build	8	Подготовка сборки, структура папок, инструкция запуска, проверка на другом устройстве или чистом окружении.
4	Страница проекта на itch.io	8	Описание игры, жанр, управление, скриншоты, видео, теги, файлы сборки, публичная или закрытая публикация.
5	Другие площадки и варианты публикации	6	GitHub Releases, Google Drive, облачные хранилища, портфолио-сайт, ограничения и плюсы вариантов.
6	Портфолио-материалы	8	README, обложка проекта, gameplay-видео, скриншоты, описание роли автора, список технологий и ассетов.
7	Предзащита и финальная доработка	0	Демонстрация проекта, получение замечаний, план исправлений, финальная проверка комплекта материалов.

Содержание модуля: Модуль переводит учебные проекты в портфолио-формат. Основной акцент делается на воспроизводимости результата: проект должен запускаться, быть понятным стороннему пользователю, иметь описание управления, визуальные материалы, страницу публикации и подтверждение авторского вклада.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;

- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: зачет по практической работе, чек-листу и демонстрации результата.

Проектные задания и артефакты:

- подготовить итоговый проект или один из крупных учебных проектов к публикации;
- создать страницу на itch.io или другой площадке и загрузить сборку или материалы демонстрации;
- оформить портфолио-карточку проекта с описанием задач, технологий и результата.

Типовые ошибки, подлежащие разбору:

- страница проекта есть, но отсутствует инструкция запуска;
- сборка не проверена вне рабочего проекта в редакторе Unreal Engine;
- в описании не указаны роль автора, использованные технологии и сторонние ассеты;
- замечания предзащиты не зафиксированы и не доведены до исправлений.

Минимальный уровень освоения: проект имеет проверяемую сборку или демонстрационный комплект, README, скриншоты и опубликованную или подготовленную страницу.

Повышенный уровень освоения: оформлен полноценный портфолио-кейс с видео, понятным описанием, инструкцией запуска, списком технологий и ссылкой на площадку.

2.3.14. Модуль 14. Промежуточная аттестация по завершении Программы

Цель модуля: оценить готовность обучающегося к самостоятельной разработке простых игровых прототипов в Unreal Engine 5 через защиту итогового проекта, демонстрацию сборки, объяснение архитектуры и ответы на вопросы комиссии.

Объем модуля: 36 ч., в том числе лекционные занятия — 0 ч., практические занятия — 0 ч., самостоятельная работа — 0 ч., контроль — 36 ч.

Планируемые результаты по модулю:

- представлять итоговый проект и подтверждать самостоятельность выполненной работы;
- демонстрировать работоспособность проекта в сборке или в согласованном демонстрационном сценарии;
- объяснять структуру Blueprint, уровень, персонажа, коллизии, UMG, State Tree, оптимизацию и использованные ассеты;
- отвечать на вопросы комиссии и аргументировать выбранные технические решения.

Таблица 18 – Тематический план модуля 14. Промежуточная аттестация по завершении Программы

№	Тема	Часы	Содержание
1	Проверка комплекта материалов	0	Проект, сборка, README, скриншоты, видео, ссылка на публикацию, перечень ассетов, инструкция запуска.
2	Демонстрация проекта	0	Показ игрового цикла, управления, цели, интерфейса, победы или поражения, стабильности запуска.
3	Защита архитектуры	22	Объяснение основных Blueprint, структуры уровня, взаимодействий, коллизий, UI, AI и оптимизационных решений.
4	Ответы на вопросы комиссии	0	Проверка понимания выполненной работы, ограничений проекта и возможных направлений развития.

№	Тема	Часы	Содержание
5	Оформление результатов аттестации	14	Фиксация оценки, замечаний, итогового решения и рекомендаций по дальнейшему развитию портфолио.

Содержание модуля: Промежуточная аттестация по завершении Программы проводится в форме защиты проекта. Проверяются визуальный результат, воспроизводимость, логика реализации, способность обучающегося объяснить Blueprint-архитектуру, интерфейс, взаимодействия объектов, элементы AI, подход к оптимизации и качество подготовленных материалов.

Практические виды работ:

- разбор демонстрационного примера в редакторе Unreal Engine 5;
- выполнение практического задания по теме модуля;
- проверка результата по чек-листу и исправление типовых ошибок;
- сохранение итогового варианта в структуре учебного проекта или портфолио.

Самостоятельная работа обучающегося: изучение конспекта, повторение действий преподавателя, доработка Blueprint-логики, оформление кратких пояснений к решению, подготовка скриншотов или короткой демонстрации результата.

Форма контроля: итоговая защита проекта перед аттестационной комиссией.

Проектные задания и артефакты:

- защитить итоговый проект по Unreal Engine 5;
- показать работоспособный build или согласованный демонстрационный сценарий;
- представить README, скриншоты, видео или ссылку на страницу проекта.

Типовые ошибки, подлежащие разбору:

- проект запускается только на компьютере автора и не имеет инструкции;
- обучающийся не может объяснить ключевые Blueprint и связи между объектами;

- отсутствуют доказательства самостоятельной работы и этапов разработки;
- комплект материалов неполный: нет сборки, README, демонстрации или ссылок.

Минимальный уровень освоения: обучающийся представляет работоспособный проект, демонстрирует основной игровой цикл и объясняет ключевые технические решения.

Повышенный уровень освоения: обучающийся защищает завершенный портфолио-проект с понятной архитектурой, проверенной сборкой, опубликованной страницей и качественными материалами.

2.4. Проектная практика и консультации

Проектная практика направлена на доработку итогового проекта, оформление портфолио и подготовку к защите. В ходе проектной практики обучающийся выбирает итоговый проект, согласует с преподавателем объем доработок, уточняет игровой цикл, проверяет запуск, исправляет ошибки, готовит README, скриншоты, видео, сборку или воспроизводимый сценарий запуска.

Консультации проводятся в дистанционном формате с использованием электронной информационно-образовательной среды, средств видеосвязи, репозитория или облачного хранилища проектных материалов. По итогам консультаций обучающийся получает рекомендации по структуре Blueprint-графов, корректности механик, читаемости уровня, UI, коллизиям, State Tree, оптимизации и подготовке защиты.

Результатом проектной практики является комплект материалов для допуска к промежуточной аттестации по завершении Программы: итоговый проект Unreal Engine 5, README, скриншоты, короткое видео или сценарий демонстрации, ссылка на страницу проекта при наличии, чек-лист проверки и краткое описание авторского вклада.

3. Организационно-педагогические условия реализации программы

3.1. Общие условия реализации

Организация обеспечивает реализацию Программы в соответствии с учебным планом, календарным учебным графиком, расписанием, локальными нормативными актами и договором об образовании. Образовательный процесс строится на сочетании объяснения материала, демонстрации работы в Unreal Engine 5, практических заданий, самостоятельной доработки, обратной связи преподавателя, промежуточной аттестации и итоговой защиты проекта.

Обучающимся предоставляется доступ к электронной информационно-образовательной среде, учебным материалам, заданиям, инструкциям, чек-листам, записям занятий при наличии, средствам коммуникации и каналам сдачи работ. Преподаватель обеспечивает сопровождение проектной траектории, проверку промежуточных результатов и рекомендации по доработке проекта.

3.2. Материально-технические условия

Для реализации Программы необходимы:

- персональный компьютер или ноутбук обучающегося, соответствующий минимальным требованиям Unreal Engine 5;
- установленный Unreal Engine 5 и Epic Games Launcher;
- среда для хранения и передачи проектных материалов, включая репозиторий, облачное хранилище или электронную информационно-образовательную среду;
- стабильный доступ к сети Интернет;
- средства видеосвязи и демонстрации экрана для дистанционных занятий и консультаций;
- средства записи коротких видео демонстраций проекта;
- браузер и учетная запись для публикации проекта на itch.io или другой согласованной площадке при необходимости;
- офисное или текстовое программное обеспечение для подготовки README, описаний, чек-листов и презентационных материалов.

При выполнении заданий допускается использование учебных assets, стандартных шаблонов Unreal Engine 5, открытых материалов, предоставленных преподавателем, и ресурсов, разрешенных для образовательного применения с соблюдением условий лицензирования.

3.3. Электронная информационно-образовательная среда

Электронная информационно-образовательная среда обеспечивает размещение учебных материалов, заданий, чек-листов, ссылок, расписания, уведомлений, результатов проверок и обратной связи. Через нее обучающийся получает доступ к содержанию Программы, отправляет проектные материалы, получает комментарии преподавателя и отслеживает статус выполнения контрольных точек.

В рамках дистанционного взаимодействия могут использоваться видеоконференции, мессенджеры, электронная почта, система управления обучением, облачные папки, репозитории и иные средства, утвержденные организацией. Конкретный набор средств определяется локальными нормативными актами и организационными условиями реализации Программы.

3.4. Кадровые условия

Реализация Программы обеспечивается педагогическими работниками и (или) специалистами-практиками, имеющими профессиональный опыт в сфере разработки программного обеспечения, игровых прототипов, интерактивных приложений, 3D-сред, визуального программирования, образовательных технологий или проектной деятельности.

Преподаватель должен владеть методикой дистанционного обучения, уметь демонстрировать работу в Unreal Engine 5, проверять Blueprint-логику, проектные материалы, игровые механики, UI, коллизии, State Tree, оптимизацию и давать обучающимся прикладную обратную связь по доработке проекта.

3.5. Условия для несовершеннолетних обучающихся

При обучении несовершеннолетних организация учитывает возрастные особенности, обеспечивает информационную безопасность и педагогически обоснованную продолжительность непрерывной работы с экранными устройствами. Расписание предусматривает перерывы и смену видов деятельности. Коммуникация, публикация проектов, использование внешних сервисов, обработка изображений, голосовых данных и иных персональных данных осуществляются с учетом согласий законных представителей и локальных правил.

Преподаватель не допускает включения в открытые репозитории ключей доступа, паролей, персональных данных и материалов, нарушающих права третьих лиц. При необходимости используются учебные учетные записи, закрытые репозитории и обезличенные наборы данных.

3.6. Условия для обучающихся с ограниченными возможностями здоровья

При наличии запроса и организационной возможности предоставляются увеличенный срок выполнения заданий, доступные форматы материалов, субтитры или текстовые конспекты, альтернативные способы представления проекта и иные специальные условия в соответствии с индивидуальными потребностями и локальными актами организации.

3.7. Учебно-методическое обеспечение

Учебно-методическое обеспечение Программы включает:

- рабочие программы модулей;
- презентации, конспекты, инструкции и демонстрационные материалы;
- практические задания и проектные шаблоны;
- чек-листы проверки механик, Blueprint-графов, уровня, UI, коллизий, State Tree, оптимизации и сборки;
- критерии промежуточной и промежуточной аттестации по завершении Программы;
- рекомендации по оформлению README, портфолио, страницы проекта и демонстрационного видео;

- примеры типовых ошибок и рекомендации по их исправлению.

3.8. Применение электронного обучения и дистанционных образовательных технологий

Программа реализуется с применением электронного обучения и дистанционных образовательных технологий. Электронное обучение используется для предоставления учебных материалов, заданий, записей, методических рекомендаций, средств коммуникации, фиксации результатов контроля и обратной связи. Дистанционные образовательные технологии используются для организации взаимодействия обучающегося и педагогических работников на расстоянии, проведения консультаций, проверки работ, промежуточной и промежуточной аттестации по завершении Программы.

При реализации Программы организация обеспечивает идентификацию обучающегося в электронной информационно-образовательной среде, доступ к материалам в течение периода обучения, возможность сдачи работ, получение обратной связи и фиксацию результатов освоения Программы. Технические требования к устройству, программному обеспечению, каналам связи и учетным данным доводятся до обучающегося до начала обучения или при предоставлении доступа к образовательной среде.

3.9. Информационная безопасность, персональные данные и этика

При реализации Программы обеспечивается соблюдение требований законодательства о персональных данных, правил работы с учетными записями, учебными материалами, проектными файлами и результатами аттестации. Доступ к электронной информационно-образовательной среде предоставляется индивидуально и не подлежит передаче третьим лицам.

Обучающийся обязан соблюдать академическую добросовестность, не представлять чужие работы как собственные, корректно указывать источники заимствований, не нарушать права третьих лиц на программный код, данные, изображения, тексты и иные материалы. При выявлении признаков несамостоятельного выполнения работы организация вправе

назначить дополнительную проверку, собеседование, повторную сдачу или иные процедуры, предусмотренные локальными нормативными актами.

3.10. Локальные процедуры обучения

Организация реализации Программы обеспечивается локальными нормативными актами ООО «Иннопрог», регулирующими прием и зачисление обучающихся, режим занятий, применение электронного обучения и дистанционных образовательных технологий, текущий контроль, текущий контроль и промежуточную аттестацию, пересдачи, апелляции, отчисление, восстановление, выдачу документов об обучении, обработку персональных данных.

Локальные процедуры применяются в части, не противоречащей законодательству Российской Федерации, условиям договора об образовании и настоящей Программе. Обучающийся информируется о значимых правилах реализации Программы до начала обучения или при предоставлении доступа к образовательной среде.

3.11. Обновление программы

Программа подлежит периодическому обновлению с учетом изменений в законодательстве Российской Федерации, локальных нормативных актах организации, версии Unreal Engine 5, практики реализации образовательного процесса, обратной связи обучающихся, требований к проектным результатам и развития инструментов game development.

Обновление может затрагивать формулировки тем, состав практических заданий, чек-листы, проектные примеры, рекомендации по публикации и аттестационные материалы при условии сохранения общей трудоемкости, целей, планируемых результатов и структуры Программы.

4. Оценка качества освоения программы

4.1. Общие положения

Оценка качества освоения Программы включает текущий контроль успеваемости, промежуточную аттестацию, проверку проектной практики и промежуточную аттестацию по завершении Программы. Оценочные процедуры направлены на проверку способности обучающегося применять знания в практической разработке игрового прототипа, объяснять свои решения, исправлять ошибки и готовить проект к демонстрации.

Оценивание проводится на основе практических работ, проектных артефактов, чек-листов, промежуточных зачетов, предзащиты и защиты итогового проекта. В ходе проверки учитываются работоспособность проекта, соответствие заданию, структура Blueprint-логики, корректность взаимодействий, читаемость уровня, качество UI, стабильность демонстрации, оптимизация, самостоятельность выполнения и качество оформления материалов.

4.2. Текущий контроль успеваемости

Текущий контроль проводится в процессе освоения модулей и может включать:

- проверку практических заданий в Unreal Engine 5;
- демонстрацию Blueprint-графов и объяснение логики работы;
- проверку структуры проекта, именования assets и сохранения уровней;
- проверку игровых механик, коллизий, UI, State Tree и оптимизации по чек-листу;
- короткие устные или письменные ответы по ключевым понятиям модуля;
- анализ ошибок, доработку проекта и повторную сдачу исправленного результата.

4.3. Промежуточная аттестация

Промежуточная аттестация проводится по модулям, для которых она предусмотрена учебным планом. Основной формой промежуточной аттестации является зачет по практической или проектной работе. Зачет может

включать демонстрацию результата, проверку проекта преподавателем, ответы на вопросы и сдачу краткого описания выполненной работы.

Примерные формы промежуточной аттестации:

- учебная сцена Unreal Engine 5 с корректной структурой проекта и инструкцией запуска;
- Blueprint-прототип с несколькими объектами, событиями, переменными и отладкой;
- механика сбора предметов, счета, таймера, здоровья, победы или поражения;
- оформленный уровень с blockout, светом, базовыми материалами и понятным маршрутом;
- кастомный персонаж или Pawn с управлением, камерой и взаимодействиями;
- карта проверки коллизий, overlap, hit-событий и trace-проверок;
- AI-сценарий на State Tree с несколькими состояниями;
- UMG-интерфейс с HUD, экраном результата и кнопками;
- отчет об оптимизации проекта и проверке сборки.

Зачет считается пройденным, если работа соответствует минимальным критериям модуля, запускается или воспроизводится, имеет проверяемую инструкцию, демонстрирует требуемую механику и содержит исправленные критические ошибки.

4.4. Пересдача, апелляция и подтверждение самостоятельности

При неудовлетворительном результате текущего контроля, промежуточной аттестации или промежуточной аттестации по завершении Программы обучающемуся может быть предоставлена возможность доработки и повторной сдачи в порядке, установленном локальными нормативными актами организации. Сроки пересдачи, количество попыток, основания допуска и порядок фиксации результата определяются локальным актом о текущем контроле, промежуточной и промежуточной аттестации по завершении Программы.

Обучающийся имеет право получить мотивированную обратную связь по результатам проверки и подать апелляцию в порядке, установленном

локальными нормативными актами организации. Самостоятельность выполнения работ подтверждается демонстрацией проекта, объяснением Blueprint-логики, ответами на вопросы, наличием промежуточных артефактов, историей доработок, README, скриншотами, видео и материалами проверки.

4.5. Допуск к промежуточной аттестации по завершении Программы

К промежуточной аттестации по завершении Программы допускается обучающийся, который:

- освоил Программу в объеме, предусмотренном учебным планом;
- выполнил обязательные практические и проектные задания;
- прошел предусмотренные формы промежуточной аттестации;
- подготовил итоговый проект на Unreal Engine 5;
- предоставил README, демонстрационные материалы и инструкцию запуска;
- устранил критические замечания преподавателя по результатам проектной практики и предзащиты.

4.6. Промежуточная аттестация по завершении Программы

Промежуточная аттестация по завершении Программы проводится в форме защиты итогового аттестационного проекта. Итоговый проект должен быть выполнен в Unreal Engine 5 и представлять собой заверченный игровой прототип или интерактивную 3D-сцену с понятной целью, реализованными механиками, пользовательским взаимодействием, интерфейсом, проверяемым сценарием прохождения и материалами портфолио.

4.6.1. Примерная тематика проектных работ

Таблица 19 – Примерная тематика итоговых проектов

№	Тема проекта	Ожидаемый результат
1	Collect the coin	Игра со сбором предметов, счетом, таймером, уровнем, UI, победой, поражением и перезапуском.
2	Шутер с летающим дроном	Игра с управляемым дроном, камерой, стрельбой, целями, попаданиями, UI, простым AI и финальным состоянием.
3	Мини-арена с волнами противников	Прототип арены с персонажем, противниками, State Tree-поведением, счетом, здоровьем и экраном результата.
4	Пазл-комната с интерактивными объектами	Уровень с кнопками, дверями, рычагами, коллизиями, условиями решения и визуальной обратной связью.
5	Платформер с препятствиями	Прототип с персонажем, прыжками, платформами, collectables, таймером, падением, победой и поражением.
6	Образовательная 3D-сцена	Интерактивная сцена с объектами, подсказками, UI, маршрутом пользователя и проверяемым сценарием взаимодействия.

4.6.2. Обязательные требования к итоговому проекту

Итоговый проект должен включать:

- проект Unreal Engine 5, который можно запустить, проверить или воспроизвести по инструкции;

- понятную игровую цель, core loop, условия победы, поражения или завершения сценария;
- игровую механику, реализованную средствами Blueprint;
- уровень или сцену с визуально читаемой структурой, светом и базовыми материалами;
- персонажа, Pawn или управляемый объект с настроенным управлением и камерой;
- корректные коллизии, взаимодействие объектов и проверяемые события;
- пользовательский интерфейс или экран состояния проекта;
- элементы AI через State Tree либо обоснованное описание, почему AI не используется в выбранной теме;
- базовую проверку производительности и исправление критических проблем;
- README, скриншоты, видео или демонстрационный сценарий, а также ссылку на страницу проекта при наличии.

4.6.3. Порядок проведения промежуточной аттестации по завершении Программы

Промежуточная аттестация по завершении Программы проводится комиссией или преподавателем в соответствии с локальными нормативными актами организации. Обучающийся заранее предоставляет проектные материалы и проходит проверку допуска. Во время защиты обучающийся демонстрирует проект, объясняет игровую идею, структуру уровней, Blueprint-логику, механики, UI, коллизии, State Tree, оптимизацию и порядок запуска.

4.6.4. Критерии оценивания итогового проекта

Таблица 20 – Критерии оценивания итогового проекта

№	Критерий	Вес, балл.	Показатели оценивания
1	Игровая идея и проектирование	10	цель, core loop, правила, условия завершения, понятность сценария игрока и соответствие заявленной теме.

№	Критерий	Вес, балл.	Показатели оценивания
2	Работа в Unreal Engine 5 и структура проекта	10	корректная структура папок, levels, assets, именование, сохранение проекта и воспроизводимость запуска.
3	Blueprint-логика и игровые механики	20	события, функции, переменные, механики, состояния, перезапуск, читаемость графов и отсутствие критических ошибок.
4	Персонаж, управление, камера и коллизии	15	удобство управления, корректность камеры, collision presets, overlap, hit-события, trace и взаимодействия.
5	UI, State Tree и пользовательская обратная связь	15	HUD, экран результата, подсказки, AI-поведение, состояния объектов и понятность реакции системы.
6	Визуальная сцена, материалы и оптимизация	10	читаемый уровень, свет, материалы, базовый шейдер, проверка производительности и исправление проблем.
7	Портфолио-материалы и публикация	10	README, скриншоты, видео, сборка, страница проекта, инструкция запуска и полнота материалов.
8	Защита и ответы на вопросы	10	ясность выступления, объяснение решений, подтверждение самостоятельности, ответы на вопросы и принятие замечаний.
Итого		100	–

4.7. Внутренняя и внешняя оценка качества реализации программы

Внутренняя оценка качества реализации Программы осуществляется через анализ результатов текущего контроля, промежуточной аттестации, промежуточной аттестации по завершении Программы, обратной связи обучающихся, качества проектных работ, полноты учебных материалов и соблюдения календарного графика. Внешняя оценка качества может

включать экспертное рассмотрение программы, анализ портфолио обучающийся, завершивший Программу и обратную связь от представителей профессионального сообщества.

5. Методические материалы

5.1. Методические указания для обучающихся

Обучающемуся рекомендуется изучать материалы модулей последовательно, выполнять практические задания в отдельной копии проекта, регулярно сохранять рабочие версии, проверять запуск проекта после каждой крупной доработки и оформлять результаты в README. Перед сдачей работы нужно проверить, что проект открывается, карта сохранена, нужный уровень запускается, UI отображается, механика работает по описанным правилам, а инструкция запуска понятна другому пользователю.

5.2. Методические указания для преподавателей и проверяющих

Преподавателю рекомендуется строить занятия через демонстрацию действия в Unreal Engine 5, разбор логики решения, самостоятельную практику обучающихся, проверку типовых ошибок и постепенную доработку проектного результата. Особое внимание следует уделять структуре проекта, читаемости Blueprint-графов, отсутствию лишнего Tick, корректности ссылок между объектами, коллизиям, сохранению уровней, воспроизводимости запуска и качеству демонстрационных материалов.

5.3. Рекомендуемая литература и ресурсы

- официальная документация Unreal Engine 5 по редактору, Blueprint, UMG, State Tree, Optimization and Performance и Packaging Projects;
- методические материалы ООО «Иннопрог» по разработке игровых прототипов на Unreal Engine 5;
- учебные демонстрационные проекты и шаблоны Unreal Engine 5, используемые в рамках Программы;
- материалы по основам геймдизайна, проектированию игровых механик, core loop, playtesting и подготовке портфолио;
- материалы по визуальному программированию, структурированию Blueprint-графов, отладке и технической документации проекта;
- руководства по подготовке README, записи демонстрационного видео, публикации проекта на itch.io и оформлению страницы проекта.

Приложение 1. Матрица формирования компетенций

Таблица 21 – Матрица формирования компетенций

Код	Компетенция	Модули, в которых формируется компетенция
ПР-1	Проектирование игрового прототипа	Модули 1, 4, 11–14.
ПР-2	Работа в Unreal Engine 5 и организация проекта	Модули 2, 5, 10–14.
ПР-3	Blueprint-разработка и отладка логики	Модули 3–4, 7–8, 11–14.
ПР-4	Реализация игровых механик	Модули 4, 6–7, 11–14.
ПР-5	Визуальная сцена, моделирование, свет и материалы	Модули 2, 5, 11–14.
ПР-6	Персонаж, управление и камера	Модули 6, 11–14.
ПР-7	Коллизии и взаимодействие объектов	Модули 4, 7, 11–14.
ПР-8	State Tree, UMG и пользовательская обратная связь	Модули 8–9, 11–14.
ПР-9	Профилирование, оптимизация и сборка	Модули 10, 12–14.
ПР-10	Портфолио, публикация и защита проекта	Модули 11–14.

Приложение 2. Оценочные материалы промежуточной аттестации

Промежуточная аттестация проводится в форме зачета по практической работе. Оценочные материалы включают проектное задание, чек-лист проверки, требования к демонстрации результата и перечень типовых вопросов по модулю. Работа засчитывается при условии запуска проекта, соответствия заданию, наличия инструкции проверки и устранения критических ошибок.

- по модулю 2 проверяется учебная сцена Unreal Engine 5;
- по модулю 3 проверяется Blueprint-прототип;
- по модулю 4 проверяется базовая игровая механика;
- по модулю 5 проверяется оформленный blockout уровня;
- по модулю 6 проверяется персонаж или Pawn с управлением и камерой;
- по модулю 7 проверяется карта коллизий и взаимодействий;
- по модулю 8 проверяется AI-сценарий на State Tree;
- по модулю 9 проверяется UMG-интерфейс;
- по модулю 10 проверяется отчет об оптимизации.

Приложение 3. Оценочные материалы промежуточной аттестации по завершении Программы

Итоговое задание. Разработать, оформить и защитить игровой прототип или интерактивную 3D-сцену на Unreal Engine 5. Проект должен демонстрировать освоение ключевых результатов Программы: проектирование игровой идеи, работу в редакторе, Blueprint-логику, игровые механики, персонажа или Pawn, коллизии, UMG-интерфейс, элементы State Tree или обоснованную альтернативу, базовую оптимизацию, подготовку сборки или воспроизводимого запуска, README и портфолио-материалы.

Состав материалов, предоставляемых на промежуточную аттестацию по завершении Программы:

- проект Unreal Engine 5 или ссылка на архив, репозиторий, облачную папку либо страницу проекта;
- инструкция запуска проекта и описание необходимой версии Unreal Engine 5;
- README с описанием цели, управления, механик, структуры проекта, ограничений и авторского вклада;
- скриншоты, короткое видео или демонстрационный сценарий прохождения;
- сборка проекта или обоснованная демонстрация запуска в редакторе;
- чек-лист проверки механик, UI, коллизий, State Tree, оптимизации и финального состояния;
- ссылка на itch.io или другую площадку при наличии публикации.

Приложение 4. Перечень локальных нормативных актов, на которые должна опираться реализация программы

Реализация Программы должна опираться на локальные нормативные акты организации, регламентирующие:

- порядок разработки, утверждения, обновления и хранения дополнительных профессиональных программ;
- порядок приема, зачисления, перевода, восстановления и отчисления обучающихся;
- порядок организации образовательного процесса с применением электронного обучения и дистанционных образовательных технологий;
- порядок текущего контроля успеваемости, промежуточной и промежуточной аттестации по завершении Программы;
- порядок пересдачи, апелляции и рассмотрения спорных ситуаций;
- порядок учета посещаемости, выполнения заданий и фиксации результатов обучения;
- порядок выдачи, учета и хранения сертификатов об обучении и справок об обучении;
- правила обработки и защиты персональных данных обучающихся;
- порядок оказания платных образовательных услуг и взаимодействия со обучающимися по договору;
- требования к хранению проектных работ, учебных материалов, оценочных листов и протоколов аттестации.

Приложение 5. Лист актуализации программы

Лист актуализации используется для фиксации изменений Программы «Unreal Engine», связанных с обновлением законодательства, профессиональных стандартов, требований к содержанию и уровню проектов, учебных материалов, оценочных средств, цифровой образовательной среды и проектной практики.

Таблица 22 – Лист актуализации программы

№	Дата	Основание изменения	Содержание изменения
1	02.08.2026	Первичное утверждение	Программа введена в действие приказом организации.
2			
3			

Приложение 6. Требования к репозиторию, README и демонстрации

Для подтверждения освоения Программы и выполнения итогового проекта обучающийся представляет проектные материалы в форме, позволяющей проверить самостоятельность, воспроизводимость и качество результата.

Минимальный состав проектных материалов:

- репозиторий или архив проекта с исходным кодом и структурой каталогов;
- файл README.md или иной документ с описанием цели, функций, структуры, установки и запуска проекта;
- описание используемых технологий, зависимостей, версии языка/фреймворка/инструментов;
- инструкции по подготовке окружения, настройке переменных, запуску приложения, тестов или демонстрационного сценария;
- демонстрационные данные, скриншоты, ссылка на развернутую версию или видеодемонстрация при наличии такой возможности;
- краткое описание ограничений, известных проблем и возможных направлений развития проекта.

Материалы должны быть достаточны для повторной проверки проекта преподавателем, экспертом или членом аттестационной комиссии без дополнительных устных пояснений, за исключением вопросов по защите.

Приложение 7. Дорожная карта подготовки итогового проекта

Дорожная карта используется как методический ориентир для поэтапной подготовки итогового аттестационного проекта по направлению «Unreal Engine-разработчик».

1. Выбор темы и согласование прикладной задачи с преподавателем или куратором.
2. Формулирование цели проекта, целевой аудитории, пользовательских сценариев и критериев приемки.
3. Подготовка структуры проекта, репозитория, базовой документации и плана реализации.
4. Реализация минимально работоспособной версии и фиксация результата в репозитории.
5. Расширение функциональности, обработка ошибок, подготовка данных, интерфейса или демонстрационного сценария в соответствии со спецификой направления.
6. Проверка качества, тестирование, рефакторинг, оформление инструкций по запуску и использованию.
7. Предзащита или консультационная проверка, устранение замечаний, подготовка презентации и демонстрации.
8. Итоговая защита проекта с ответами на вопросы комиссии и фиксацией результата промежуточной аттестации по завершении Программы.

Приложение 8. Контрольный лист соответствия требованиям программы

Контрольный лист применяется для внутренней проверки полноты Программы и готовности ее реализации.

Таблица 23 – Контрольный лист соответствия требованиям программы

№	Проверяемый элемент	Статус	Комментарий
1	Указаны вид, направленность и уровень программы, трудоемкость, форма обучения, язык и срок освоения.	выполнено	
2	Представлены учебный план, календарный учебный график и рабочие программы модулей.	выполнено	
3	Описаны планируемые результаты, предметные результаты и связь с практическими ориентирами.	выполнено	
4	Определены организационно-педагогические условия, ЭИОС, ЭО и ДОТ, кадровые и методические условия.	выполнено	
5	Описаны текущий контроль, промежуточная аттестация, промежуточная аттестация по завершении Программы, пересдача и апелляция.	выполнено	

Продолжение таблицы 23

№	Проверяемый элемент	Статус	Комментарий
6	Определены документы по итогам обучения и перечень локальных актов, обеспечивающих реализацию.	выполнено	