

Муниципальное бюджетное образовательное учреждение дополнительного образования
«Дворец творчества детей и молодежи» городского округа Тольятти



Рассмотрено
На заседании методического совета
Протокол №4 от 11 июня 2026 года

«УТВЕРЖДАЮ»
Директор МБОУДО ДТДМ

Л. В. Скрипинская
Приказ № от 15 июня 2026 года

**Дополнительная общеобразовательная общеразвивающая программа
«Гейм-мастер: Godot Engine»**

- Направленность: техническая
- Возраст учащихся: 12-15 лет
- Срок реализации: 1 год

Разработчик:
Сидельникова М.А.,
педагог дополнительного образования

г. Тольятти 2026 г.

1. Пояснительная записка

Введение

Образовательная программа «Гейм-мастер: Godot Engine» создаёт условия для приобщения учащихся к профессиональной разработке игр, способствует развитию системного и проектного мышления. Программа формирует у обучающихся навыки работы в профессиональной среде, позволяя перейти от создания прототипов игр к разработке полноценных игровых приложений. Образовательная программа «Гейм-мастер: Godot Engine» программа имеет углубленный уровень и требует от учащихся продвинутых навыков программирования на языке высокого уровня, понимания принципов объектно-ориентированного программирования и математической подготовки для работы с физикой и 2D пространством.

Нормативные основания для разработки программы:

- Федеральный закон от 29.12.2012 года № 273-ФЗ «Об образовании в Российской Федерации»;

- Концепция развития дополнительного образования до 2030 года (утверждена распоряжением Правительства РФ от 31.03.2022 № 678-р);

- Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 года № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;

- Приказ Министерства просвещения Российской Федерации от 27.07.2022 года № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Стратегия социально-экономического развития Самарской области на период до 2030 года (утверждена распоряжением Правительства Самарской области от 12.07.2017 № 441);

- Методические рекомендации по разработке дополнительных общеобразовательных общеразвивающих программ (новая редакция, 2025 год).

Направленность дополнительной общеобразовательной программы: техническая

Актуальность программы заключается в том, что она соответствует действующим нормативным актам и государственным программным документам, в ней представлены современные идеи и актуальные направления развития науки, техники и социальной сферы. Программа целиком удовлетворяет потребность общества и детей в решении актуальных для них задач и содействует формированию у учащихся современных знаний, умений и навыков в области технических наук, технологической грамотности и инженерного мышления.

При реализации образовательной программы используются дистанционные технологии обучения, что позволяет удовлетворить современные требования заказчиков образовательных услуг.

Новизна образовательной программы «Гейм-мастер: Godot Engine» заключается в том, что в программу включен раздел «Воспитание», который обеспечивает условия для организации целостного процесса обучения и воспитания. Воспитание в образовательном пространстве Российской Федерации рассматривается как стратегический общенациональный приоритет. Одной из задач развития дополнительного образования детей, в соответствии с «Концепцией развития дополнительного образования детей до 2030 года», является «организация воспитательной деятельности на основе социокультурных, духовно-нравственных ценностей российского общества и государства, а также формирование у детей и молодежи общероссийской гражданской идентичности, патриотизма и гражданской ответственности.

Педагогическая целесообразность программы. Дополнительная общеобразовательная программа «Гейм-мастер: Godot Engine» способствует повышению уровня информационной культуры посредством творческой и практической деятельности в области информационных технологий.

В процессе реализации программы используются элементы педагогических технологий в соответствии с целями и задачами детского объединения.

В целях развития познавательной и творческой активности, учащихся используется технология проблемного обучения, при этом перед учащимися последовательно и целенаправленно ставятся познавательные задачи, решая которые учащиеся активно осваивают знание и опыт познавательной деятельности. Для того, чтобы ориентировать учебный процесс на потенциальные возможности учащихся и их реализацию применяется технология развивающего обучения. Технология дифференцированного обучения создает оптимальные условия для выявления,

развития интересов и способностей учащихся. Информационные технологии используются как средство обучения и как инструмент автоматизации учебной деятельности. Эффективно используются и другие технологии: личностно – ориентированного обучения, саморазвития, игрового обучения и другие.

Цели дополнительной общеобразовательной программы. Основная цель программы: создание условий для формирования у обучающихся компетенций в области 2D-разработки игр на базе игрового движка Godot Engine и языка программирования GDScript, а также для их профессиональной ориентации в сфере IT через разработку собственных игровых проектов.

Задачи:

обучающие:

- Сформировать у обучающихся представление об архитектуре игрового движка Godot Engine.
- Обеспечить освоение синтаксиса и базовых конструкций языка GDScript
- Научить применять инструменты 2D-физики, системы сигналов (Signals) и обработки пользовательского ввода для реализации игровых механик.
- Сформировать навыки работы с тайловыми картами (TileMap).
- Познакомить с методологией разработки программного обеспечения.

развивающие:

- Способствовать развитию алгоритмического, системного и проектного мышления через декомпозицию сложных игровых задач на логические модули.
- Стимулировать познавательный интерес, навыки самостоятельной работы и поиска решений при отладке кода и реализации нестандартных игровых механик.
- Развивать навыки планирования проектной деятельности, включая составление дизайн-документа (GDD) и структурного плана проекта.

воспитательные:

- Формирование интереса к технической деятельности, истории техники в России и Самарской области, понимание значения техники в жизни российского общества, уважения к достижениям в технике своих земляков.
- Стимулирование познавательного интереса, навыков самостоятельной работы и поиска решений при отладке кода и реализации нестандартных игровых механик.
- Развитие навыков планирования проектной деятельности, включая составление дизайн-документа (GDD) и структурного плана проекта.

Возраст учащихся: 12-15 лет. Возраст 12–15 лет является оптимальным для освоения

Godot Engine, так как у подростков в этот период активно развивается абстрактно-логическое мышление, позволяющее понимать сложные концепции движка. К этому возрасту у них уже есть базовая алгоритмическая подготовка, необходимая для перехода к текстовому программированию на GDScript и работы с 2D-физикой. Кроме того, в этом возрасте высок интерес к созданию собственного цифрового продукта. Возможность разработать полноценную игру служит мощным мотивирующим фактором. Программа, сфокусированная на проектной деятельности, полностью соответствует как интеллектуальным возможностям, так и психологическим потребностям учащихся 12–15 лет..

Срок реализации программы: 1 год

Объем программы

Календарный учебный график

№	Год обучения	Кол-во недель в год	Кол-во часов в год
1.	1 год обучения	42	168

Год обучения	Начало учебного года	Окончание учебного года	Количество учебных недель в год	Продолжительность учебного часа	Кол-во учебных занятий в неделю
1	01.09	31.08	42	40 мин.	2

Формы организации образовательного процесса: групповая

Формы обучения: очная, с применением дистанционных технологий

Режим занятий: занятия по образовательной программе проводятся 2 раза в неделю по 2 часа.

Планируемые результаты

Личностные: устанавливать связь между целью учебной деятельности и ее мотивом; определять общие для всех правила поведения; оценивать усваиваемое содержание учебного материала исходя из личностных ценностей; ориентация на понимание причин успеха в творческой деятельности; устанавливать связь между целью деятельности и ее результатом.

Метапредметные. Познавательные: ориентироваться в своей системе знаний (определять границы знания/незнания); находить ответы на вопросы в тексте, иллюстрациях, используя свой жизненный опыт; проводить анализ учебного материала; проводить сравнение, объясняя критерии сравнения; уметь определять

уровень усвоения учебного материала.

Регулятивные: определять и формулировать цель своей деятельности; формулировать учебные задачи; работать по предложенному плану, инструкции; высказывать свое предположение на основе учебного материала; осуществлять итоговый и пошаговый контроль в своей творческой деятельности; вносить необходимые коррективы в действие после его завершения на основе оценки в характере сделанных ошибок; осуществлять поиск информации с использованием литературы и сети Интернет.

Коммуникативные: слушать и понимать речь других; уметь с достаточной полнотой и точностью выражать свои мысли; владеть диалогической формой речи в соответствии с грамматическими и синтаксическими нормами родного языка; сотрудничать и оказывать взаимопомощь, доброжелательно и уважительно строить свое общение со сверстниками и взрослыми; формировать собственное мнение и позицию.

Предметные:

Учащийся должен знать:

- Архитектуру игрового движка Godot Engine, включая концепцию организации проекта на основе сцен (Scene) и узлов (Node).
- Базовый синтаксис языка GDScript, типы данных, правила объявления и использования переменных.
- Принципы управления потоком выполнения программы с помощью условных конструкций (if/else) и циклов (for, while).
- Назначение и ключевые методы встроенных функций движка, таких как `_process(delta)` и `_physics_process(delta)`.
- Принципы работы 2D-физики в Godot, включая использование узлов `CharacterBody2D` и `StaticBody2D`, а также методы обнаружения столкновений.
- Методологию создания игровых уровней с использованием узлов `TileMap` и настройки коллизий.
- Принципы работы с пользовательским интерфейсом (UI) на основе узлов `Control`.
- Основы тестирования программного обеспечения, классификацию видов тестирования и жизненный цикл дефекта («бага»).

Учащийся должен уметь:

- Ориентироваться в интерфейсе Godot Engine, создавать и настраивать новый 2D-проект.
- Создавать иерархию узлов, сохранять и инстанцировать сцены для построения

структуры игры.

- Импортировать и настраивать ассеты (изображения, звуки), привязывая их к соответствующим узлам в сцене.
- Писать скрипты на GDScript для реализации логики поведения игровых объектов, включая управление свойствами, обработку условий и циклов.
- Реализовывать движение персонажа с учетом физики, включая бег, прыжки и инерцию, используя `CharacterBody2D` и встроенные методы проверки столкновений.
- Создавать игровые уровни в редакторе `TileMap`, настраивать коллизии для платформ и препятствий.
- Программировать игровые механики, такие как сбор предметов (с использованием `Area2D` и сигналов), простой искусственный интеллект для врагов и взаимодействие объектов.
- Разрабатывать адаптивный пользовательский интерфейс (меню, индикаторы), используя узлы `Control`, `Label` и `TextureButton`.
- Применять методы отладки для поиска и исправления ошибок в коде с помощью вывода в консоль.
- Планировать, реализовывать и тестировать индивидуальный игровой проект от концепции до готового прототипа.
- Проводить взаимное тестирование (peer review) проектов, анализировать обратную связь и готовить проект к публичной защите.

Критерии и способы определения результативности

Для того чтобы оценить усвоение программы, в течение года используются следующие методы диагностики: собеседование, наблюдение, анкетирование, выполнение творческих заданий, анализ практических работ, тестирование, участие в конкурсах и викторинах.

По завершению учебного плана каждого раздела оценивание знаний проводится посредством викторины, анализа творческой (практической) работы, проекта.

Применяется 3-бальная система оценки знаний, умений и навыков обучающихся (выделяется три уровня: ниже среднего, средний, выше среднего) Итоговая оценка результативности освоения программы проводится путем вычисления среднего показателя, основываясь на суммарной составляющей по итогам освоения всех разделов программы.

Уровень освоения программы ниже среднего - ребенок овладел менее чем 50% предусмотренных знаний, умений и навыков, испытывает серьезные затруднения при работе с учебным материалом; в состоянии выполнять лишь простейшие практические задания педагога.

Средний уровень освоения программы - объем усвоенных знаний, приобретенных умений и навыков составляет 50-70%; работает с учебным материалом с помощью педагога; в основном выполняет задания на основе образца; удовлетворительно владеет теоретической информацией по темам курса.

Уровень освоения программы выше среднего предполагает освоение учащимся на 70 - 100% материала, предусмотренного программой; работает с учебными материалами самостоятельно, не испытывает особых трудностей; выполняет практические задания с элементами творчества; свободно владеет теоретической информацией по курсу; умеет анализировать литературные источники, применять полученную информацию на практике.

Оценочные материалы

№ п/п	Показатель	Диагностический инструментарий
1	Количественный	Анализ данных списочного состава
2	Результат образовательной деятельности	Собеседование, наблюдение, интерактивное занятие, анкетирование, выполнение творческих заданий, выполнение практических работ, тестирование, участие в конкурсах, викторинах в течение года.
3	Результаты исследования развития учащихся	- Методика В.С. Юркевич (изучение познавательной активности учащихся) - методика Д.Ж. Морено (изучение межличностных и межгрупповых отношений) - методика О.В. Соловьёва (изучение учебно-познавательной деятельности) - методика Р.В. Овчаровой (изучение коммуникативных склонностей) - методика Ю.М. Орлова (изучение потребности в достижении цели)

2. Учебный план

1 год обучения

№ п/п	Перечень разделов (модулей) содержания образования	Количество часов		
		Всего	Теория	Практика
1.	Знакомство с интерфейсом и структурой проекта	40	12	28
2.	Основы языка GDScript и скриптинг	40	10	30
3.	Разработка аркадных игр	30	8	22
4.	Разработка платформера	40	12	28
5.	Итоговый проект и защита	18	4	14
	Итого:	168	46	122

3. Содержание

Раздел 1. Знакомство с интерфейсом и структурой проекта

Тема 1. Введение в Godot Engine и создание первого проекта

Теория: Правила поведения в компьютерном классе, организация рабочего места, нормы СанПиН при работе за ПК. Краткий обзор вклада отечественных ученых и инженеров в мировую IT-индустрию. Примеры успешных российских IT-компаний и продуктов. Обзор профессий в IT (программист, геймдизайнер, 3D-художник, QA-инженер). Кто такой разработчик игр и чем он занимается? Что такое игровой движок? Обзор экосистемы Godot: версии 3.x и 4.x. Преимущества. Установка движка. Обзор интерфейса.

Практика: Викторина «Угадай IT-профессию». Тест «Кем ты хочешь стать в IT?». Установка ПО (Godot Engine) и создание первого проекта. Запуск Godot. Создание нового 2D-проекта. Обзор интерфейса: панели «Сцены», «Инспектор», «Файловая система», «Вывод», «Сценарий». Создание первой 2D-сцены. Настройка интерфейса, изменение местоположения окон, настройка темы.

Тема 2. Иерархия «Сцены» и «Узлы»

Теория: Ключевая концепция Godot: всё является узлом (*Node*). Сцена как коллекция узлов. Древо сцен (*Scene Tree*). Инстанцирование (создание экземпляров) сцен.

Практика: Создание иерархии узлов. Сохранение сцены. Создание одной сцены и её добавление в качестве дочернего узла в другую сцену. Запуск сцены.

Тема 3. Работа с ресурсами и ассетами

Теория: Понятие ресурса (*Resource*) в Godot. Импорт изображений, звуков, шрифтов. Свойства импорта (сжатие, фильтрация).

Практика: Импорт спрайтов для персонажа и фона. Настройка свойств изображения в «Инспекторе». Привязка ресурсов к узлам *Sprite*.

Раздел 2. Основы языка GDScript и скриптинг

Тема 4. Введение в GDScript: синтаксис и переменные

Теория: GDScript как язык, ориентированный на движок. Переменные и их области видимости. Базовые типы данных: *int*, *float*, *string*, *bool*, *Vector2*.

Практика: Создание первого скрипта и его привязка к узлу. Объявление переменных для хранения скорости, здоровья, ссылок на другие узлы.

Тема 5. Управление потоком выполнения: условия и циклы

Теория: Условные операторы (*if*, *elif*, *else*) для ветвления логики. Циклы (*for*, *while*) для повторения действий.

Практика: Написание скрипта, который меняет поведение объекта в зависимости от

его здоровья (if health < 0). Создание скрипта, который порождает 10 врагов в цикле.

Тема 6. Функции и методы в GDScript

Теория: Определение собственных функций (func). Ключевое слово self. Встроенные функции движка (_process(delta), _physics_process(delta)). Особенности встроенных функций.

Практика: Реализация движения персонажа внутри функции _physics_process(delta) для обеспечения плавности и независимости от частоты кадров.

Раздел 3. Разработка аркадных игр

Тема 7. Создание игры «Пинг-понг»

Теория: Проектирование игровой логики: подсчет очков, условия победы/поражения. Реализация движения объектов через изменение свойства position. Обработка столкновений через сигналы body_entered.

Практика: Реализация движения ракеток по вертикали. Программирование отскока мяча от стен и ракеток. Создание интерфейса для отображения счета.

Тема 8. Создание игры «Dodge the Creeps»

Теория: Принципы процедурной генерации контента. Движение врагов (мобов) с помощью PathFollow2D или простого изменения position. Реализация игровой петли (Game Loop) через _process().

Практика: Создание игрока, управляемого мышью или клавиатурой. Генерация врагов, движущихся к игроку. Реализация логики проигрыша при столкновении.

Раздел 4. Разработка платформера

Тема 9. Основы создания платформера: Движение и физика

Теория: Детальный разбор CharacterBody2D. Реализация плавного движения с учетом ускорения и трения. Алгоритм прыжка с учетом гравитации. Обнаружение «земли» (is_on_floor()).

Практика: Программирование контроллера персонажа: бег, прыжки, инерция.

Тема 10. Создание игрового мира: Тайловые карты и коллизии

Теория: Использование TileMap для создания уровней. Работа с TileSet'ом. Статические коллизии (StaticBody2D) для создания препятствий и платформ.

Практика: Отрисовка уровня в редакторе TileMap. Настройка коллизий для платформ

Тема 11. Игровые механики: Сбор предметов и враги

Теория: Взаимодействие объектов через зоны (Area2D). Сигналы входа/выхода из зоны для сбора монеток или активации бонусов. Простой ИИ для врагов (патрулирование, преследование).

Практика: Реализация сбора предметов (монет) с обновлением счета. Создание простого врага, который ходит между двумя точками.

Тема 12. Пользовательский интерфейс (UI)

Теория: Узлы UI: Control, Label, TextureButton. Якоря (Anchors) и контейнеры (Containers) для адаптивного интерфейса.

Практика: Создание меню паузы, экрана "Game Over" и таблицы рекордов.

Раздел 5. Итоговый проект и защита

Тема 13. Разработка концепции и планирование индивидуального проекта

Теория: Методология проектной деятельности. Этапы жизненного цикла программного продукта: инициация, планирование, реализация, тестирование. Принципы декомпозиции задачи на логические модули и системы (игровая логика, пользовательский интерфейс, управление ресурсами). Разработка технического задания (ТЗ) и дизайн-документа (GDD) как инструментов планирования.

Практика: Генерация и согласование идей для индивидуального проекта. Формулирование концепции игры, определение ключевых игровых механик, целевой аудитории и критериев успешности. Составление структурного плана проекта и первичного дизайн-документа.

Тема 14. Самостоятельная работа над проектом

Практика: Реализация индивидуального проекта в интегрированной среде разработки Godot Engine. Применение полного технологического стека: создание сцен, программирование логики на GDScript, работа с физикой, создание пользовательского интерфейса. Решение технических сложностей, оптимизацию алгоритмов и архитектурные вопросы.

Тема 15. Тестирование, отладка и подготовка к защите

Теория: Основы тестирования программного обеспечения: классификация видов тестирования (модульное, интеграционное, функциональное). Понятие «баг» (дефект), его жизненный цикл. Принципы отладки (debugging) кода с использованием встроенных инструментов Godot (вывод в консоль, точки останова).

Практика: Проведение взаимного тестирования проектов (*peer review*) с целью выявления ошибок и уязвимостей в логике. Анализ полученной обратной связи, исправление выявленных дефектов (отладка). Подготовка демонстрационных материалов и текста выступления для защиты проекта.

Тема 16. Публичная защита проекта

Практика: Презентация результатов проектной деятельности перед аудиторией. Демонстрация функциональности разработанной игры. Обсуждение архитектурных решений, примененных технологий и возникших в ходе работы трудностей. Ответы на вопросы аудитории и преподавателя. Подведение итогов курса.

4. Воспитание

Цель, задачи, целевые ориентиры воспитания учащихся

Целью воспитания является развитие личности, самоопределение и социализация учащихся на основе социокультурных, духовно-нравственных ценностей и принятых в российском обществе правил и норм поведения в интересах человека, семьи, общества и государства, формирование чувства патриотизма, гражданственности, уважения к памяти защитников Отечества и подвигам Героев Отечества, закону и правопорядку, человеку труда и старшему поколению, взаимного уважения, бережного отношения к культурному наследию и традициям многонационального народа Российской Федерации, природе и окружающей среде.

Задачи воспитания по программе являются:

- усвоение учащимися знаний норм, духовно-нравственных ценностей, организация общения между ними на содержательной основе целевых ориентиров воспитания;
- формирование интереса к технической деятельности, истории техники в России и Самарской области, понимание значения техники в жизни российского общества, уважения к достижениям в технике своих земляков;
- формирование и развитие личностного отношения учащихся к учебным занятиям, к собственным нравственным позициям и этике поведения в детском коллективе;
- приобретение учащимися опыта участия в технических проектах и их оценки;
- социализация, самореализация при освоении предметного и метапредметного содержания программы.

Целевые ориентиры воспитания учащихся по программе:

- воспитание уважения к жизни, достоинству, свободе каждого человека, понимания ценности жизни, здоровья и безопасности (своей и других людей), развитие физической активности;
- формирование ориентации на солидарность, взаимную помощь и поддержку, особенно поддержку нуждающихся в помощи;
- воспитание уважения к труду, результатам труда, уважения к старшим.

Формы и методы воспитания

Решение задач просвещения учащихся, создания и поддержки воспитывающей среды общения и успешной деятельности, формирования межличностных отношений на основе российских традиционных духовных ценностей осуществляется на каждом учебном занятии.

Основной формой воспитания учащихся при реализации программы является организация их взаимодействия на учебных занятиях при решении технических упражнений, заданий, разработке и реализации проектов. Значимую роль в воспитании имеет участие в городских, областных и Всероссийских конкурсах технического творчества, в массовых мероприятиях от учрежденческого до областного уровней. В воспитательной деятельности с учащимися по программе используются методы воспитания: метод убеждения (рассказ, разъяснение, внушение), метод положительного примера (педагога и других взрослых, детей), метод упражнений, метод переключения деятельности, метод развития самоконтроля и самооценки учащихся, стимулирования и поощрения.

Работа с родителями (законными представителями) осуществляется в форме:

- родительских собраний;
- открытых занятий для родителей;
- творческого взаимодействия при подготовке и проведении различных массовых мероприятий.

Диагностика результатов воспитательной работы осуществляется с помощью:

- педагогического наблюдения;
- отзывов, опросов родителей (законных представителей), беседы с учащимися, отзывы других участников мероприятий и др.

Воспитательная работа осуществляется на основной учебной базе во Дворце творчества детей и молодежи в рамках учебных занятий, массовых мероприятий.

Календарный план воспитательной работы

№	Наименование мероприятия	Дата	Практический результат
1	«Дни открытых дверей»	сентябрь	Фотоматериалы с выступлением учащихся
2	Участие в конкурсах, чемпионатах по информатике	В течении года	Анализ выступлений. Грамоты, сертификаты
3	Новогодний праздник «Серебряный ключик»	Декабрь	Фотоматериалы
4	Встреча с ветеранами СВО	Февраль	Фотоматериалы
5	Родительские собрания «Роль дополнительного образования в воспитании учащихся»	Сентябрь	

	«Итоги работы за год»	Май	
6	Городской праздник «Широкая масленица»	Февраль-март	Фотоматериалы
7	Участие в акции «Мы помним, мы гордимся»	май	Фотоматериалы
8	Участие в общеучрежденческих мероприятиях	В течении года	Фотоматериалы

5. Ресурсное обеспечение программы

Информационно-методическое обеспечение

Используются следующие формы организации образовательного процесса:

Образовательная деятельность	Формы организации
Учебная деятельность	Теоретические и практические занятия, занятие-беседа, видеоурок, занятие-соревнование, занятие-импровизация, конкурсы, олимпиады, проектная деятельность
Воспитательная деятельность	Защита проектов, учрежденные и городские массовые мероприятия, социально-значимые мероприятия

Методические материалы

№ п/п	Деятельность	Средства
1	Методы обучения	Словесный, наглядный, практический, игровой и т.д.
2	Методы воспитания	Убеждения, поощрения, стимулирования, мотивации
3	Педагогические технологии	Развивающего обучения, дифференцированного обучения, игрового обучения, личностно-ориентированного обучения, дистанционные образовательные технологии и т. д.
4	Дидактические материалы	Раздаточные материалы, задания, упражнения и т.д.

Для реализации образовательной программы используются все основные виды методической продукции. Для разъяснения приемов и методов, анализа опыта, описания педагогических технологий, пропаганды наиболее важных и актуальных направлений педагогической деятельности разработаны информационно-пропагандистская методическая продукция (методическое описание, аннотация, информационный плакат, информационно – методическая выставка). Для указания и разъяснения цели и порядка действия, технологии и методики организации образовательного процесса, проведения мероприятий разработана организационно-инструктивная методическая продукция (инструктивно – методическое письмо, методическая записка, методическая памятка, методическая рекомендация, методическая разработка, тематическая папка). С целью иллюстрации и более полного раскрытия темы, отраженную в других видах методической продукции, разрабатываются прикладная методическая продукция (сценарий, тематическая подборка, картотека, положения о соревнованиях, учебно–методический материал). Методическое обеспечение образовательной программы соответствует современным требованиям и обеспечивает высокий уровень результативности образовательной деятельности.

Материально – техническое обеспечение

Компьютерная техника: персональные компьютеры или ноутбуки в количестве, соответствующем числу учащихся; мониторы, клавиатура и мышь для каждого рабочего места, наушники с микрофоном для индивидуальной работы.

Программное обеспечение: игровой движок *Godot Engine*, веб-браузер, графический редактор, аудиоредактор.

Сетевое и периферийное оборудование: локальная сеть с высокоскоростным доступом в интернет; проектор и экран или интерактивная доска для демонстрации учебного материала; принтер или МФУ для печати учебных пособий и инструкций.

Организация рабочего пространства: столы и стулья, соответствующие санитарно-гигиеническим нормам для учащихся; кабинет с хорошим уровнем естественного и искусственного освещения и исправной системой вентиляции.

6. Список литературы

Основная литература

1. Куликовский, А. А. Изучаем Godot 4. Создание игр на языке GDScript / А. А. Куликовский. — Санкт-Петербург : БХВ-Петербург, 2024. — 384 с.
2. Bradfield, C. Godot 4 Game Development Projects: Build 3D and 2D games using Godot 4, C#, and GDScript / C. Bradfield. — Packt Publishing, 2023. — 650 p.

Нормативные и технические документы

3. Официальное руководство по Godot 4.x : документация / *Godot Engine contributors*. — [Godot Engine], 2024. Режим доступа: <https://docs.godotengine.org/en/stable> (дата обращения: 27.05.2026).

Дополнительная литература

4. Johnson, C. Godot 4 Game Development Cookbook: Over 50 practical recipes to leverage the latest Godot 4 features / C. Johnson. — Packt Publishing, 2023.
5. Бура, М. Game Programming Patterns / М. Бура. — *Game Programming Patterns*, 2014. Режим доступа: <https://gameprogrammingpatterns.com> (дата обращения: 27.05.2026).