


Департамент образования администрации города Южно-Сахалинска
Муниципальное автономное общеобразовательное учреждение
Гимназия № 3 города Южно-Сахалинска
Детский технопарк «Кванториум»

РАССМОТРЕНО на заседании
Методического совета
Протокол от 23.05.2022 № 4

УТВЕРЖДАЮ
Директор

 А.В. Умнова
Приказ от 25.05 2022 № 453

Дополнительная общеобразовательная общеразвивающая программа
«Junior в IT»

Направленность: техническая
Уровень сложности: базовый
Адресат программы: 10-18 лет
Срок реализации: 9 мес.
Автор-составитель:
Зайцев Евгений Сергеевич,
педагог дополнительного образования

г. Южно-Сахалинск
2022

Оглавление

Целевой раздел	3
Пояснительная записка.....	3
Содержательный раздел	8
Учебный план программы.....	8
Содержание учебной программы	8
Система оценки достижения планируемых результатов освоения программы	9
Календарный учебный график.....	10
Организационный раздел	11
Методическое обеспечение программы	11
Материально-техническое обеспечение программы.....	14
Кадровое обеспечение программы.....	15
Приложение 1	16
Приложение 2	18
Приложение 3	23
Приложение 4	28
Приложение 5	33
Приложение 6	40
Приложение 7	43
Приложение 8	46
Приложение 9	49

Целевой раздел Пояснительная записка

Настоящая дополнительная общеобразовательная общеразвивающая программа **«Junior в IT»** (далее - программа, ДООП) реализуется в рамках образовательного проекта «Детский технопарк «Кванториум» (ДТ Кванториум, Кванториум) и составлена с учетом следующих нормативно-правовых актов:

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»
2. Концепция развития дополнительного образования детей до 2030 года (утв. Распоряжением Правительства Российской Федерации от 31.03.2022 № 678-р)
3. Распоряжение Правительства РФ от 29.05.2015 № 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»
4. Постановление Главного государственного санитарного врача РФ от 28.01.2021 № 2 «Об утверждении санитарных правил норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»
5. Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»
6. Приказ Министерства труда и социальной защиты Российской Федерации от 05.05.2018 № 298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»
7. Приказ Министерства просвещения Российской Федерации от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
8. Приказ Министерства образования и науки Российской Федерации от 23.08.2017 № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»
9. Письмо Министерства просвещения Российской Федерации от 31.01.2022 № ДГ-245/06 «О направлении методических рекомендаций» (вместе с «Методическими рекомендациями по реализации дополнительных общеобразовательных программ с применением электронного обучения и дистанционных образовательных технологий»)
10. Письмо Министерства образования и науки Российской Федерации от 18.11.2015 № 09-3242 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»)
11. Письмо Министерства образования и науки Российской Федерации от 11.12.2006 № 06-1844 «О примерных требованиях к программам дополнительного образования детей»
12. Распоряжение Министерства Образования Сахалинской области от 16.09.2021 № 3.12-1170-р «Об утверждении методических рекомендаций по проектированию и реализации дополнительных общеобразовательных общеразвивающих программ» (вместе с «Методическими рекомендациями по проектированию и реализации дополнительных общеобразовательных общеразвивающих программ»)
13. Устав МАОУ Гимназия № 3 города Южно-Сахалинска
14. Иные локальные нормативные акты МАОУ Гимназия № 3 г. Южно-Сахалинска

Направленность программы - техническая.

Тип программы - одноуровневая, модифицированная, модульная.

Уровень сложности программы - базовый уровень. Предполагает формирование у обучающихся базовых компетенций, характерных для данного вида деятельности.

Язык реализации программы - государственный язык РФ – русский.

Актуальность программы

Динамично развивающиеся цифровые технологии достигли удивительных возможностей. Сегодня сложно себе представить какую-либо область человеческой деятельности без применения технических средств и сети Интернет.

Все, что нас окружает - наполнено информацией и технологиями, в которые мы вовлечены. От качества организации этой среды зависит наше восприятие процессов, которые с нами происходят. Задача начинающего IT-специалиста понять устройство сферы информационных технологий и спроектировать среду вокруг пользователя так, чтоб она была более понятна обычному обывателю. На сегодняшний день IT-специалист не просто проектирует удобную и технологичную вещь или информационную среду, он проектирует основной пользовательский опыт взаимодействия потребителя с информационной средой в целом.

В условиях свободной конкуренции потребитель становится все более разборчивым и требовательным к качеству услуг, сервиса, предметного мира и среды, окружающей его. Промышленность всегда реагирует на меняющиеся запросы потребителей. Поэтому, востребованность специалистов, способных обеспечить это качество будет постоянно расти.

Актуальность данной программы также состоит в том, что она составлена с учётом современных потребностей рынка в специалистах в области информационной технологий. Учитывается и междисциплинарность информационных технологий. Предусмотрено приобретение навыков в области применения информационных технологий в робототехнике, дизайне.

В современном мире без информационных технологий сейчас не обходится ни одна сфера, и раннее обучение детей в данном направлении позволит им без труда входить в информационно-технологическую сферу любого предприятия.

Актуальность для муниципалитета и региона заключается в том, что позволит формировать заинтересованные в IT-сфере кадры (которых в современном обществе сейчас недостаток), которые в дальнейшем можно будет привлекать на рабочие места в муниципалитете.

Для обучающихся значимым открытием станет осознание, что в их силах овладеть любыми технологиями, которые позволят эффективнее выполнять любые задачи на протяжении всей жизни.

Отличительные особенности

Программа представляет собой переработанную и углубленную версию ранее реализуемой в ДТ Кванториум программы «IT – квантум. Базовый модуль: Информационные системы и технологии». В настоящей программе существенно увеличено количество часов, что позволит более детально осуществить подготовку обучающихся по выбранному направлению. Содержание материала также расширено и структурировано таким образом, что дает возможность познакомить проявивших интерес к данной области ребят с большим набором знаний и компетенций, необходимых специалисту по информационным технологиям. Больше внимание в настоящей программе уделяется и развитию необходимых в современном мире кросс-контекстных и экзистенциальных навыков.

Так же отличительной особенностью программы является параллельная реализация модулей программы по выбору обучающихся. Модули: Web-разработка; Инженерный модуль; Game-Dev модуль, реализуются параллельно, так как современные технологии в разработке информационной и методической базы в IT-сфере, а также методика, особенности составления плана занятий педагогом, используемая при реализации программы технология наставничества по формам

«обучающийся-обучающемуся» и «педагог-обучающемуся» позволяют курировать и проводить занятия параллельно по 3м разным темам.

Новизна программы заключается:

Новизна программы состоит в том, что она учитывает новые технологические уклады, который требует новый способ мышления и тесного взаимодействия при постоянном повышении уровня междисциплинарности проектов, а также использует принципы вытягивающей модели обучения.

Введение в дополнительное образование общеобразовательной и общеразвивающей программы «Junior в IT» с использованием таких методов, как командная работа, поиск проблем и их практические решение, анализ и обобщение опыта, подготовка исследовательских и инженерно-технических проектов и их защита, элементы соревнований, неизбежно изменит картину восприятия обучающимися IT-дисциплин, переводя их из разряда умозрительных в разряд прикладных.

В программе так же используются электронные методы обучения, что обеспечивает её гибкость и возможность применения в условиях дистанционной работы.

Так же данная программа учитывает особенности обучения специалистов по узким направленностям, которые сейчас очень популярны и востребованы на рынке труда, такие как: Разработчик видеоигр (от level-дизайнера до Альфа- и Бета- тестера), Веб-разработчик (от создания сайтов до создания веб-приложений и интерфейсов), специалист IoT. Это открывает широкие возможности для раннего самоопределения подрастающего поколения и позволяет совершить профессиональные пробы, узнать больше о современном развитии области.

Адресат программы

Программа ориентирована на дополнительное образование обучающихся 10-18 лет, проявляющих интерес к программированию, конструированию, информационным технологиям в целом, стремящихся к саморазвитию, профессиональному самоопределению, освоивших первоначальные навыки программирования и логики работы программного обеспечения для разработки. К зачислению на программу допускаются дети, прошедшие программы IT-квантума, а также дети, успешно сдавшие входной тест (пример теста приведён в Приложении 5). Группы формируются по разновозрастному принципу. При таком способе комплектования детского коллектива наилучшим образом реализуются технологии наставничества.

Программа не предназначена для освоения детьми с ОВЗ.

Максимальное количество обучающихся в группе - 15 человек.

Форма обучения – очная. При необходимости (в случае введения ограничительных мер на реализацию ДООП в очном формате, связанных с санитарно-эпидемиологической обстановкой, чрезвычайными ситуациями и т.п.) возможно применение дистанционных образовательных технологий и элементов электронного обучения, с использованием материалов, соответствующих содержанию программы.

Методы обучения

Особенностью организации образовательного процесса в настоящей программе является использование преимущественно современных методов обучения:

Кейс-метод. Непосредственная цель метода case-study: совместными усилиями группы о проанализировать ситуацию - «case», возникающую при конкретном положении дел, и выработать практическое решение. Решение кейса позволяет развить навыки критического мышления, актуализирует определенный комплекс знаний, который необходимо усвоить при решении заявленной проблемы.

Метод проектов - это совокупность приемов, действий обучающихся в их определенной последовательности для достижения поставленной задачи - решения проблемы, лично значимой для

обучающихся и оформленной в виде некоего конечного продукта. Проектный метод объединяет исследовательские, поисковые, проблемные, творческие методы.

Тип и формы организации занятий

Программой предусмотрено проведение аудиторных и внеаудиторных занятий: лекции, опросы, практические работы, решение кейсов, проектные работы, экскурсии, мастер классы.

Формы организации деятельности - групповая, работа в микро-группах, работа в парах, индивидуальная, всем составом объединения

Итоговый контроль проводится по результатам защиты кейсов или проектов.

При необходимости организации работы в дистанционном режиме используются следующие типы занятий и формы деятельности:

- учебное занятие взаимодействия педагога и обучающихся - исключительно в электронной форме и с применением ДОТ (в формате видеоконференций, вебинаров, онлайн лекций и т.п.)
- учебное занятие самостоятельной работы учащихся - самостоятельная работа обучающихся оффлайн (самостоятельное изучение учебного материала, выполнение заданий педагога, работа на образовательных платформах, сайтах, посещение виртуальных туров и экскурсий, мастер-классов и т.п.)
- контрольные учебные занятия - контрольные срезы, тестирования, зачеты в режимах онлайн и оффлайн, дистанционные конкурсы и т.п.)
- консультации (индивидуальные и групповые) в режиме онлайн и оффлайн.

Для работы обучающимся предоставляются разработанные информационные материалы: обучающие видео - и аудиоматериалы, ссылки на источники информации, направляются вопросы, тесты, кейсы, практические задания и т.д. В обязательном порядке предусмотрена обратная связь от педагога каждому ребенку. Все используемые информационные материалы, направляемые обучающимся, соответствуют целям, задачам, содержанию реализуемой программы и возрастным особенностям обучающихся.

Объём программы – 144 академических часа.

Срок освоения программы – 9 месяцев, 36 недель.

Срок реализации программы - сентябрь - май.

Режим занятий – два раза в неделю по два академических часа с 10-минутным перерывом.

Продолжительность одного академического часа - 40 минут. При необходимости организации работы в дистанционном режиме академический час равняется 30 минутам.

Цель программы

Основная цель данной программы - профессиональная ориентация обучающихся через формирование устойчивой положительной мотивации к занятиям информационными технологиями

Задачи программы

<i>Предметные результаты</i>	<i>Метапредметные результаты</i>	<i>Личностные результаты</i>
1) Дать представление об устройстве компьютеров и компьютерных сетей, их истории и современных тенденциях развития; 2) Дать знания об основных принципах программирования и навыки написания среднего уровня сложности программ; 3) Выработать навыки чтения кода, самостоятельного проектирования и исправления; 4) Обучить программировать в различных средах; 5) Дать знания об устройстве всемирной глобальной сети, основных принципах ее функционирования.	1) Способствовать развитию собственных познавательных интересов; 2) Научить подстраивать своё мышление под определенные задачи и проекты (дизайн мышления), отходить от плана или придерживаться инструкций; 3) Привить творческий подход к поставленной задаче; 4) Дать знания основ проектной и исследовательской деятельности;	1) Выработать навыки работы в команде; 2) Развить коммуникативность; 3) Натренировать ораторское искусство; 4) Развить креативность; 5) Привить умение самостоятельно находить решения при столкновении со сложными задачами; 6) Развить стремление к получению новых знаний самостоятельно.

Планируемые результаты

<i>Предметные результаты</i>	<i>Метапредметные результаты</i>	<i>Личностные результаты</i>
1) Получат знания об устройстве компьютеров и компьютерных сетей, их истории и современных тенденциях развития. 2) Будут иметь знания об основных принципах программирования и навыки написания среднего уровня сложности программ. 3) Получат навыки чтения кода, самостоятельно проектирования и исправления. 4) Смогут программировать в различных средах. 5) Приобретут знания об устройстве всемирной глобальной сети, основных принципах ее функционирования.	1) Будут проявлять собственный познавательный интерес; 2) Научатся подстраивать своё мышление под определенные задачи и проекты (дизайн мышление), отходить от плана или придерживаться инструкций; 3) Будут демонстрировать творческий подход к поставленной задаче; 4) Получат знания основ проектной и исследовательской деятельности.	1) Проявят навыки работы в команде. 2) Будут демонстрировать коммуникативность. 3) Продемонстрируют ораторское искусство. 4) Будут проявлять креативность. 5) Продемонстрируют умение самостоятельно находить решения при столкновении со сложными задачами. 6) Проявят стремление к получению новых знаний самостоятельно.

**Содержательный раздел
Учебный план программы¹**

№ п/п	Название модуля	Количество часов			Формы контроля
		Всего	Теория	Практика	
Модуль 1. (инвариантный)	Знакомство с ДТ Кванториум	18	5	13	Педагогическое наблюдение. Беседа. Анкетирование Игра.
Модуль 2. (по выбору)	Web-разработка	126	24	102	Презентация проекта или получившегося кейса
Модуль 3. (по выбору)	Инженерный модуль	126	30	96	Презентация проекта или получившегося кейса
Модуль 4. (по выбору)	Game-Dev модуль	126	24	102	Презентация проекта или получившегося кейса
	Всего часов	144			

Содержание учебной программы

№ п/п	Название модуля	Теоретическая часть	Практическая часть
1.	Модуль 1. Знакомство с ДТ Кванториум	Ознакомление с деятельностью детского технопарка.	Мастер-классы по направлениям.
2.	Модуль 2. Web-разработка	Основы веб-разработки и серверного администрирования веб приложений	Выполнение практических заданий. Работа над кейсом или проектом в данном направлении. Разработка собственного сайта или веб приложения.
3.	Модуль 3. Инженерный модуль	Основы схемотехники и серверного администрирования IoT приложений и устройств	Выполнение практических заданий. Работа над кейсом или проектом в данном направлении. Разработка своего предмета мира IoT.
4.	Модуль 4. Game-Dev модуль	Основы разработки видеоигр и серверного администрирования сетевой части игры	Выполнение практических заданий. Работа над кейсом или проектом в данном направлении. Создание начального прототипа или выпуск собственного проекта игры уровня бетта- теста.

¹ Программы модулей представлены в Приложениях 1-4

Система оценки достижения планируемых результатов освоения программы

Реализация программы предусматривает различные виды контроля: входной, текущий и итоговый.

Текущий контроль проводится планомерно в течение всего периода обучения и направлен на определение степени усвоения обучающимися учебного материала, а также мотивированности и заинтересованности обучающихся в обучении. Проводится в форме: педагогического (включенного) наблюдения, результатов выполнения контрольных заданий и практических работ, презентаций, опрос, беседа.

Итоговый контроль организуется в конце срока обучения и нацелен на определение результатов обучения и изменений уровня развития обучающихся, их творческих способностей. Проводится в форме защиты кейса или проекта, над которым дети трудились в течение года.

Формы предъявления и демонстрации образовательных результатов обучающихся: презентации, защиты проектов или кейсов, конкурсы, соревнования.

Формы фиксации образовательных результатов: портфолио достижений обучающихся (характеристика, грамоты, сертификаты, дипломы, экспертные заключения и т.п), журнал, анкета, тест.

Оценочные материалы

Формы и методы диагностики для определения достижения планируемых результатов представлены в таблице 1.

Таблица 1

Планируемые результаты		Диагностический инструментарий (способы формы и методы)
Личностные	1) Проявят навыки работы в команде. 2) Будут демонстрировать коммуникативность. 3) Продемонстрируют ораторское искусство. 4) Проявят креативность. 5) Продемонстрируют умение самостоятельно находить решения при столкновении со сложными задачами. 6) Проявят стремление к получению новых знаний самостоятельно.	Педагогическое наблюдение; Анкетирование (методика «Пословицы», разработ. к.п.н С. М. Петровой; методика изучения степени социализации обучающихся, разработ. проф. М.И. Рожковым)
Метапредметные	1) Будут проявлять собственный познавательный интерес; 2) Научатся подстраивать своё мышление под определенные задачи и проекты (дизайн мышления), отходить от плана или придерживаться инструкций; 3) Будут демонстрировать творческий подход к поставленной задаче; 4) Получат знания основ проектной и исследовательской деятельности.	Педагогическое наблюдение; Личная беседа;
Предметные	1) Получат знания об устройстве компьютеров и компьютерных сетей, их истории и современных тенденциях развития. 2) Будут иметь знания об основных принципах программирования и навыки	Педагогическое наблюдение; Тесты; Практические задания; Презентация проектов или кейсов;

	написание среднего уровня сложности программ. 3) Получат навыки чтения кода, самостоятельно проектирования и исправления. 4) Смогут программировать в различных средах. 5) Приобретут знания об устройстве всемирной глобальной сети, основных принципах ее функционирования. различной сложности.	
--	---	--

Для оценки освоения обучающимися образовательной программы используется информационная карта (Приложение 5). Информация заносится в карту в соответствии с разработанными критериями.

Обучающиеся, успешно освоившие программу (с результатом высокий или средний уровень освоения) получают сертификат. Допускаются иные формы поощрения обучающихся (грамоты, дипломы и т.п.).

Календарный учебный график

Группа 1

Год обучения	Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во дней	Кол-во часов	Режим занятий
первый	05.09.2022	31.05.2023	36	72	144	2 раза в неделю по 2 ак. часа

Группа 2

Год обучения	Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во дней	Кол-во часов	Режим занятий
первый	05.09.2022	31.05.2023	36	72	144	2 раза в неделю по 2 ак. часа

Группа 3

Год обучения	Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во дней	Кол-во часов	Режим занятий
первый	05.09.2022	31.05.2023	36	72	144	2 раза в неделю по 2 ак. часа

Группа 4

Год обучения	Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во дней	Кол-во часов	Режим занятий
первый	05.09.2022	31.05.2023	36	72	144	2 раза в неделю по 2 ак. часа

Организационный раздел

Методическое обеспечение программы

Основная цель и задачи этой программы – профессионально ориентировать обучающихся, сформировать устойчивую положительную мотивацию к занятиям информационными технологиями, показать им, что направление интересно и перспективно. Задача педагога – развить у детей навыки, которые им потребуются в проектной работе и в дальнейшем освоении программ квантума. Все умения и навыки приобретаются только через опыт.

Поэтому в программе большое значение уделяется практике через кейс-технологии и метод проектов.

Кейс-технология – это техника обучения, использующая описание реальной ситуации, специально подготовленный материал с описанием конкретной проблемы, которую необходимо разрешить в составе группы. Кейс-технологии направлены на исследовательскую или инженерно-проектировочную деятельность. Интегрирует в себе технологию развивающего и проектного обучения. Выступают в обучении как синергетическая технология («погружение» в ситуацию, «умножение» знаний, «озарение», «открытие»). Позволяют создать ситуацию успеха.

Метод проектов – это совокупность приемов, действий обучающихся в их определенной последовательности для достижения поставленной задачи – решение проблемы, лично значимой для обучающихся оформленной в виде некоего конечного продукта. Проектный метод объединяет исследовательские, поисковые, творческие методы.

В основе реализации программы лежит **методический инструментарий**, составленный и разработанный специально для Детских технопарков Кванториум Фондом новых форм развития образования:

- Тулkit IT-квантум (методический инструментарий тьюторов);
- Основы проектной деятельности (сборник методические материалы);
- Детские инженерные и исследовательские проекты (методические материалы);
- Инженерные и исследовательские задачи (настольное приложение к учебно-методическому пособию для наставников).
- «Шпаргалка по дизайн-мышлению» (сборник методических материалов)
- «Учимся шевелить мозгами» (Общekomпетентностные упражнения и тренировочные задания);
- «Шпаргалка по рефлексии» (сборник методических материалов);

В качестве **дидактических материалов** в программе используются:

- Кейсы, разработанные педагогом (Приложения 6-8);
- обучающие видео;
- мультимедийные материалы;
- компьютерные программные средства;
- дидактические игры;
- платформы для набора информации, проведения тестирования и тренажеры:
 - Сайт для начинающих программистов. — Текст : электронный // Blockly: [сайт]. — URL: <http://blockly.ru/> (дата обращения: 04.05.2022).
 - Введение в C++ · Stepik / Мишин Вадим. — Текст : электронный // Stepik: [сайт]. — URL: <https://stepik.org/course/58723> (дата обращения: 04.05.2022).
 - Легкий старт в Java. Вводный курс для чайников · Stepik / Петр Михалевский. — Текст : электронный // Stepik: [сайт]. — URL: <https://stepik.org/course/90684> (дата обращения: 04.05.2022).
 - "Поколение Python" : курс для начинающих · Stepik / Тимур Гуев, Руслан Чаниев, Анри Табуев — Текст : электронный // Stepik: [сайт]. — URL: <https://stepik.org/course/58852> (дата обращения: 04.05.2022).
 - "Поколение Python" : курс для продвинутых · Stepik / Тимур Гуев, Руслан Чаниев,

Благотворительный фонд "Айкью Опшн"— Текст : электронный // Stepik: [сайт]. — URL: <https://stepik.org/course/68343> (дата обращения: 04.05.2022).

- HTML Academy: интерактивные онлайн-курсы по HTML, CSS и JavaScript/ Текст : электронный// HTML Academy: [сайт]. – URL: <https://htmlacademy.ru/> (дата обращения: 04.05.2022).
- База знаний Амперки [Амперка / Вики]/ Текст : электронный// Wiki Amperka: [сайт] – URL: <http://wiki.amperka.ru/> (дата обращения: 04.05.2022).

Дидактический материал подобран в соответствии с учебным планом (по каждому разделу, теме), возрастными и психологическими особенностями обучающихся, уровнем их развития и способностями.

Перечень рекомендуемых учебных изданий, интернет-ресурсов

Для педагогов:

1. 10 лучших методов генерирования идей / М. Полгин. — Текст : электронный // 4BRAIN: [сайт]. — URL: <https://cutt.ly/CfjS9Nd> (дата обращения: 04.05.2022).
2. Доусон, Майкл. Изучаем C++ через программирование игр / Доусон Майкл. — Д71. — Санкт-Петербург : Питер, 2016. — 353 с. — Текст : непосредственный.
3. Кёнинг, Эндрю, Му, Барбара. Эффективное программирование на C++. Серия C++ In-Depth, т. 2. : / Эндрю Кёнинг, Барбара Му. - Пер. с англ. — М. : Издательский дом “Вильямс”, 2002. — 384 с. : ил. — Парал. тит. англ. — Текст : непосредственный.
4. Лаворе, Роберт. Объектно-ориентированное программирование в C++ / Роберт Лаворе. — Санкт-Петербург : Питер, 2016. — 922 с. — Текст : непосредственный.
5. Методы прогнозирования. — Текст : электронный // Управление Производством: [сайт]. — URL: <https://up-pro.ru/encyclopedia/metody-prognozirovaniya/> (дата обращения: 04.05.2022).
6. Монахов, Вадим. Язык программирования Java и среда NetBeans (+ CD-ROM) / В. Монахов. - М.: БХВ-Петербург, 2012. - 720 с. — Текст : непосредственный.
7. Прата, Стивен. Язык программирования C#. Лекции и упражнения, 6-е изд. : Пер. с англ. — Москва : ООО “И.Д. Вильямс”, 2015. — 1024 с. : ил. - Парал. тит. англ. — Текст : непосредственный.
8. Рязанов, Иван. Основы проектной деятельности. Рязанов И. — М. : Фонд новых форм развития образования, 2017 — 52 с. — Текст : непосредственный.
9. Соммер, Улли. Программирование микроконтроллерных плат Arduino/Freeduino / У. Соммер. - Санкт-Петербург: BHV, 2016. - 256 с. — Текст : непосредственный.

Для родителей:

10. 10 лучших методов генерирования идей / М. Полгин. — Текст : электронный // 4BRAIN: [сайт]. — URL: <https://cutt.ly/CfjS9Nd> (дата обращения: 04.05.2022).
11. Рязанов, Иван. Основы проектной деятельности. Рязанов И. — М. : Фонд новых форм развития образования, 2017 — 52 с. — Текст : непосредственный.
12. Кёнинг, Эндрю, Му, Барбара. Эффективное программирование на C++. Серия C++ In-Depth, т. 2. : / Эндрю Кёнинг, Барбара Му. - Пер. с англ. — М. : Издательский дом “Вильямс”, 2002. — 384 с. : ил. — Парал. тит. англ. — Текст : непосредственный.

Для обучающихся:

13. Бриггс Джейсон. Python для детей. Самоучитель по программированию. - 1-е изд. — Москва : Манн, Иванов и Фербер, 2022. - 320 с. — Текст : непосредственный

14. Спрол Антон. Думай как программист. Креативный подход к созданию кода. C++ версия. - 1-е изд. – Москва : Бомбора, 2018. - 272 с. – Текст : непосредственный
15. Бонд Джереми Гибсон. Unity и C#. Геймдев от идеи до реализации. - 1-е изд. - Санкт-Петербург : Питер, 2022. - 928 с. – Текст : непосредственный

Система воспитательной работы

Воспитание обучающихся осуществляется в соответствии с Программой воспитания ДТ Кванториум и Календарным планом воспитательной работы на 2022-2023 учебный год.

Основной **целью воспитания** в объединении является развитие личности обучающегося посредством его самоопределения и социализации.

Достижению поставленной цели способствует решение ряда задач:

- способствовать формированию опыта самоопределения (личностного и профессионального) в разных сферах человеческой жизни посредством участия в профессиональных пробах;
- способствовать приобретению опыта социального взаимодействия и опыта участия в социально-одобряемой деятельности.

Практическая реализация цели и задач воспитания осуществляется через:

- использование в воспитании обучающихся потенциала ДООП и учебного занятия как источника поддержки и развития интереса к познанию, творчеству;
- реализацию потенциала педагога (собственные базовые (личностные и профессиональные) ценности) в деятельности по созданию условий для развития личности обучающихся;
- реализацию потенциала наставничества в воспитании обучающихся как основу взаимодействия людей разных поколений, мотивацию к саморазвитию и самореализации;
- содействие приобретению опыта личностного и профессионального самоопределения на основе личностных проб в совместной деятельности;
- реализацию воспитательных возможностей общих ключевых дел ДТ Кванториум, поддержку традиции их коллективного планирования, организации, проведения и анализа в сообществе, нацелено на формирование коллектива (Приложение 9);
- организацию работы с родителями или законными представителями, направленную на совместное решение проблем личностного развития детей;
- общение со спикерами, являющимися экспертами различной направленности – технической, естественнонаучной, социальной;
- участие в мероприятиях, направленных на общеразвивающие, социокультурные задачи и формирование интеллекта в целом.

Предполагаемые результаты воспитания:

- будет формироваться потребность в участии в социально-значимых и социально-одобряемых мероприятиях;
- будет сформирован коллектив ДТ Кванториум, в рамках которого приобретает опыт социально-значимых отношений друг с другом и с педагогом, с другими обучающимися посредством участия в ключевых общих делах
- будет создана предметно-пространственная среда, способствующая возможности совместной деятельности детей и взрослых, а также для развития творческого потенциала обучающегося и его способностей, интереса к познанию.

Достижение заявленных результатов отслеживается на основе проведения самоанализа воспитательного процесса. Основными направлениями самоанализа являются:

1. Результаты социализации и саморазвития обучающихся. Основным критерий для определения достижения результативности - динамика личностного развития.
2. Состояние организуемой предметно-пространственной среды. Основным критерий - наличие интересной событийно-насыщенной личностно-развивающей предметно-пространственной среды.

Способами получения информации являются педагогическое наблюдение, беседа с детьми, родителями, педагогами, а также при необходимости анкетирование родителей и детей, позволяющее измерить удовлетворенность воспитательным процессом.

В ходе изучения достижения результатов воспитательного процесса может быть использован следующий диагностический инструментарий:

1. Методика «Пословицы», разработ. к.п.н С. М. Петровой;
2. Методика изучения степени социализации обучающихся, разработ. проф. М.И. Рожковым;
3. Дифференциально-диагностический опросник (ДДО) Е.А. Климова;
4. Методика определения склонностей личности к различным сферам профессиональной деятельности Л.А. Йовайши
5. Методика изучения статусов профессиональной идентичности

Материально-техническое обеспечение программы

В целях обеспечения охраны здоровья и безопасных условий обучения и воспитания программой предусмотрено проведение обязательных **инструктажей по охране труда для обучающихся**. Изучение вопросов охраны труда, пожарной безопасности и электробезопасности проводится с целью формирования у обучающихся сознательного и ответственного отношения к вопросам личной безопасности и безопасности окружающих. Прохождение инструктажей обучающимися и соблюдение правил ТБ является обязательным.

Инструкции разработаны в соответствии с СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодёжи», Федеральным законом № 273-ФЗ от 29.12.2012 г. «Об образовании в Российской Федерации», Письмом Минобрнауки России № 12-1077 от 25.08.2015 г. «Рекомендации по созданию и функционированию системы управления охраной труда и обеспечением безопасности образовательного процесса в образовательных организациях, осуществляющих образовательную деятельность» и иными нормативно-правовыми актами, Уставом и Правилами внутреннего распорядка обучающихся МАОУ Гимназии № 3 г. Южно-Сахалинска.

В соответствии с действующим законодательством в ДТ Кванториум предусмотрены следующие виды инструктажей:

1. Вводный (ИОТ - 054 - 2022)
2. Первичный (ИОТ – 110 - 2022)
3. Повторный (ИОТ – 110 - 2022)
4. Внеплановый (ИОТ – 110 - 2022)
5. Целевой (ИОТ – 110 - 2022)

При организации учебных занятий соблюдаются гигиенические критерии допустимых условий и видов работ для ведения образовательной деятельности в соответствии с СанПиНом 1.2.3685-21 «Гигиенические нормативы требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания». Используемые для реализации программы помещения (учебные кабинеты, проектные, лектории) и их оснащённость соответствуют

требованиям санитарных норм и правил, установленных СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

Необходимые материалы и оборудование:

1. Персональные компьютеры
2. Браузеры с выходом в Интернет
3. Наборы, основанные на платформе Arduino. (различные Тройка-модули и шилды, микроконтроллеры Arduino Uno/Nano/Mega).

Кадровое обеспечение программы

К реализации программы привлекается педагог дополнительного образования (ПДО), отвечающий квалификационным требованиям, указанным в профессиональном стандарте педагога дополнительного образования детей и взрослых.

ПРОГРАММА МОДУЛЯ 1. ЗНАКОМСТВО С КВАНТОРИУМОМ
к дополнительной общеобразовательной общеразвивающей программе
технической направленности
«Junior в IT»

Уровень сложности: стартовый
Возраст обучающихся: 10-18 лет
Срок реализации: 1 мес.
Автор-составитель:
Зайцев Евгений Сергеевич,
педагог дополнительного образования

ВВЕДЕНИЕ

Программа модуля входит в структуру дополнительной общеобразовательной общеразвивающей программы «Junior в IT» технической направленности и является обязательным приложением к программе.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ МОДУЛЯ

<i>Предметные результаты</i>	<i>Метапредметные результаты</i>	<i>Личностные результаты</i>
1. Будут иметь предметное представление о направлениях деятельности ДТ 2. Будут изучены правила ТБ при нахождении в ДТ	1. Будут уметь договариваться 2. Будут уметь контролировать и оценивать свою работу	1. Будут уметь распознавать и формулировать свои интересы, включая профессиональные 2. Будут уметь работать в команде с разными возрастами участников

УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Название темы	Количество часов			Формы контроля
		Всего	Теория	Практика	
Модуль 1.	Знакомство с ДТ Кванториум	18	5	13	
1.1.	Комплектование групп	4	-	4	Тестирование
1.2.	Ознакомление с деятельностью ДТ Кванториум	10	4	6	Педагогическое наблюдение. Беседа. Анкетирование
1.3.	Тимбилдинг	2	-	2	Педагогическое наблюдение. Игра
1.4.	Введение	2	1	1	
	Всего часов	18	5	13	

СОДЕРЖАНИЕ МОДУЛЯ

№ п/п	Название темы и описание практической и теоретической части
1.1	Комплектование групп Практическая часть: Комплектование групп
1.2	Ознакомление с деятельностью ДТ Кванториум Теоретическая часть: Цели, задачи, направления деятельности ДТ Кванториум (информационные технологии, технологии виртуальной и дополненной реальности, промышленная робототехника, промышленный дизайн, хайтек-технологии, био-технологии). Правила внутреннего распорядка. Вводный инструктаж по ТБ Практическая часть: Экскурсия по детскому технопарку. Организация мастер-классов по направлениям для ознакомления с возможностями ДТ Кванториум
1.3	Тимбилдинг Практическая часть: Организация общекванторианской игры на сплочение, формирование коллектива кванторианцев.
1.4	Введение Теоретическая часть: Цели и задачи каждого модуля. Навыки и особенности освоения. Первичный инструктаж по ТБ Практическая часть: Определение уровня сформированности компетенций. Выбор модуля для дальнейшего обучения.

ПРОГРАММА МОДУЛЯ 2. WEB-разработка
к дополнительной общеобразовательной общеразвивающей программе
технической направленности
«Junior в IT»

Уровень сложности: стартовый
Возраст обучающихся: 10-18 лет
Срок реализации: 8 мес.
Автор-составитель:
Зайцев Евгений Сергеевич,
педагог дополнительного образования

ВВЕДЕНИЕ

Программа модуля входит в структуру дополнительной общеобразовательной общеразвивающей программы «Junior в IT» технической направленности и является обязательным приложением к программе.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ МОДУЛЯ

<i>Предметные результаты</i>	<i>Метапредметные результаты</i>	<i>Личностные результаты</i>
1) Навыки чтения кода, самостоятельно проектирования и исправления. 2) Программирование в различных средах. 3) Знания об устройстве всемирной глобальной сети, основных принципах ее функционирования. 4) Проявят навыки создания web-приложений, наполнения web-страниц информацией и контентом. 5) Освоят разработка программ различной сложности.	1) Будут проявлять собственный познавательный интерес; 2) Научатся подстраивать своё мышление под определенные задачи и проекты (дизайн мышления), отходить от плана или придерживаться инструкций; 3) Будут демонстрировать творческий подход к поставленной задаче; 4) Получат знания основ проектной и исследовательской деятельности.	1) Проявят навыки работы в команде. 2) Будут демонстрировать коммуникативность. 3) Ораторское искусство. 4) Проявят креативность. 5) Продемонстрируют умение самостоятельно находить решения при столкновении со сложными задачами. 6) Проявят стремление к получению новых знаний самостоятельно.

УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Название разделов, тем	Количество часов			Формы контроля
		Всего	Теория	Практика	
Модуль 1. (по выбору)	Web-разработка	126	24	102	
1	HTML+CSS	28	6	22	
1.1	Основы HTML	9	2	7	Практические задания.
1.2	Основы CSS	9	2	7	Практические задания.
1.3	Вёрстка сайта. HTML+CSS	10	2	8	Практические задания. Результат работы над проектом.
2	JavaScript	24	6	18	
2.1	JavaScript. Основы	12	3	9	Практические задания.
2.2	Java Web и Сайтов	12	3	9	Практические задания. Результат работы над проектом.
3	PHP	28	4	24	
3.1	PHP. Основы	14	2	12	Практические задания.
3.2	PHP для работы сайта	14	2	12	Практические задания. Результат работы над проектом.
4	WEB разработка	20	4	16	

4.1	PHP + JavaScript + HTML	10	2	8	Практические задания.
4.2	Настройка своего веб сервера Apache	10	2	8	Практические задания. Результат работы над проектом.
5	Web-разработка на других языках	24	4	20	
5.1	PyThon. Для Web.	8	2	6	Практические задания.
5.2	Работа с сайтами и Web- приложениями на PyThon.	7	1	6	Практические задания.
5.3	Кейс “Я – Web-разработчик”	9	1	8	Практические задания. Результат работы над проектом или кейсом “Я Web-разработчик”.
Итоговый контроль. Защита проектов или кейсов		2	0	2	Презентация кейса или проектных работ.

СОДЕРЖАНИЕ МОДУЛЯ

№ п/п	Название раздела, темы и описание практической и теоретической части	Формы контроля
Раздел 1. HTML+CSS		
1.1	Основы HTML Теоретическая часть: Основные теги и правила разметки Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
1.2	Основы CSS Теоретическая часть: основные инструкции и команды Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
1.3	Вёрстка сайта. HTML+CSS Теоретическая часть: правила работы над сайтом Практическая часть: разработка собственного сайта Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 2. JavaScript		
2.1	JavaScript. Основы Теоретическая часть: основы работы с JS и основные команды. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.

2.2	Java Web и Сайтов Теоретическая часть: основы разработки на JS для веб-приложений Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 3. PHP		
3.1	PHP. Основы Теоретическая часть: основы разработки на PHP. Основные команды Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
3.2	PHP для работы сайта Теоретическая часть: основы разработки на PHP для веб-сайтов Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 4. WEB разработка		
4.1	JavaScript. Основы Теоретическая часть: основы работы с JS и основные команды. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
4.2	JavaScript. Основы Теоретическая часть: основы работы с JS и основные команды. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 5. Web-разработка на других языках		
5.1.	PyThon. Для Web Теоретическая часть: основы работы с PyThon в Web и основные команды. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
5.2.	Работа с сайтами и Web- приложениями на PyThon. Теоретическая часть: основные API и их команды для работы с сайтами Практическая часть: отработка практических заданий и задач	Практические задания. Анализ. Диалог.

	Форма организации деятельности обучающихся: Групповая. Индивидуальная	
5.3	Кейс “Я – Web-разработчик” Теоретическая часть: вводная часть кейса и обсуждение решения Практическая часть: отработка задачи кейса Форма организации деятельности обучающихся: Групповая. Индивидуальная	Защита кейсовой работы.
Презентация кейса или проектных работ		
	Защита проектной работы или кейса	Презентация проектных работ

ПРОГРАММА МОДУЛЯ 3. ИНЖЕНЕРНЫЙ МОДУЛЬ
к дополнительной общеобразовательной общеразвивающей программе
технической направленности
«Junior в IT»

Уровень сложности: стартовый
Возраст обучающихся: 10-18 лет
Срок реализации: 8 мес.
Автор-составитель:
Зайцев Евгений Сергеевич,
педагог дополнительного образования

ВВЕДЕНИЕ

Программа модуля входит в структуру дополнительной общеобразовательной общеразвивающей программы «Junior в IT» технической направленности и является обязательным приложением к программе.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ МОДУЛЯ

<i>Предметные результаты</i>	<i>Метапредметные результаты</i>	<i>Личностные результаты</i>
1) Знания об основных принципах программирования микроконтроллеров. 2) Навыки написания среднего уровня сложности программ. 3) Навыки чтения кода, самостоятельно проектирования и исправления. 4) Смогут программировать в различных средах. 5) Покажут навыки создания сетевых взаимодействий между устройствами 6) Освоят разработку программ различной сложности.	1) Будут проявлять собственный познавательный интерес; 2) Научатся подстраивать своё мышление под определенные задачи и проекты (дизайн мышления), отходить от плана или придерживаться инструкций; 3) Будут демонстрировать творческий подход к поставленной задаче; 4) Получат знания основ проектной и исследовательской деятельности.	1) Проявят навыки работы в команде. 2) Будут демонстрировать коммуникативность. 3) Ораторское искусство. 4) Проявят креативность. 5) Продемонстрируют умение самостоятельно находить решения при столкновении со сложными задачами. 6) Проявят стремление к получению новых знаний самостоятельно.

УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Название разделов, тем	Количество часов			Формы контроля
		Всего	Теория	Практика	
Модуль 2. (по выбору)	Инженерный модуль	126	30	96	
1	Основы схемотехники	30	6	24	
1.1	Физика проводников	10	2	8	Практические задания. Результат работы над собственным устройством.
1.2	Элементы схем	10	2	8	Практические задания. Результат работы над собственным устройством
1.3	Датчики	10	2	8	Практические задания. Результат работы над собственным устройством
2	Arduino	22	6	16	
2.1	ArduinoIDE. Основы	11	3	8	Практические задания.
2.2	C++ и Arduino. Основы	11	3	8	Практические задания. Результат работы над устройством
3	C++ для микроконтроллеров	30	8	22	
3.1	C++. Логика	16	5	11	Практические задания

3.2	Логика, как направление в математике	14	3	11	Практические задания
4	Сложные конструкции программ. Взаимодействие датчиков	12	4	8	
4.1	Робот “прилипала”	12	4	8	Результат выполнения практического задания по работе над роботом
5	Системы безопасности и мониторинга	30	6	24	
5.1	Основы систем безопасности и псевдо AI	10	2	8	Практические задания
5.2	Бегающий анализатор	10	2	8	Результат выполнения практического задания по работе над роботом
5.3	Кейс “Я – Инженер”	10	2	8	Результат выполнения кейса
Итоговый контроль. Защита проектов или кейсов		2	0	2	Презентация кейса или проектных работ

СОДЕРЖАНИЕ МОДУЛЯ

№ п/п	Название раздела, темы и описание практической и теоретической части	Формы контроля
Раздел 1. Основы схемотехники		
1.1	Физика проводников Теоретическая часть: Основные законы электрических цепей и физические явления Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
1.2	Элементы схем Теоретическая часть: Основные элементы схем в схемотехнике. Проводимость, полупроводимость. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
1.3	Датчики Теоретическая часть: принцип работы датчиков разных типов и свойств Практическая часть: выработка практических занятий по сборке определенного типа устройств Форма организации деятельности обучающихся:	Практические задания. Анализ. Диалог.

	Групповая. Индивидуальная	
Раздел 2. Arduino.		
2.1	ArduinoIDE. Основы Теоретическая часть: основы работы в Arduino IDE и основные команды. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
2.2	С++ и Arduino. Основы Теоретическая часть: использование языка программирования С++ в Arduino IDE Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 3. С++ для микроконтроллеров		
3.1	С++. Логика Теоретическая часть: основы работы логических выражений (если, то, или, не) Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
3.2.	Логика, как направление в математике Теоретическая часть: Логические операторы в математике. Таблица истинности. Логические операции и операнды. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 4. Сложные конструкции программ. Взаимодействие датчиков		
4.1	Робот “прилипала” Теоретическая часть: основы робототехники на примере робота, который движется за хозяином. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 5. Системы безопасности и мониторинга		
5.1	Основы систем безопасности и псевдо AI Теоретическая часть: основы работы систем безопасности, основы работы искусственного интеллекта, AI для систем безопасности Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.

5.2	Бегающий анализатор Теоретическая часть: использование других датчиков в роботах Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
5.3	Кейс “Я – Инженер” Теоретическая часть: вводная часть кейса и обсуждение решения Практическая часть: отработка задачи кейса Форма организации деятельности обучающихся: Групповая. Индивидуальная	Защита кейсовой работы.
Презентация кейса или проектных работ		
	Защита проектной работы или кейса	Презентация проектной работы

ПРОГРАММА МОДУЛЯ 4. GAME-DEV
к дополнительной общеобразовательной общеразвивающей программе
технической направленности
«Junior в IT»

Уровень сложности: стартовый
Возраст обучающихся: 10-18 лет
Срок реализации: 8 мес.
Автор-составитель:
Зайцев Евгений Сергеевич,
педагог дополнительного образования

ВВЕДЕНИЕ

Программа модуля входит в структуру дополнительной общеобразовательной общеразвивающей программы «Junior в IT» технической направленности и является обязательным приложением к программе.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ МОДУЛЯ

<i>Предметные результаты</i>	<i>Метапредметные результаты</i>	<i>Личностные результаты</i>
1) Знание основных этапов разработки видеоигр. 2) Знание основных принципов моделирования персонажа. 3) Умение настраивать физику в игровых движках. 4) Способность разобраться в работе AI в играх. 5) Умение разрабатывать игры на различных движках и языках программирования 6) Навыки FullStack-разработки.	1) Будут проявлять собственный познавательный интерес; 2) Научатся подстраивать своё мышление под определенные задачи и проекты (дизайн мышления), отходить от плана или придерживаться инструкций; 3) Будут демонстрировать творческий подход к поставленной задаче; 4) Получат знания основ проектной и исследовательской деятельности.	1) Проявят навыки работы в команде. 2) Будут демонстрировать коммуникативность. 3) Проявят ораторское искусство. 4) Проявят креативность. 5) Продемонстрируют умение самостоятельно находить решения при столкновении со сложными задачами. 6) Проявят стремление к получению новых знаний самостоятельно.

УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Название разделов, тем	Количество часов			Формы контроля
		Всего	Теория	Практика	
Модуль 3 (по выбору)	Game-Dev модуль	126	24	102	
1	Основы создания игр	28	6	22	
1.1	Основы Unreal Engine	9	2	7	Практические задания. Результат работы над прототипом игры.
1.2	Основы GoDot	9	2	7	Практические задания. Результат работы над прототипом игры.
1.3	Основы Unity	10	2	8	Практические задания. Результат работы над прототипом игры.
2	3D моделирование для игр	24	6	18	
2.1	Blender. Основы	12	3	9	Практические задания. Результат работы над моделью для персонажа для одного из движков.
2.2	Maya. Основы	12	3	9	Практические задания. Результат работы над моделью для персонажа для одного из движков.

3	С# для разработки	28	4	24	
3.1	С#. Основы	14	2	12	Практические задания
3.2	С# в Unity	14	2	12	Практические задания. Результат работы над прототипом игры.
4	Game-Dev без движка	20	4	16	
4.1	Визуализация кодом С#	10	2	8	Практические задания
4.2	Игровое программирование	10	2	8	Практические задания. Результат визуализации кода.
5	Game-Dev от и до	24	4	20	
5.1	Unreal Engine. NPC+AI+Взаимодействие	8	2	6	Практические задания. Результат работы над прототипом игры.
5.2	Unity. NPC+AI+Взаимодействие	7	1	6	Практические задания. Результат работы над прототипом игры.
5.3	Кейс “Я –Game-Developer”	9	1	8	Результат работы над кейсом.
Итоговый контроль. Защита проектов или кейсов		2	0	2	Презентация кейса или проектных работ.

СОДЕРЖАНИЕ МОДУЛЯ

№ п/п	Название раздела, темы и описание практической и теоретической части	Формы контроля
Раздел 1. Основы создания игр		
1.1	Основы Unreal Engine Теоретическая часть: основы разработки в UE4. Blueprints Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
1.2	Основы GoDot Теоретическая часть: основы разработки в GoDot. GDScript Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
1.3	Основы Unity Теоретическая часть: основы разработки в Unity. С# скрипты. Практическая часть: практические задачи Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 2. 3D моделирование для игр.		
2.1	Blender. Основы Теоретическая часть: основы работы в Blender. Разбитие на примитивы. Практическая часть: отработка практических заданий и задач	Практические задания. Анализ. Диалог.

	Форма организации деятельности обучающихся: Групповая. Индивидуальная	
2.2.	Maya. Основы Теоретическая часть: основы работы в Autodesk Maya. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 3. C# для разработки.		
3.1	C#. Основы Теоретическая часть: основы программирования на языке C# Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
3.2	C# в Unity Теоретическая часть: использование языка C# для написания скриптов в Unity. Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 4. Game-Dev без движка.		
4.1	Визуализация кодом C# Теоретическая часть: команды для визуализации кода в C# и написания интерфейса игры без движка Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
4.2	Игровое программирование Теоретическая часть: написание визуальной новеллы без использования движка Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
Раздел 5. Game-Dev от и до.		
5.1	Unreal Engine. NPC+AI+Взаимодействие Теоретическая часть: детальный разбор создания игр с неигровыми персонажами и их искусственным интеллектом на UE4 Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся: Групповая. Индивидуальная	Практические задания. Анализ. Диалог.
5.2	Unity. NPC+AI+Взаимодействие Теоретическая часть: детальный разбор создания игр с неигровыми персонажами и их искусственным интеллектом на Unity Практическая часть: отработка практических заданий и задач Форма организации деятельности обучающихся:	Практические задания. Анализ. Диалог.

	Групповая. Индивидуальная	
5.3	Кейс “Я –Game-Developer” Теоретическая часть: вводная часть кейса и обсуждение решения Практическая часть: отработка задачи кейса Форма организации деятельности обучающихся: Групповая. Индивидуальная	Защита кейсовой работы.
Презентация кейса или проектных работ		
	Защита проектной работы или кейса	Презентация проектной работы

Информационная карта оценки освоения обучающимися образовательной программы

Квантум:

Группа:

Педагог:

№	ФИО	Сумма баллов по компетенциям			Сумма баллов по Проекту	Итого сумма баллов	Показатель освоения программы
		личностные	метапредметные	предметные (по результатам сдачи кейсов или проектов)			

Показатели освоения программы обучающимися по баллам:

Высокий уровень освоения – 50-58

Средний уровень освоения – 35-50

Низкий уровень освоения – 20-35

Шкала критериев для оценивания развития компетенций:

ФИО обучающегося:

Квантум:

Параметр оценивания	Характеристика низкого уровня	Оценка уровня			Характеристика высокого уровня
		0 плохо	2 средне	3 отлично	
ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ:					
Работа в команде, Коммуникация	Практически не участвует в формировании команды для работы над проектом, плохо реагирует на обратную связь и практически не учитывает ее в работе, не умеет делегировать задачи				Активно участвует в формировании команды для работы над проектом, распределении функций, и ролей в проекте, адекватно реагирует на обратную связь и учитывает ее в работе
Ораторское искусство	Не может выражать четко свои мысли. Говорит не уверенно, скомкано, сжевано. Речь тихая, эмоционально слабая.				Говорит четко, уверенно, с сильной подачей. Речь внятная, разборчивая, понятная.
Креативность	Демонстрирует привязанность к стереотипам, стремиться подражать другим, чтобы не выделяться и не отличаться, не высказывает собственные оригинальные идеи. не				Активно демонстрирует способность нешаблонно мыслить, генерировать идеи проявляет воображение, мыслит образно, демонстрирует способность выявлять отличительные особенности, выявлять

	демонстрирует умение видеть необычное в обыденном				потенциал, который ранее был не замечен
Умение самостоятельно находить решения при столкновении со сложными задачами	Ждет подсказок от педагога или других учеников. Работает по плану, который для него разработал кто-то другой.				Самостоятельно находит решение, подбирает материал, который нужно изучить для решения конкретной задачи.
Стремление к получению новых знаний самостоятельно	Занимается только в течение занятий, не проявляет интереса к материалу, не предложенному педагогом.				Занимается дополнительно самостоятельно, изучает материал, который сам найдет.
ИТОГО БАЛЛОВ:					
МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ:					
Собственный познавательный интерес	Разбирается только в том, что разбирает наставник				Помимо информации, данной наставником, интересуется дополнительным материалом, задает дополнительные вопросы, не рассмотренные педагогом.
Дизайн мышления	Демонстрирует привязанность плану выполнения работы, отклонение от плана вызывает отрицательные эффекты на ходе выполнения работы.				Активно демонстрирует способность отходить от плана для повышения эффективности работы, готов самостоятельно переделать план, если сомневается в его эффективности.
Творческий подход к поставленной задаче	Демонстрирует привязанность к стереотипам работы над задачей.				Активно демонстрирует творческое мышление, предлагает идеи решения задач, отходит от стереотипов.
Основы проектной деятельности.	При поставке задачи не демонстрирует никаких шагов к решению или придерживается кого-то из группы.				При постановке задачи поэтапно начинает работу над её решением, проходя все необходимые этапы работы над проектом.
ИТОГО БАЛЛОВ:					
ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ:					
Знания об устройстве компьютеров и компьютерных сетей, их истории и современных тенденциях развития.	Теряется при вопросах об устройстве ПК или сетевого оборудования. Не может разобрать и собрать ПК или сервер из предоставленных необходимых компонентов.				Спокойно ориентируется в сетевом оборудовании и деталях ПК. Может разобрать и собрать ПК или сервер из предоставленных необходимых компонентов.
Знания об основных принципах программирования и навыки написания среднего уровня сложности программ.	Все полученные программы просты в реализации, не используются сложные конструкции, прямолнейны и не оптимизированы.				Свободно владеет любыми конструкциями кода, программы оптимизированы, одинаково работают на различных современных устройствах.
Навыки чтения кода, самостоятельно проектирования и исправления.	Теряется в чужом коде, написанном по стандарту PEP-8, при анализе кода не понимает, что за чем следует, не может найти ошибку				Спокойно и свободно ориентируется в коде, написанном по стандарту PEP-8, может найти в нем ошибку и исправить её

Программирование в различных средах.	Совсем не дается создание веб приложения или предмета IoT, или сетевой игры.				Сделал кейс или проект с взаимодействием через интернет.
Знания об устройстве всемирной глобальной сети, основных принципах ее функционирования.	Тяжело дается работы в сети интернет, не понимание принципов передачи сигналов и информации посредством сети. Совсем не дается создание веб приложения или предмета IoT, или сетевой игры.				Легко и свободно добывает информацию в сети, умеет её анализировать и правильно подстраивает запросы в сети, дабы получить нужную информацию. Сделал кейс или проект с взаимодействием через интернет.
ИТОГО БАЛЛОВ:					

Оценка входного теста происходит по установленным критериям. За каждый вопрос устанавливается 1 балл

26-31 баллов – высокий уровень

20-25 баллов – средний уровень

12-19 баллов – низкий уровень

Пример теста промежуточного контроля

- Какой из перечисленных типов данных не является типом данных в C++?
 - float
 - real
 - double
 - int
 - Какие служебные символы используются для обозначения начала и конца блока кода?
 - { }
 - ()
 - begin end
 - < >
 - Укажите правильную форму записи цикла do while
-
- Какими знаками заканчивается большинство строк кода в Си++?
 - (точка)
 - (запятая)
 - (двоеточие)
 - (точка с запятой)
 - Язык программирования C++ разработал
 - Дональд Кнут
 - Кен Томпсон
 - Бьерн Страуструп
 - Никлаус Вирт
 - Цикл с предусловием?
 - For
 - While
 - do while
 - Чтобы подключить заголовочный файл в программу на C++, например iostream необходимо написать:
 - #include <> с iostream внутри скобок
 - #include <> ; с iostream.h внутри скобок

3. `include (iostream)`
4. `include #iostream,h;`
8. Какому зарезервированному слову программа передает управление в случае, если значение переменной или выражения оператора `switch` не совпадает ни с одним константным выражением?
 1. Default
 2. All
 3. Contingency
 4. other
9. Цикл с постусловием?
 1. For
 2. Do while
 3. while
10. Какая из следующих записей – правильный комментарий в C++?
 1. `/*комментарий*/`
 2. `{комментарий}`
 3. `**Комментарий**`
 4. `*/Комментарий/*`
11. Укажите объектно-ориентированный язык программирования
 1. Все варианты ответов
 2. Java о C++
 3. Eiffel 13
12. До каких пор будут выполняться операторы в теле цикла `while (x < 100)`?
 1. Пока `x` строго меньше ста
 2. Пока `x` больше ста
 3. Пока `x` равен сотне
 4. Пока `x` меньше или равен сотне
13. Чему равна переменная `a`, после выполнения этого кода:

```
int a;  
for(a = 0; a < 10; a++) {}?
```

 1. 9
 2. 10
 3. 1
14. Какую функцию должны содержать все программы на C++?
 1. `program()`
 2. `system()`
 3. `start()`
 4. `main()`
15. Укажите правильное определение функции `main` в соответствии со спецификацией стандарта ANSI
 1. `int main(void)`
 2. `void main()`
 3. `void main(void)`
 4. `int main()`
16. Какой из следующих операторов – оператор сравнения двух переменных?
 1. `equal`
 2. `=`
 3. `==`
 4. `:=`
17. Общий формат оператора множественного выбора – `switch`
 1.

```
switch (switch_expression) {  
    case constant1, case constant2, statement1;
```

- ```

 brake;
 case constantN: statementN;
 brake;
 default:statement N+1;}
2. switch (switch_expression) {
 case constant1: statement1;
 brake;
 case constant2: statement2;
 brake;
 case constantN: statementN;
 brake;
 else: statement N+1;}
3. switch (switch_expression) {
 case constant1: statement1;
 brake;
 case constant2: statement2;
 brake;
 case constantN: statementN;
 brake;
 default: statement N+1;}

```
18. Название C# предложил
    1. Рик Масситти
    2. Кэн Томпсон
    3. Дональд Кнут
    4. Бьер Страуструп
  19. Выберите правильный вариант объявления константной переменной в C++, где type – тип данных в C++ variable – имя переменной value – константное значение
    1. const type variable := value;
    2. const variable = value;
    3. const type variable = value;
  20. Какой из нижеперечисленных операторов, не является циклом в C++?
    1. do while
    2. while
    3. repeat until
    4. for
  21. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется:
    1. Интерпретатор
    2. сканер
    3. транслятор
    4. компилятор
  22. Какое значение, по умолчанию, возвращает программа операционной системе в случае успешного завершения?
    1. 1
    2. Программа не возвращает значения.
    3. 1
    4. 0
  23. Простые принципы данных в C++.
    1. Целые - int, вещественные - float или double, символьные – char
    2. Целые - int, вещественные - float или real, символьные - char
    3. Целые - bool, вещественные - float или double, символьные - string
    4. Целые - int, вещественные - float или double, символьные - char

24. Какой оператор не допускает перехода от одного константного выражения к другому?
1. break;
  2. точка с запятой;
  3. Stop;
  4. end;
25. Какой служебный знак ставится после оператора case?
1. :
  2. .
  3. -
  4. ;
26. Каков результат работы следующего фрагмента кода? `int x = 0; switch(x) {case 1: cout << "Один"; case 0: cout << "Нуль"; case 2: cout << "Привет мир";}`
1. Нуль
  2. Привет мир
  3. Один
  4. НульПривет мир
27. Что будет напечатано? `int main() { for(int i = 0; i < 4; ++i){switch(i) {case 0 : std::cout << "0"; case 1 : std::cout << "1"; continue; case 2 : std::cout << "2"; break; default : std::cout << "D"; break; } std::cout << ".";} return 0;}`
1. 0.1.2
  2. 011.2.D.
  3. Ошибка компиляции в строке 10
  4. 0112.D.
  5. 01.2.D
28. Тело любого цикла выполняется до тех пор, пока его условие ...
1. ложно
  2. у цикла нет условия
  3. истинно
29. Структура объявления переменных в C++
- [=]
  - [:=]
  - [=];
  - [==];
30. В приведенном коде измените или добавьте один символ чтобы код напечатал 20 звездочек - \*
1. `int i, N = 20; for(i=0, i < N; i--) printf("*");`
  2. `int i, N = 20; for(i=0; i < N; i--) printf("*");`
  3. `int i, N=20; for(i = 20; i < N; i--) printf("*");`
  4. `int i, N=20; for(i = 19; i < N; i--) printf("*");`
  5. `int i, N=40; for(i = 0; i < N; i--) printf("*");`
31. Какие среды программирования (IDE) предназначены для разработки программных средств?
1. MVS, NetBeans, QT Creator, RAD Studio, Dev-C++
  2. MVS, Code::Blocks, QT Creator, RAD Studio, MathCAD
  3. MVS, Code::Blocks, QT Creator, AutoCAD, Eclipse
-

**Оценка защиты проектов или кейсов** происходит по установленным критериям.

За каждый критерий устанавливается от 1 до 4 баллов.

**от 12 баллов** - высокий уровень

**5-11 баллов** - средний уровень

**4 балла** - низкий уровень

**Критерии оценивания проектов (кейсов):**

**1. Проект**

- 1.1. Оригинальность и качества решения
- 1.2. Исследование и отчет
- 1.3. Зрелищность

**2. Программирование и инженерное решение**

- 2.1. Сложность
- 2.2. Понимание технической части
- 2.3. Инженерные решения
- 2.4. Эстетичность

**3. Презентация**

- 3.1. Навыки общения и аргументация
- 3.2. Скорость мышления

**4. Командная работа**

- 4.1. Уровень понимания проекта
- 4.2. Сплоченность коллектива
- 4.3. Командный дух

### Кейс “Я – Web-разработчик”

#### О кейсе:

Каждый кванторианец однажды напишет проект, который будет представлен на каком-либо конкурсе. О своём проекте он мог бы рассказать другим кванторианцам. А ещё ему интересно будет узнать, какие интересные места есть в городе, чтобы почитать о них и посетить. Интересно было бы почитать и о последних новостях Кванториума.

**Категория кейса:** базовый

**Примерный возраст обучающихся:** 10-18 лет

**Место в структуре программы:** после занятий об основах языка Python для Web-разработки

**Количество академических часов, на которые рассчитан кейс:** от 9 до 16.

#### Учебно-тематическое планирование:

|                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                           |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Блок 1. Постановка проблемы и поиск возможных путей ее решения</b>                                                                                                                                                                                      |                                                                                                                                                                                                                           |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                         |
| до 2 часов                                                                                                                                                                                                                                                 | Произвести постановку проблемной ситуации и осуществить поиск возможных путей ее решения                                                                                                                                  |
| 1. Представление проблемной ситуации в виде ограничения.<br>2. Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения идеального конечного результата.<br>3. Выбор конкретного метода решения из всех предложенных |                                                                                                                                                                                                                           |
| <b>Блок 2. Проектирование решения. Распределение по командам, определение ролей</b>                                                                                                                                                                        |                                                                                                                                                                                                                           |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                         |
| до 4 часов                                                                                                                                                                                                                                                 | Спроектировать решение, определившись с архитектурой, средствами реализации, языком и средами программирования. Разбившись на команды, определить роль каждого участника (программист, контент-менеджер, дизайнер и т.д.) |
| 1. Изучение необходимых технологий.<br>2. Проектирование решения.<br>3. Построение алгоритма решения.<br>4. Разбивка на команды по 3-4 человека<br>5. Распределение ролей в команде.                                                                       |                                                                                                                                                                                                                           |
| <b>Блок 3. Создание рабочего прототипа</b>                                                                                                                                                                                                                 |                                                                                                                                                                                                                           |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                         |



|                                                                                                                       |                                                                                      |
|-----------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| до 8 часов                                                                                                            | Создать и запрограммировать бота                                                     |
| 1. Создание бота в Telegram.<br>2. Написание функционала бота, используя язык Python<br>3. Наполнение бота контентом. |                                                                                      |
| <b>Блок 4. Тестирование и исправление ошибок</b>                                                                      |                                                                                      |
| <b>Предполагаемая продолжительность</b>                                                                               | <b>Цель блока</b>                                                                    |
| до 2 часов                                                                                                            | Протестировать решение, если выявятся ошибки в процессе работы бота, то исправить их |
| 1. Тестирование готового прототипа<br>2. Внесение исправлений при выявлении ошибок                                    |                                                                                      |

#### Предполагаемые результаты обучающихся

**Артефакты:** Telegram-бот, имеющий информацию об интересных местах города, новостях и проектах Кванториума в том городе, где обучается ребёнок.

##### Soft skills:

1. Умение генерировать идеи указанными методами.
2. Умение слушать и слышать собеседника.
3. Умение аргументировано обосновывать свою точку зрения.
4. Умение искать информацию в различных источниках и структурировать ее.
5. Умение работать в команде.
6. Умение грамотно письменно формулировать свои мысли.
7. Критическое мышление и умение объективно оценивать свои результаты.

##### Hard skills:

1. Умение составлять алгоритм работы программы в виде блок-схемы.
2. Умение работать с базой данных.
3. Умение программировать на языке Python, использовать различные подключаемые модули и библиотеки, умение взаимодействовать с API Telegram.
4. Умение готовить и красиво оформлять контент, который наполняет бота.

#### Дополнительно (вариативная часть)

Руководство наставника

##### Текст-легенда кейса:

В настоящее время в нашем кванториуме нет инструмента, который использовали бы кванторианцы для того, чтобы делиться друг с другом своими достижениями в виде выполненных проектов, интересными местами, которые можно посетить в нашем городе, узнавать последние новости Кванториума. Ребятам предлагается создать такой инструмент, который был бы удобен в первую очередь им самим. Учитывая то, что сейчас очень модно пользоваться различными мессенджерами, было бы неплохо реализовать такой инструмент именно через мессенджер, к примеру, Telegram или

VK или может быть использовать традиционное решение в виде создания сайта. С конкретным решением обучающиеся определяются сами в ходе генерации идей.

**Материалы в помощь:**

Дополнительные ссылки, инструкции, иллюстрации и прочие материалы, которые помогут наставнику сопровождать работу детей на всех этапах реализации кейса.

**Обратить внимание:**

Представить примерный путь обучающихся по жизненному циклу и обозначить возможные узкие места: на что обратить внимание, какими вопросами можно помочь обучающимся и прочее

**Руководство для обучающегося**

Текст-легенда кейса

Жизненная преламбула, описание ситуации или иное описание, погружающее обучающегося в проблематику кейса.

Жизненный цикл:

Проблематизация -> Целеполагание -> Поиск решения -> Планирование -> Реализация замысла -> Финализация

Представить примерный путь обучающихся по жизненному циклу и дать указания и советы по прохождению: что посмотреть, почитать дополнительно, какие вопросы надо задать самим себе на каждом из этапов жизненного цикла.

### Кейс “Я - Инженер”

#### О кейсе:

Интернет вещей, что это такое? Кто это придумал? Как это работает? Самый яркий пример – телефон. Сейчас благодаря интернету, мы не только можем кому-то отправить письмо, мы можем увидеть человека, даже если он находится в диаметрально противоположной точке планеты. И понять, как это всё функционирует внутри, и сделать хотя бы установку для обмена звонками, позволит каждому ребенку понять функционирование сотовых сетей и интернета.

**Категория кейса:** базовый

**Примерный возраст обучающихся:** 10-18 лет

**Место в структуре программы:** после занятий об основах Интернета вещей и Ардуино

**Количество академических часов, на которые рассчитан кейс:** от 10 до 18.

#### Учебно-тематическое планирование:

|                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Блок 1. Постановка проблемы и поиск возможных путей ее решения</b>                                                                                                                                                                                      |                                                                                                                                                                                                                  |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                |
| 2 часа                                                                                                                                                                                                                                                     | Произвести постановку проблемной ситуации и осуществить поиск возможных путей ее решения                                                                                                                         |
| 1. Представление проблемной ситуации в виде ограничения.<br>2. Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения идеального конечного результата.<br>3. Выбор конкретного метода решения из всех предложенных |                                                                                                                                                                                                                  |
| <b>Блок 2. Проектирование решения. Распределение по командам, определение ролей</b>                                                                                                                                                                        |                                                                                                                                                                                                                  |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                |
| 2 часа                                                                                                                                                                                                                                                     | Спроектировать решение, определившись с архитектурой, средствами реализации, языком и средами программирования. Разбившись на команды, определить роль каждого участника (программист, инженер, дизайнер и т.д.) |
| 1. Изучение необходимых технологий.<br>2. Проектирование решения.<br>3. Построение алгоритма решения.<br>4. Разбивка на команды по 3-4 человека<br>5. Распределение ролей в команде.                                                                       |                                                                                                                                                                                                                  |
| <b>Блок 3. Создание рабочего прототипа</b>                                                                                                                                                                                                                 |                                                                                                                                                                                                                  |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                |

|                                                                                                                   |                                                                                      |
|-------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| 4 часа                                                                                                            | Создать и запрограммировать бота                                                     |
| 1. Создание прототипа мобильного.<br>2. Написание основного функционала установки<br>3. Наполнение доп.функциями. |                                                                                      |
| <b>Блок 4. Тестирование и исправление ошибок</b>                                                                  |                                                                                      |
| <b>Предполагаемая продолжительность</b>                                                                           | <b>Цель блока</b>                                                                    |
| 2 часа                                                                                                            | Протестировать решение, если выявятся ошибки в процессе работы бота, то исправить их |
| 1. Тестирование готового прототипа<br>2. Внесение исправлений при выявлении ошибок                                |                                                                                      |

### Предполагаемые результаты обучающихся

**Артефакты:** прототип мобильного телефона на базе Arduino.

#### Soft skills:

1. Умение генерировать идеи указанными методами.
2. Умение слушать и слышать собеседника.
3. Умение аргументированно обосновывать свою точку зрения.
4. Умение искать информацию в различных источниках и структурировать ее.
5. Умение работать в команде.
6. Умение грамотно письменно формулировать свои мысли.
7. Критическое мышление и умение объективно оценивать свои результаты.

#### Hard skills:

1. Умение составлять алгоритм работы программы в виде блок-схемы.
2. Умение работать с микросхемами и одноплатами контроллерами
3. Умение программировать на языке C++, использовать различные подключаемые модули и библиотеки.

### Дополнительно (вариативная часть)

Руководство наставника

#### Текст-легенда кейса:

Ребят, все мы пользуемся мобильными телефонами? Сколько стоит примерно самый дешевый в нашем кабинете? А хотите сделать звонящий и рабочий мобильный телефон, но в 10 раз дешевле? Заодно с вами как раз таки погрузимся в IoT и станем уже настоящими разработчиками.

#### Материалы в помощь:

Дополнительные ссылки, инструкции, иллюстрации и прочие материалы, которые помогут наставнику сопроводить работу детей на всех этапах реализации кейса.

#### Обратить внимание:

Представить примерный путь обучающихся по жизненному циклу и обозначить возможные узкие места: на что обратить внимание, какими вопросами можно помочь обучающимся и прочее

**Руководство для обучающегося**

Текст-легенда кейса

Жизненная преамбула, описание ситуации или иное описание, погружающее обучающегося в проблематику кейса.

Жизненный цикл:

Проблематизация -> Целеполагание -> Поиск решения -> Планирование -> Реализация замысла -> Финализация

Представить примерный путь обучающихся по жизненному циклу и дать указания и советы по прохождению: что посмотреть, почитать дополнительно, какие вопросы надо задать самим себе на каждом из этапов жизненного цикла.

### Кейс “Я Game-Developer”

#### О кейсе:

Все мы любим играть. Я тоже, не скрою. Но новые игры либо не интересны, либо дорого стоят. Да и наверное, не считая сложного программирования для взлома, программирование и написание игр стоит на первом месте. Все больше и больше людей приобщается к игровой индустрии. Поэтому нужно и нам не отставать и знать тут все.

**Категория кейса:** базовый

**Примерный возраст обучающихся:** 10-18 лет

**Место в структуре программы:** после основ разработки в Unreal Engine v.4

**Количество академических часов, на которые рассчитан кейс:** от 9 до 18.

#### Учебно-тематическое планирование:

|                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                   |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Блок 1. Постановка проблемы и поиск возможных путей ее решения</b>                                                                                                                                                                                      |                                                                                                                                                                                                                   |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                 |
| 2 часа                                                                                                                                                                                                                                                     | Произвести постановку проблемной ситуации и осуществить поиск возможных путей ее решения                                                                                                                          |
| 1. Представление проблемной ситуации в виде ограничения.<br>2. Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения идеального конечного результата.<br>3. Выбор конкретного метода решения из всех предложенных |                                                                                                                                                                                                                   |
| <b>Блок 2. Проектирование решения. Распределение по командам, определение ролей</b>                                                                                                                                                                        |                                                                                                                                                                                                                   |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                 |
| 2 часа                                                                                                                                                                                                                                                     | Спроектировать решение, определившись с архитектурой, средствами реализации, языком и средами программирования. Разбившись на команды, определить роль каждого участника (программист, дизайнер, режиссер и т.д.) |
| 1. Изучение необходимых технологий.<br>2. Проектирование решения.<br>3. Построение алгоритма решения.<br>4. Разбивка на команды по 3-4 человека.<br>5. Распределение ролей в команде.                                                                      |                                                                                                                                                                                                                   |
| <b>Блок 3. Создание рабочего прототипа</b>                                                                                                                                                                                                                 |                                                                                                                                                                                                                   |
| <b>Предполагаемая продолжительность</b>                                                                                                                                                                                                                    | <b>Цель блока</b>                                                                                                                                                                                                 |

|                                                                                                             |                                                                                      |
|-------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| 4 часа                                                                                                      | Создать и запрограммировать бота                                                     |
| 1. Создание первых уровней<br>2. Написание основного функционала мира и персонажа<br>3. Наполнение графикой |                                                                                      |
| <b>Блок 4. Тестирование и исправление ошибок</b>                                                            |                                                                                      |
| <b>Предполагаемая продолжительность</b>                                                                     | <b>Цель блока</b>                                                                    |
| 1 час и более                                                                                               | Протестировать решение, если выявятся ошибки в процессе работы бота, то исправить их |
| 1. Тестирование готовых уровней<br>2. Внесение исправлений при выявлении ошибок                             |                                                                                      |

### Предполагаемые результаты обучающихся

**Артефакты:** своя собственная игра на мобильное устройство или ПК

#### Soft skills:

1. Умение генерировать идеи указанными методами.
2. Умение слушать и слышать собеседника.
3. Умение аргументировано обосновывать свою точку зрения.
4. Умение искать информацию в различных источниках и структурировать ее.
5. Умение работать в команде.
6. Умение грамотно письменно формулировать свои мысли.
7. Критическое мышление и умение объективно оценивать свои результаты.

#### Hard skills:

1. Умение составлять алгоритм работы программы в виде блок-схемы.
2. Умение работать инструментами для создания игр
3. Умение программировать на языке C++, использовать различные подключаемые модули и библиотеки.

### Дополнительно (вариативная часть)

Руководство наставника

#### Текст-легенда кейса:

Большинство вас пришло сюда именно для этого. Все этого жаждали и хотели. Все меня постоянно об этом спрашивали. Ну что ж, свершилось, нам пора. Мы приступаем к созданию игр. Пора показать, что вы любите и что вы хотите. Вперед!

#### Материалы в помощь:

Дополнительные ссылки, инструкции, иллюстрации и прочие материалы, которые помогут наставнику сопроводить работу детей на всех этапах реализации кейса.

#### Обратить внимание:

Представить примерный путь обучающихся по жизненному циклу и обозначить возможные узкие места: на что обратить внимание, какими вопросами можно помочь обучающимся и прочее

**Руководство для обучающегося**

Текст-легенда кейса

Жизненная преамбула, описание ситуации или иное описание, погружающее обучающегося в проблематику кейса.

Жизненный цикл:

Проблематизация -> Целеполагание -> Поиск решения -> Планирование -> Реализация замысла -> Финализация

Представить примерный путь обучающихся по жизненному циклу и дать указания и советы по прохождению: что посмотреть, почитать дополнительно, какие вопросы надо задать самим себе на каждом из этапов жизненного цикла.



### Общие ключевые дела ДТ Кванториум

| №  | Ключевые дела                                | Срок реализации | Ответственный                                | Примечание |
|----|----------------------------------------------|-----------------|----------------------------------------------|------------|
| 1. | Посвящение в Кваторианцы.<br>Тимбилдинг      | сентябрь        | Педагог-организатор                          |            |
| 2. | День рождения Кванториума                    | декабрь         | Педагог-организатор                          |            |
| 3. | Хакатоны (по направлениям)                   | в течение года  | Педагог-организатор.<br>Педагоги направлений |            |
| 4. | Тематические лекции от приглашенных спикеров | в течение года  | Педагог-организатор                          |            |
| 5. | Тематические недели                          | в течение года  | Педагог-организатор                          |            |
| 6. | Экскурсии на предприятия                     | в течение года  | Педагог-организатор                          |            |
| 7. | Научный StandUp                              | май             | Педагог-организатор.<br>Педагоги направлений |            |