

Автономная некоммерческая организация
«Красноярский детский технопарк «Кванториум»

РЕКОМЕНДОВАНО
методическим советом

Протокол № 9
от «30» мая 2022 г.

УТВЕРЖДАЮ

Генеральный директор
Кениг С.Р.



Приказ № 34
от 30 мая 2022 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Основы мобильной разработки»

Срок реализации:

1 год

Возраст детей:

14-18 лет

Составитель программы:

Нагуслаев Н.Т.

г. Красноярск, 2022 г.

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Основы мобильной разработки» (далее – программа), имеет техническую направленность, углубленный уровень сложности.

Программа для обучающихся 14-18 лет, объем годовой нагрузки 144 часа, направлена на приобретение навыков проектной деятельности и разработку собственных кроссплатформенных приложений.

Для освоения данной программы необходимы знания, полученные при изучении курса «Основы программирования».

Знания и навыки, полученные в результате изучения программы, помогут обучающимся освоить базовые понятия в управлении проектами, а также на практике применить теоретические знания для решения собственных проектных задач.

Сегодня важным звеном, а также ожидаемым карьерным развитием специалиста в любой компании и особенно в IT-сфере является деятельность по управлению проектами или «проектный менеджмент». Успех любого проекта в IT, будь то создание веб-сайта, интернет-магазина, робота или запуск нового мобильного приложения, зависит во многом от менеджера проекта (руководителя проектной команды), который наряду с глубокими знаниями в области технологий и программирования, должен владеть навыками, методами и инструментами кроссплатформенной разработки.

Новизна программы

Осознание уникальности и многогранности управления IT-проектами может стать мотивацией обучающихся к погружению в эту сферу профессиональной деятельности, а освоение ее современных подходов и инструментов станет выгодной инвестицией в их будущее, ведь решение любой жизненной задачи можно рассматривать как проект, а методологию управления проектами считать одним из самых эффективных способов успешного достижения поставленной цели.

Программа рассчитана на три этапа обучения:

1. Освоение принципов работы с проектами и определение темы проекта.
2. Отработка на практике различных методологий управления собственными проектами
3. Разработка программной части на основе знаний, полученных в результате прохождения программы «Основы программирования».

Данная программа формирует профессиональные компетенции, которые позволят обучающимся в будущем создавать собственные программы, игры, сайты. Программа направлена на развитие навыков по управлению проектами, совершенствование навыков программирования и разработки ПО, представления о профессии программиста и менеджера по проектам.

Актуальность программы

Большинство курсов и образовательных программ по программированию нацелены только на изучение языка программирования. Данная программа нацелена на изучение программирования с точки зрения реальной разработки в IT-компаниях, и позволит обучающимся увидеть более полную картину работы программиста и разработки проекта.

Изучение основ управления IT-проектами способствует формированию современного мировоззрения, развитию интеллектуальных способностей, развивает способности к самообразованию, умению ставить задачи, планировать их выполнение, умению разбивать решение задачи на подзадачи, способности работать с информацией: находить, оценивать и использовать информацию из различных источников. Обучение по данной программе дает обучающимся возможность реализовать собственные проекты.

Педагогическая целесообразность

Индустрия информационных технологий не похожа ни на одну из других отраслей и имеет свою специфику.

Как правило, над кодом мобильных приложений, игр, корпоративных информационных систем, созданием программно-аппаратных комплексов, цифровых гаджетов, web-сервисов и т.п. работает команда высококлассных профессионалов различной специализации: программисты и инженеры, дизайнеры и тестировщики и т.п., которые превращают идеи в реальные продукты, необходимые миллионам людей. Такими сложными командами необходимо уметь управлять. Поэтому важным звеном в деятельности любой IT-компании является управление проектами или «проектный менеджмент».

К сожалению, в термин «управление проектами» разные люди вкладывают разный смысл, зачастую не имея четкого представления о моделях, методах, инструментах и технологиях управления проектами. Между тем, управление проектами – это профессиональная деятельность, основанная на использовании современных научных знаний, навыков, методов, средств и технологий и ориентированная на эффективное получение желаемых результатов.

Применение методологии управления проектами дает возможность четко определить цели и результаты проекта, дать им количественные характеристики, временные, стоимостные и качественные параметры проекта, создать четкий план проекта, выделить, оценить риски и предотвратить возможные негативные последствия во время реализации проекта.

На сегодняшний день методология управления проектами доказала свое право считаться одним из самых эффективных способов успешной реализации поставленных целей.

Проектная деятельность, связанная с непрерывным улучшением и изменением функциональности IT-продуктов, ближе к изобретательству и творчеству, чем к ремеслу, поэтому не всегда хорошо прогнозируется. Это в свою очередь, служит причиной сложности организации процесса взаимодействия между всеми заинтересованными сторонами: заказчиками,

инвесторами, участниками команды разработчиков, пользователями, увеличению количества связей и точек их пересечения, что объективно вызывает трудности в принятии решений, ведет к ошибкам в продукте, превышению стоимости разработки, затягиванию выполнения графиков работ.

Кроме того, важный фактор разработки программного обеспечения – потенциально огромное количество пользователей продукта труда разработчиков. Компьютерные программы разрабатываются для людей (заказчиков/пользователей) и, следовательно, всегда сопровождается постоянным социальным взаимодействием не только внутри команды, но и с людьми «снаружи».

Таким образом, социальный аспект в разработке программного обеспечения – это очень важная составляющая многих ИТ-проектов, которая отличает их от проектов создания материальных вещей.

Осознание сложности и уникальности управления ИТ-проектами, в свою очередь, может служить и мотивацией учеников к погружению именно в эту сферу профессиональной деятельности, овладению искусством укладываться в сроки, бюджет проекта, рационально использовать ресурсы проекта, организовывать работу команды специалистов - с одной стороны; прогнозировать, каким должен быть и будет продукт, насколько он будет востребован - с другой.

Учитывая, что проектная деятельность свойственна на определенном этапе практически всем видам человеческой деятельности, очевиден и социальный эффект программы.

Цель программы

Формирование навыков работы с технологиями разработки и управления проектами.

Задачи:

- сформировать представление о методах управления проектами;
- сформировать навыки коллективной разработки;
- усовершенствовать навыки разработки, тестирования и отладки несложных программ;
- сформировать самостоятельность и творческий подход к решению проектных задач;
- сформировать навыки работы с документацией к языку программирования и программным продуктам;
- расширить кругозор обучающихся в области программирования.

Отличительные особенности данной дополнительной общеразвивающей программы от существующих образовательных программ

Основной целью программы является создание условий для осознания обучающимися важности построения дальнейшей индивидуальной образовательной траектории и ранней профориентации через знакомство с перспективными направлениями развития ИТ-индустрии, спектром ИТ-

профессий и возможностями профессиональной самореализации, спецификой деятельности различных специалистов.

Предметом и главной темой обсуждения в этой программе станет управление проектами - деятельность, направленная на достижение поставленных задач, реализацию определённых планов, используя имеющиеся ресурсы - время, капитал, людей, материалы и инструменты; минимизацию рисков и отклонений от плана, эффективного управления изменениями. Также знакомство с концепцией кроссплатформенного программирования и разработка одного приложения, которое будет работать на устройствах с разными операционными системами.

В ходе программы важно изменить сложившийся у большинства людей стереотип: чтобы состояться как специалист в IT-индустрии недостаточно научиться хорошо программировать.

Важно показать, что возрастает значение профессиональных компетенций коллективной разработки программного обеспечения, знание современных направлений, методов и технологий управления IT-проектами:

понимание обязанностей различных участников команды по разработке программного обеспечения: руководитель разработки программного обеспечения, руководитель технической группы (teamleader), архитектор, программист, тестировщик, дизайнер, верстальщик, аналитик;

владение современными стратегиями и технологиями организации коллективной разработки программного обеспечения, включая системы управления версиями, процессы непрерывной интеграции, стандарты оформления кода и методы инспекции кода;

понимание основных направлений развития методов коллективной разработки, их отличий и целесообразности применения в зависимости от типа решаемых задач и требований организации;

владение методологиями разработки программных продуктов.

Возраст обучающихся, участвующих в программе

Программа «Основы мобильной разработки» рассчитана на обучающихся 14-18 лет.

Условия вхождения в программу

Набор на Программу осуществляется в соответствии с Правилами приема и отчисления обучающихся автономной некоммерческой организации «Красноярский детский технопарк «Кванториум».

Программа адресована подросткам, прошедшим обучение по ДООП «Основы программирования» и «Проектирование и разработка программного обеспечения для клиент-серверного веб-приложения».

Срок реализации программы

Программа рассчитана на 1 год обучения. Годовая нагрузка на обучающегося составляет 144 часа.

Режим занятий, формы и методы обучения

Учебные занятия проходят по очной форме обучения. Режим занятий – 2 раза в неделю по 2 академических часа (1 академический час 40 минут) с обязательным перерывом, что определяется Санитарно-эпидемиологическими правилами и нормативами СанПиН 2.4.4.3172-14.

Занятия проводятся в комбинированной форме – изложение нового материала, проверка пройденного материала, закрепление полученных знаний, самостоятельная работа.

При проведении занятий используются следующие формы работы:

демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном или экранами компьютеров на ученических рабочих местах;

фронтальная, когда обучающиеся синхронно работают под управлением педагога;

самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия;

повторение и усвоение пройденного материала осуществляется через контрольные и проверочные работы, анализ полученных результатов;

закрепление знаний, умений и навыков через постановку задачи и самостоятельную работу обучающегося под руководством педагога;

применение полученных знаний и навыков через прикладную работу обучающегося, использующего на практике приобретенные компетенции.

Ожидаемые образовательные результаты и эффекты, способы предъявления и отслеживания результатов

Личностное и профессиональное самоопределение (мотивация к получению профессий в наукоемких областях через интерес к организационно-управленческому аспекту в IT-индустрии) - один из важнейших планируемых образовательных результатов.

Знакомя обучающихся с современным методами управления IT-проектами, которые призваны способствовать развитию экономики, улучшить качество жизни и работы людей, ускорить и изменить процессы и способы решения различных задач, следует обратить внимание обучающихся на то, что процессы управления IT-проектами характеризуются многоплановостью и многофункциональностью, требуют от специалистов дополнительных навыков и компетенций, многие из которых связаны как с умением программировать, так и с навыками планирования, принятия решений и продуктивной коммуникации, высокой степенью личной ответственности и профессиональной исполнительности. Освоение начальных форм познавательной и личной рефлексии - одна из основных задач.

Немаловажным является ценный опыт мобильной кроссплатформенной разработки. Кроссплатформенный подход позволяет создавать приложения для различных платформ с одной кодовой базой, что экономит время и деньги и избавляет от ненужных усилий. Проникновение смартфонов в повседневную жизнь растет во всем мире, а выбор мобильной операционной системы является

вопросом личных предпочтений пользователей, а не вопросом производительности или доступности, будет целесообразно в конечном итоге создать мобильное приложение как для Android, так и для iOS.

Отличительной особенностью образовательной программы является ориентация на кооперированную деятельность обучающихся, на развитие у них коммуникативных универсальных учебных действий:

умение организовывать продуктивное сотрудничество и совместную деятельность с одноклассниками;

работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учёта интересов; формулировать, аргументировать и отстаивать своё мнение.

Обучающиеся учатся работать самостоятельно и в команде, вступать в диалог и вести его, выполнять разные роли и обязанности. Приобретаемый ими социальный опыт и навыки практически значимы.

Обучающиеся будут знать:

возможности современных и перспективных средств разработки программных продуктов, технических средств;

методологии разработки программного обеспечения и технологии программирования;

методологии разработки кроссплатформенных мобильных приложений;

методологии управления проектами.

Обучающиеся будут уметь:

вырабатывать варианты реализации программного обеспечения;

проводить оценку и обоснование выбранных решений;

разрабатывать алгоритм задачи;

разрабатывать структуру и интерфейс приложений;

разрабатывать приложения, которые будут стабильно работать на нескольких операционных системах;

работать с документацией к языку и инструментам разработки;

работать в сети интернет для поиска информации;

работать в команде;

управлять собственными проектами.

Сформированные навыки:

работы в изучаемых программных средах;

навыки составления алгоритмов;

применения на практике команд и операторов изучаемых языков;

разработки, тестирования и отладки десктопных программ;

разработки, тестирования и отладки мобильных приложений под разные операционные системы;

работы в сети Интернет для поиска информации;

работы в команде;

коллективного программирования;

управления проектом.

Личностные результаты

Обучающиеся приобретут навыки самостоятельной организации своей деятельности, саморазвития и самовоспитания.

У обучающихся сформируется готовность и способность к самостоятельной, творческой деятельности, к образованию, в том числе самообразованию; готовность к осознанному выбору будущей профессии.

Метапредметные результаты

Умение планировать; умение анализировать; алгоритмизировать; умение организовывать свой труд и труд команды.

Опыт

проектной деятельности;
создания, редактирования, оформления, сохранения, передачи информационных объектов различного типа с помощью современных программных средств;
информационной деятельности в различных сферах;
эффективного применения информационных образовательных ресурсов в учебной деятельности, в том числе самообразовании;
эффективной организации индивидуального информационного пространства.

Механизм оценки результативности

Контроль качества осуществляется в форме тестов, выполнения практических и проектных работ.

По итогам каждого этапа проводится итоговая аттестация в форме представления части проектной работы в форме презентации и демонстрации программной части.

Итоговый контроль освоения образовательной программы осуществляется через защиту индивидуального (группового) ИТ-проекта. Технология проведения итогового контроля - экспертная оценка в рамках НТК (научно-технической конференции) АНО ДТ «Красноярский Кванториум» с привлечением представителей компаний и экспертов в данной области. Данный уровень позволяет участникам получить экспертную обратную связь относительно представленного проекта, а также понять, через комментарии экспертов, перспективы развития проекта.

2.УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Наименование разделов	Всего часов	Теория	Практика
1	Закрепление базовых навыков	21	7	14
	Вводный урок. Правила работы и техника безопасности при работе на компьютере	1	1	0
	Выполнение заданий по C++ краткий повтор пробелов	6	2	4
	Выполнение заданий по Web краткий повтор пробелов	8	2	6
	Выполнение заданий по Python краткий повтор пробелов	6	2	4
2	Управление проектами	34	13	21
	Методологии управления проектом	6	2	4
	Как правильно определить тему проекта	2	1	1
	Формулирование актуальности проекта.	2	1	1
	Поиск проблемы	2	1	1
	Формулировка цели.	2	1	1
	Формулировка задач.	2	1	1
	Поиск аналогов. Анализ.	2	1	1
	Проработка технического решения.	4	1	3
	Изучение технологий производства.	4	2	2
	Составление плана работы.	2	1	1
	Оформление презентации.	4	1	3
	Промежуточная аттестация. Предзащита проектной идеи.	2	0	2
3	Основы разработки под React Native	68	28	40
	Введение в мобильную разработку	2	2	0
	Знакомство с разработкой под Android	6	2	4
	Знакомство с разработкой под IOS	6	2	4
	Введение в React Native	14	8	6
	Платформо-зависимый код	4	2	2
	Разработка текстового интерфейса	2	1	1
	Контролы	4	1	3
	Навигация	4	2	2
	Анимация	4	2	2
	Подъем состояния	6	2	4

	Функциональные компоненты	2	1	3
	Очередность	2	1	1
	История ходов	4	0	4
	Создание игры	6	2	4
4	Основы разработки под Flutter	24	11	13
	Введение в Flutter	2	2	0
	Настройка окружения	2	1	1
	Основы языка Dart	4	2	2
	Добавление экранов	2	1	1
	Анимация и графика	2	1	1
	Плагин Flame	2	1	1
	Отрисовка экрана	2	1	1
	Отрисовка примитивов	2	1	1
	Работа с текстом	2	1	1
	Создание приложения	4	0	4
	ИТОГ	144	56	88

3.СОДЕРЖАНИЕ ПРОГРАММЫ

Закрепление базовых навыков

Тема: Выполнение заданий по C++ и краткий повтор пробелов

Теоретическая работа: повторение ключевых тем модуля C++.

Практическая работа: работа с переменными. Команды if-else. Сравнение чисел. Булева алгебра в программировании. Вложенные циклы. Циклы while, for, do-while. Бесконечный цикл. Выход из цикла. Массивы. Размерности массивов. Указатели. Составление программ с использованием функций. Реализация концепций структурного программирования с помощью функций.

Тема: Выполнение заданий по Web и краткий повтор пробелов

Теоретическая работа: повторение ключевых тем модуля Web.

Практическая работа: Объявление и использование в программе переменных, констант и выражений логического типа. Использование простых и составных условий. Полная и сокращенная формы условного оператора. Создание разветвляющейся программы с помощью условных операторов. Оператор множественного выбора. Организация множественного ветвления в программе. Работа с различными формами циклических операторов: циклами со счетчиком, циклами с предусловием и с постусловием. Основные тэги HTML. Пишем статический сайт. Добавляем стили для сайта.

Тема: Python и краткий повтор пробелов

Теоретическая работа: повторение ключевых тем модуля Python.

Практическая работа: Работа с простыми типами данных: целым, вещественным, символьным и логическим. Организация ввода и вывода данных. Работа с логическим типом данных. Объявление и использование в программе переменных, констант и выражений логического типа. Использование простых и составных условий. Полная и сокращенная формы условного оператора. Создание разветвляющейся программы с помощью условных операторов. Оператор множественного выбора. Организация множественного ветвления в программе. Повтор команд. Переменные цикла. Вложенные циклы. Операторы while, for. Бесконечный цикл. Выход из цикла. Импортирование модулей. Работа с модулями

Управление проектами

Тема: Методологии управления проектом

Теоретическая работа: Что такое проект и командный проект. Отличия проекта от другой командной работы. Основные методологии управления проектом. Гибкие методологии. Scrum. Agile. Водопад. MVP.

Практическая работа: Работа на тренажере. Жизнь как проект. Разбор популярных сервисов на MVP и последующие версии.

Тема: Как правильно определить тему проекта

Теоретическая работа: Откуда берется тема проекта. Проблематика. Виды проектов и зависимость темы проекта от вида.

Практическая работа: Распределение темы на виды проектов. Составление тем готовых проектов.

Тема: Формулирование актуальности проекта.

Теоретическая работа: Обоснование актуальности проекта. Примеры актуальности проекта.

Практическая работа: Определение актуальности тестовых проектов.

Тема: Поиск проблемы

Теоретическая работа: проблемное интервью. HADI-циклы. Анализ проблемы

Практическая работа: Определите основные проблемы, которые существуют в конкретной ситуации. Определить, чья это проблема.

Тема: Формулировка цели.

Теоретическая работа: На какой вопрос должна отвечать цель. Примеры целей проектов.

Практическая работа: Определение цели тестовых проектов.

Тема: Формулировка задач.

Теоретическая работа: На какой вопрос должна отвечать задача. Примеры задач проектов. Отличия цели от задачи и их связь.

Практическая работа: Определение задач к целям тестовых проектов.

Тема: Поиск аналогов. Анализ.

Теоретическая работа: понятие аналога и анализа. Поиск аналогов и методы анализа информации.

Практическая работа: построение сравнительной таблицы аналогов.

Тема: Проработка технического решения.

Теоретическая работа: Сферы разработки проектов. Виды проектов в IT. Технические особенности каждого вида.

Практическая работа: Определение сферы проекта. Определение технического решения для конкретного проекта. Особенности реализации.

Тема: Изучение технологий производства.

Теоретическая работа: Определение сложности проекта. Роли в проектах разных видов. Распределение компетенций.

Практическая работа: Распределение ролей в зависимости от компетенций. Определение сложности конкретного проекта.

Тема: Составление плана работы.

Теоретическая работа: Основные шаги для составления плана работы.

Практическая работа: Примерный календарный план. Объем работ. Распределение ролей.

Тема: Оформление презентации.

Теоретическая работа: Структура презентации. Сценарий. Содержание. Подача. Дизайн.

Практическая работа: Разработка сценария презентации. Разработка презентации в tilda или power point. Разработка и отработка выступления.

Промежуточная аттестация

Практическая работа: Предзащита проектной идеи.

Основы разработки под React Native

Тема: Введение в мобильную разработку

Теоретическая работа: Что такое мобильная разработка. Чем отличается разработка под IOS и Android. Как влияет ОС?

Тема: Введение в React Native

Теоретическая работа: Чем нативная разработка отличается от кроссплатформенной. Методы работы с React и построение приложения.

Практическая работа: Создание переменных. Типизация. Виды циклов. Условия. Массивы.

Тема: Платформено-зависимый код

Теоретическая работа: Понятие платформено-зависимого кода

Практическая работа: Компоненты операционной системы Android, компоненты операционной системы IOS.

Тема: Разработка текстового интерфейса

Теоретическая работа: Компоненты текстового форматирования.

Практическая работа: Работа с текстом. Создание базового текстового интерфейса.

Тема: Компоненты

Теоретическая работа: Структура компонентов.

Практическая работа: Добавление в приложение функциональных и классовых компонентов.

Тема: Навигация

Теоретическая работа: Основные библиотеки навигации. Плюсы и минусы react-navigation

Практическая работа: react-navigation, wix/react-native-navigation, react-router, airbnb/native-navigation

Тема: Анимация

Теоретическая работа: Компонент Animated

Практическая работа: Как сделать анимацию на iOS. Как сделать анимацию на Android. Почему не получается сделать кроссплатформенно.

Тема: Подъем состояния

Теоретическая работа: Начало создания игры крестики-нолики. Хранение состояния игры в компоненте.

Практическая работа: Конструктор игрового поля. Массив с состоянием игры.

Тема: Функциональные компоненты

Теоретическая работа: Метод `render`. Разница функциональных компонентов от классов.

Практическая работа: Замена класса для отметки на функциональный компонент.

Тема: Очередность

Теоретическая работа: Определение очереди и выбор реализации.

Практическая работа: Разработка сценария игры. Вставка метода чередования ходов.

Тема: История ходов

Практическая работа: Создание массива с массивами сыгранных полей.

Тема: Создание игры

Теоретическая работа: Подъем состояния из предыдущих ходов.

Практическая работа: Доработка игры.

Промежуточная аттестация: Защита игры

Основы разработки под Flutter

Тема: Введение в Flutter

Теоретическая работа: Что за платформа? Какие приложения работают на Flutter. Какой языка использует Flutter.

Тема: Настройка окружения

Теоретическая работа: Какой IDE пользоваться? Эмуляция устройств с разными операционными системами.

Практическая работа: Настройка IDE. Настройка эмулятора. Запуск "Hello world".

Тема: Основы языка Dart

Теоретическая работа: Экскурсия по языку Dart. Основные конструкции.

Практическая работа: Разработка базового приложения. Отработка циклов, условий, переменных и работа с ними.

Тема: Добавление экранов

Теоретическая работа: Добавление экранов. Разница экранов от активностей.

Практическая работа: Добавление нескольких экранов и переключение между ними. Постройка сценария приложения.

Тема: Анимация и графика

Теоретическая работа: Выбор подхода к анимации. Неявная анимация. Анимация героев.

Практическая работа: Основные концепции и классы анимации. Поэтапная анимация

Тема: Плагин Flame

Теоретическая работа: Плагин для разработки игр.

Практическая работа: игровой цикл, компонентная/объектная система, связывает физический движок (box2d), аудио, поддержка эффекты и частицы поддержка жестов и ввода

Тема: Отрисовка экрана

Теоретическая работа: Основы работы с графикой и порядком отрисовки.

Практическая работа: Отрисовка макета приложения.

Тема: Отрисовка примитивов

Теоретическая работа: Как реализуется создание интерфейса в iOS и Android.

Практическая работа: Отрисовка интерфейса, используя графические примитивы flutter.

Тема: Работа с текстом

Теоретическая работа: Форматирование текста

Практическая работа: Методы работы с текстом, создание текстовых полей и текстовых уведомлений.

Тема: Создание приложения

Практическая работа: Разработка собственного кроссплатформенного приложения.

Итоговая работа: Защита приложения.

4.МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ПРОГРАММЕ

Помещения для лабораторных занятий и самостоятельной работы обучающихся должны быть оснащены персональными компьютерами с установленной средой разработки программного обеспечения на языке объектно-ориентированного программирования Microsoft Visual Studio Community 2019 (и выше), IDE PyCharm 2018.3.5 (и выше), Python 3.7.4 (и выше), Visual Studio Code 1.35.0 (и выше), Android Studio 4.1 расширение для VSCode - Live Server, расширение для VSCode - CSS Peak, расширение для VSCode - Prettier, Microsoft Office 2013 (и выше), с возможностью подключения к сети Интернет и доска с маркером или интерактивная доска. Помещение для занятий лекционного типа должно быть оснащено мультимедийным проектором и персональными компьютерами с установленной средой разработки программного обеспечения на языке объектно-ориентированного программирования Microsoft Visual Studio Community 2019 (и выше), Microsoft Office 2013 (и выше), с возможностью подключения к сети Интернет, доска с маркером или интерактивная доска.

5.СПИСОК ЛИТЕРАТУРЫ

1. Борович П.С. Бутко Е.Ю., 2011 г. Учебное пособие «Среда программирования Scratch»
2. . Прата, С. Язык программирования C++: Лекции и упражнения / С. Прата. – М.: Вильямс, 2005. – 1097 с.
3. Шилдт, Г. Теория и практика C++ / Г. Шилдт. – СПб.: BHV, 1996. – 416 с.
4. <http://scratch.mit.edu>
5. <https://en.cppreference.com/w/>
6. <https://www.python.org>
7. <https://learn.javascript.ru>
8. <https://flutter.dev/docs/development/packages-and-plugins/developing-packages>
9. <https://reactnative.dev/docs/getting-started>