

Министерство образования Ярославской области
Государственное профессиональное образовательное
автономное учреждение Ярославской области
Ростовский колледж отраслевых технологий

Центр цифрового образования детей «IT-куб»

УТВЕРЖДАЮ:

Директор ГПОАУ ЯО
Ростовский колледж
отраслевых технологий
Т.Н. Кудрявцева



**Дополнительная общеобразовательная
общеразвивающая программа технической
направленности**

«МОБИЛЬНАЯ РАЗРАБОТКА»

Направленность: техническая
Уровень программы: продвинутый
Срок реализации: 1 год (144 часа)
Возраст обучающихся : 12-16 лет

2024 год

СОДЕРЖАНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	3
УЧЕБНЫЙ ПЛАН	13
СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА.....	16
КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ	23
СПИСОК ЛИТЕРАТУРЫ	26
Приложение 1	27
Приложение 2	28

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1. Нормативно-правовая база разработки и реализации программы.

Программа разработана на основе следующих нормативных документов:

- Федерального закона от 29.12.2012 № 273-ФЗ (ред. от 31.07.2020) «Об образовании в Российской Федерации» (с изм. и доп., вступ. в силу с 01.09.2020);
- Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденного приказом Министерства просвещения Российской Федерации от 27.07.2022 г. № 629;
- Постановления Главного государственного санитарного врача Российской Федерации 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»;
- Постановления Главного государственного санитарного врача Российской Федерации от 28.01.2021 № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»;
- Приказа Минобрнауки России №882, Минпросвещения России №391 от 05.08.2020 «Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ» (вместе с «Порядком организации и осуществления образовательной деятельности при сетевой форме реализации образовательных программ»);
- Методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб» (письмо Министерства просвещения Российской Федерации от 10 ноября 2021 г. № ТВ-1984/04);
- Стратегии развития воспитания в Российской Федерации на период до 2025 года (Утверждена распоряжением Правительства РФ от 29.05.2015 № 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»);

– Концепции развития дополнительного образования детей до 2030 года, утвержденной распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р.

2. Новизна, актуальность, педагогическая целесообразность реализации программы.

Актуальность программы обусловлена интересом общества к информационным технологиям, мобильным устройствам и развитию Российского рынка программного обеспечения. Количество пользователей мобильными телефонами на операционных системах Android и iOS растет с каждым днем. Сегодня специалистами в области информационных технологий разрабатываются мобильные приложения, которые позволяют решать огромное количество задач. Кроме того, большое внимание уделяется программированию и реализации различных алгоритмов на современном языке программирования Java.

Программу отличает новизна содержания. В программе предусмотрен краткое повторение основ языка Java, после которого обучающиеся приступают к изучению структуры мобильного приложения на Android и разработке приложений в Android Studio.

Педагогическая целесообразность программы определяется образовательным замыслом. Особенность программы «Мобильная разработка» заключается в постепенном усложнении материала: изучение начинается с построения блок-схем алгоритмов и основ языка программирования Java, создания консольных приложений и к концу курса обучающиеся смогут реализовывать мобильные приложения для вычисления сложных параметров и простых логических мобильных игр.

Направленность программы: техническая.

3. Адресат программы.

Адресатом программы являются дети в возрасте от 12 до 16 лет.

Содержание и объем стартовых знаний, необходимых для начального этапа освоения программы: базовые навыки работы с текстовыми редакторами, браузерами, поисковыми системами, файловыми менеджерами (проводником), знание типов данных, основных алгоритмических структур в Java.

4. Срок реализации программы.

Срок реализации программы составляет 1 год.

Уровень программы: продвинутый.

Программа предполагает использование форм организации материала, обеспечивающих доступ к сложным (возможно узкоспециализированным) и нетривиальным разделам в рамках содержательно-тематического направления программы. Также предполагает углубленное изучение содержания программы и доступ к около профессиональным и профессиональным знаниям в рамках содержательно-тематического направления программы.

5. Форма реализации программы.

Форма обучения – очная.

Образовательные технологии: информационные технологии, проектная технология, здоровьесберегающие технологии, технология проблемного обучения.

Форма организации содержания и процесса педагогической деятельности – комплексная.

Тип организации работы учеников: групповая работа, индивидуальная, коллективная.

Виды занятий: лекции и практические занятия.

Наполняемость группы: от 10 до 12 человек.

6. Объем программы и режим работы

Объем программы: 144 часа.

Режим занятий: 2 - 3 раза в неделю по 2 академических часа.

Продолжительность часа – 40 минут.

7. Цель программы:

Целью программы является создание условий для освоения обучающимися перспективной практики разработки мобильных приложений для Android.

8. Задачи программы.

Программа направлена на решение следующих задач:

- сформировать представления о современных мобильных операционных системах, об архитектуре приложений для мобильной платформы Android;
- сформировать навыки реализации различных алгоритмов на языке программирования Java;
- сформировать умения создавать типовые мобильные приложения на языке Java;
- научить подготавливать контент для мобильных приложений;
- способствовать формированию алгоритмического и логического мышления;
- способствовать получению практического опыта проектной деятельности;
- способствовать формированию познавательных и регулятивных универсальных учебных действий;
- совершенствовать навык поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
- формировать коммуникативные навыки (диалогическая и монологическая речь при защите проектов).
- развивать рефлексивную деятельность учащихся;
- воспитывать трудолюбие, целеустремленность, уважение к труду;
- воспитывать информационную культуру.

9. Планируемые результаты освоения программы

В результате освоения программы, учащиеся будут:

Знать:

- основные современные мобильные системы и их отличия;
- основы разработки мобильных приложений в Android Studio;
- предназначение элементов пользовательского интерфейса, папок и файлов в проекте.

Уметь:

- описывать данные на языке XML;
- разрабатывать алгоритмы различной сложности на языке Java;
- создавать простые мобильные приложения в Android Studio;
- создавать анимированные приложения;

Владеть:

- основной терминологией в области мобильной разработки;
- приемами обработки изображений для загрузки в элементы приложения;

Метапредметные результаты:

- доводить начатые проекты до конца;
- выполнять основные логические действия (анализ, синтез, установление причинно-следственных связей);
- представлять проект, отвечать на вопросы по содержанию проекта;
- оценивать свои проекты и проекты своих одноклассников по заданным критериям;
- владеть приемами поиска информации в сети интернет.

Личностные результаты:

- развить навыки по планированию и регулированию собственной деятельности по реализации проекта;
- развитие внимательности, аккуратности, дисциплинированности, усидчивости в процессе проектной деятельности;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками.

10. Формы представления результатов

Формы аттестации: беседа, проверочная работа, демонстрация решения, наблюдение.

Входной контроль осуществляется в начале реализации программы в форме наблюдения и имеет диагностические задачи. Цель входной диагностики – зафиксировать начальный уровень подготовки обучающихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью.

Наблюдение осуществляется в течение реализации программы.

Промежуточный контроль осуществляется в целях диагностики теоретических знаний и практических умений и навыков по итогам освоения одного из разделов курса. Проводится в форме решения практических задач.

Итоговый контроль проводится с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения – представляет из себя решение итогового тестирования.

11. Оценочные материалы, формирующие систему оценивания

Промежуточная аттестация состоит из 2 практических задач, в которых требуется создание приложения для мобильного устройства на Android. Примеры задач приведены в приложении 1. Максимальный балл за промежуточную аттестацию: 40 баллов.

Оценивание тестирования осуществляется по следующим уровням:

высокий уровень – учащийся набрал не менее 80% от максимально возможного количества баллов (от 32 баллов).

средний уровень – учащийся набрал не менее 50% от максимально возможного количества баллов (от 20 баллов).

низкий уровень – учащийся набрал менее 50% от максимально возможного количества баллов (менее 20 баллов).

Итоговый контроль проводится с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения – представляет из себя защиту проекта.

Критерии оценки проекта

№	Название критерия	Максимальный балл
1.	Актуальность и проработанность проблемы	До 5 баллов
2.	Четкость формулировки целей и задач	До 5 баллов
3.	Технологическая сложность проекта	До 10 баллов
4.	Новизна и оригинальность решения	До 5 баллов
5.	Качество разработанного продукта	До 5 баллов
6.	Защита проекта: <ul style="list-style-type: none"> • качество презентации; • четкость и ясность изложения, умение взаимодействовать с аудиторией, отвечать на вопросы 	До 5 баллов
7.	Наличие самооценки и перспектив дальнейшей разработки проекта	До 5 баллов
Итого		40 баллов

Оценивание тестирования осуществляется по следующим уровням:

высокий уровень – учащийся набрал не менее 80% от максимально возможного количества баллов (от 32 баллов).

средний уровень – учащийся набрал не менее 50% от максимально возможного количества баллов (от 20 баллов).

низкий уровень – учащийся набрал менее 50% от максимально возможного количества баллов (менее 20 баллов).

Общими критериями оценки результативности обучения являются:

- оценка уровня теоретических знаний: свобода восприятия теоретической информации, осмысленность и свобода использования специальной терминологии, свобода ориентации в теоретическом материале;

- оценка уровня практической подготовки: соответствие развития уровня практических умений и навыков программным требованиям, свобода владения специальным оборудованием и оснащением;

- оценка уровня достижения личностных результатов: культура организации самостоятельной деятельности, культура работы с информацией, аккуратность и ответственность при работе.

Оценка итоговых результатов освоения программы осуществляется по трем уровням:

Высокий уровень – достижение 80- 100% показателей освоения программы.

Средний уровень – достижение 50- 79% показателей освоения программы.

Низкий уровень - достижение менее чем 50% показателей освоения программы.

Достигнутые обучающимся знания, умения и навыки заносятся в сводную таблицу результатов обучения.

Оценка уровней освоения программы

Уровни	Параметры	Показатели
Высокий уровень (80-100%)	Теоретические знания	Обучающийся освоил материал в полном объеме. Знает и понимает значение терминов, употребляет их осознанно и в полном соответствии с содержанием. Самостоятельно ориентируется в содержании материала по темам.
	Практические умения и навыки	Обучающийся овладел 80-100% умений и навыков, предусмотренных программой за конкретный период. Умет работать самостоятельно, применяя практические умения и навыки. Правильно и по назначению применяет инструменты. Умеет выполнять основные логические действия (анализ, синтез, установление причинно-следственных связей). Способен планировать и регулировать свою деятельность по реализации проекта. Умеет осуществлять поиск информации, в том числе в сети Интернет; выслушивать собеседника и вести диалог; выражать свои мысли в соответствии с задачами и условиями коммуникации.
	Личностные результаты	Обучающийся обладает внутренней мотивацией. Способен самостоятельно организовать собственную деятельность. Сформирована культура работы с информацией. Работу выполняет аккуратно, доводит до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.

Средний уровень (50-79%)	Теоретические знания	Учащийся освоил базовые знания, ориентируется в содержании материала по темам, иногда обращается за помощью к педагогу. Использует специальную терминологию, однако сочетает её с бытовой
	Практические умения и навыки	Владеет базовыми навыками и умениями, но не всегда может выполнить задание самостоятельно, просит помощи педагога. В основном выполняет задания на основе образца. Способен разрабатывать алгоритм решения при помощи преподавателя. Делает ошибки в работе, но может устранить их после наводящих вопросов или самостоятельно Испытывает незначительные затруднения при выполнении основных логических действий (анализ, синтез, установление причинно-следственных связей). Способен планировать и регулировать свою деятельность по реализации проекта с помощью педагога. Испытывает незначительные сложности в осуществлении коммуникации с педагогом и сверстниками.
	Личностные результаты	Внутренняя мотивация к обучению сочетается с внешней. В работе допускает небрежность. Работу не всегда выполняет аккуратно и/или доводит до конца. Оценить результаты своей деятельности может с подсказкой педагога.
Низкий уровень (меньше 50%)	Теоретические знания	Владеет минимальными знаниями, ориентируется в содержании материала по темам только с помощью педагога. Избегает употреблять специальные термины.
	Практические умения и навыки	Владеет минимальными начальными навыками и умениями. Учащийся способен выполнять каждую операцию только с подсказкой педагога или товарищей. В работе допускает грубые ошибки, не может их найти их даже после указания преподавателя. В состоянии выполнять лишь простейшие практические задания педагога. Испытывает существенные затруднения при выполнении основных логических действий (анализ, синтез, установление причинно-следственных связей). Не способен планировать и регулировать свою деятельность по реализации проекта. Испытывает значительные сложности в осуществлении коммуникации с педагогом и сверстниками.
	Личностные результаты	Преобладает внешняя мотивация к обучению. Работу часто выполняет неаккуратно и/или не доводит до конца. Не способен самостоятельно и объективно оценить результаты своей работы.

Сводная таблица результатов обучения по программе

№ п/п	Фамилия, имя обучающегося	Оценка теоретических знаний	Оценка практических умений и навыков (предметных и метапредметных)	Личностные результаты	Итоговая оценка
----------	------------------------------	-----------------------------------	--	--------------------------	--------------------

УЧЕБНЫЙ ПЛАН

№ п/п	Название разделов, тем	Количество часов			Форма аттестации/ контроля
		Всего	Теория	Практика	
1.	Инструктаж по технике безопасности. Актуализация знания по теме «Алгоритмы: понятия, свойства, способы представления информации»	2	1	1	Беседа
2.	Алгоритмы: свойства и способы представления. Построение блок схем линейных алгоритмов и их реализация в Java	2	1	1	
3.	Построение блок схем разветвляющихся алгоритмов и их реализация в Java	4	1	3	
4.	Построение блок схем циклических алгоритмов и их реализация в Java	4	1	3	
5.	Методы класса ArrayList	2	1	1	
6.	Решение задач с использованием класса ArrayList	4	0	4	Демонстрация решения
7.	Повторение темы: символьные, строковые типы данных, массивы в Java.	4	1	3	Демонстрация решения
8.	Классы и объекты. Конструкторы классов.	2	1	1	
9.	Методы и их перегрузка. Статические поля и методы.	2	1	1	
10.	Разработка собственных классов.	2	0	2	Демонстрация решения
11.	Принципы объектно – ориентированного программирования (ООП): наследование, полиморфизм инкапсуляция.	2	1	1	
12.	Разработка и использование классов с использованием принципов ООП.	2	0	2	Демонстрация решения
13.	Промежуточная аттестация	2	0	2	Проверочная работа
14.	Язык разметки XML.	2	1	1	
15.	Изучение структуры приложения для Android. Настройка среды разработки.	2	1	1	
16.	Основные элементы управления: кнопки, поля для ввода и поля для вывода текста.	2	1	1	Демонстрация решения

17.	Элементы компоновки элементов графического интерфейса. Основные свойства элементов графического интерфейса.	4	1	3	Демонстрация решения
18.	Разработка простых приложений с использованием элементов управления Button, EditText, TextView	4	0	4	Демонстрация решения
19.	Реализация диалога с пользователем.	2	1	1	
20.	Элементы управления ToggleButton, CheckBox, SeekBar, RadioButton.	2	1	1	
21.	Разработка простых приложений с использованием элементов управления ToggleButton, CheckBox, SeekBar, RadioButton.	4	0	4	Демонстрация решения
22.	Файлы ресурсов и локализация приложений.	2	1	1	
23.	Разработка мультязычного приложения.	4	0	4	Демонстрация решения
24.	Элементы управления Spinner и AutoComliteTextView.	2	1	1	
25.	Разработка простых приложений с использованием элементов управления Spinner и AutoComliteTextView.	4	0	4	Демонстрация решения
26.	Элемент управления ListView. Адаптеры списков.	2	1	1	
27.	Разработка приложений с использованием компонента ListView и стандартных макетов.	4	0	4	Демонстрация решения
28.	Разработка приложений с использованием ListView и пользовательских макетов.	4	1	3	Демонстрация решения
29.	Многооконные мобильные приложения. Передача данных между окнами.	2	1	1	Демонстрация решения
30.	Разработка многооконных приложений	4	0	4	Демонстрация решения
31.	Разработка авторизации с использованием FireBase	2	1	1	
32.	Основы баз данных	4	1	3	
33.	Разработка приложения с использованием базы данных в FireBase	4	0	4	Демонстрация решения
34.	Сохранение параметров. Создание меню в приложении.	2	1	1	

35.	Класс для анимации объектов Animation.	2	1	1	
36.	Создание приложения с использованием класса Animation.	4	0	4	Демонстрация решения
37.	Компонент для создания анимации SurfaceView.	2	1	1	
38.	Создание анимированного приложения с использованием SurfaceView	4	0	4	Демонстрация решения
39.	Создание игрового приложения с использованием SurfaceView	4	0	4	Демонстрация решения
40.	Определение и согласование темы и цели итогового проекта.	4	1	3	
41.	Планирование работы над итоговым проектом.	2	1	1	
42.	Исследование предметной области проекта	4	0	4	
43.	Подготовка контента приложения	4	0	4	
44.	Создание макетов приложения	4	0	4	
45.	Демонстрация идей проектов	2	0	2	
46.	Программирование логики программ	4	0	4	
47.	Тестирование программ и исправление ошибок	4	0	4	
48.	Подготовка к защите проектов	2	0	2	
49.	Демонстрация проектов	2	0	2	Защита проекта
	Итого	144	27	117	

СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

1. Инструктаж по технике безопасности. Актуализация знания по теме «Алгоритмы: понятия, свойства, способы представления информации».

Теория (1 час): инструктаж о правилах поведения на занятиях и технике безопасности.

Практика (1 час): повторение понятия алгоритм и видов представления в виде викторины

2. Алгоритмы: свойства и способы представления. Построение блок схем линейных алгоритмов и их реализация в Java.

Теория (1 час): изображение и примеры использования основных элементов блок – схем.

Практика (1 час): построение блок – схем линейных алгоритмов для решения вычислительных задач и их реализация в Java.

3. Построение блок схем разветвляющихся алгоритмов и их реализация в Java.

Теория (1 час): изображение блок схем с ветвлением, примеры использования блоков ветвления.

Практика (3 час): построение блок – схем разветвляющихся алгоритмов для решения вычислительных задач и их реализация в Java.

4. Построение блок схем циклических алгоритмов и их реализация в Java.

Теория (1 час): изображение блок схем с циклами и примеры использования блоков цикла.

Практика (3 часа): построение блок – схем циклических алгоритмов для решения вычислительных задач и их реализация в Java.

5. Методы класса ArrayList.

Теория (1 час): динамический массив, методы класса ArrayList.

Практика (1 час): создание программ для обработки массивов с разной длиной.

6. Решение задач с использованием класса ArrayList.

Практика (4 час): решение задач с использованием динамических массивов.

7. Повторение темы: символьные, строковые типы данных, массивы в Java.

Теория (1 час): символьный тип данных, получение кода символа. Функции обработки строк, работа со строками как с массивом символов.

Практика (3 час): реализация алгоритмов обработки строк.

8. Классы и объекты. Конструкторы классов.

Теория (1 час): классы, объекты, поля и методы классов.

Практика (1 час): создание программы с использованием собственного класса.

9. Методы и их перегрузка. Статические поля и методы.

Теория (1 час): методы класса, перегрузка. Статичные поля и методы класса.

Практика (1 час): создание программ с использованием классов с перегруженными методами.

10. Разработка собственных классов.

Практика (4 час): разработка пользовательских классов и программ с их использованием.

11. Принципы объектно – ориентированного программирования (ООП): наследование, полиморфизм инкапсуляция.

Теория (1 час): принципы ООП. Реализации наследования, инкапсуляции и полиморфизма в Java.

Практика (1 час): создание программ с использованием объектно-ориентированного подхода.

12. Разработка и использование классов с использованием принципов ООП.

Практика (2 час): создание программ с использованием объектно – ориентированного подхода.

13. Промежуточная аттестация.

Практика (2 часа): тестирование.

14. Язык разметки XML.

Теория (1 час): структура XML – документа. Элементы, атрибуты и тэги. Примеры XML – документов.

Практика (1 час): создание XML – документа, описывающего поля ранее реализованного класса.

15. Изучение структуры приложения для Android. Настройка среды разработки.

Теория (1 час): структура проекта для Android. Манифест, файлы ресурсов. Эмуляторы и реальные устройства.

Практика (1 час): создание приложения «Hello, world!». Запуск приложения на реальном устройстве.

16. Основные элементы управления: кнопки, поля для ввода и поля для вывода текста.

Теория (1 час): атрибуты основных элементов управления.

Практика (1 час): атрибуты основных элементов управления.

17. Элементы компоновки элементов графического интерфейса. Основные свойства элементов графического интерфейса.

Теория (1 час): единицы измерения высоты и ширины элементов. Layout, LinearLayout, RelativeLayout, TableLayout, GridLayout, ScrollView. Вложенные layout. Свойства gravity и layout_gravity.

Практика (3 часа): разработка приложений с различными элементами компоновки.

18. Разработка простых приложений с использованием элементов управления Button, EditText, TextView.

Практика (4 часа): разработка приложений с использованием различных элементов управления и компоновки.

19. Реализация диалога с пользователем.

Теория (1 час): вызов и настройка Toast сообщений. Классы AlertDialog, DatePickerDialog, TimePickerDialog.

Практика (1 часа): создание приложений с использованием Toast, AlertDialog, DatePickerDialog, TimePickerDialog.

20. Элементы управления ToggleButton, CheckBox, SeekBar, RadioButton.

Теория (1 час): предназначение и атрибуты элементов управления ToggleButton, CheckBox, SeekBar, RadioButton.

Практика (1 часа): создание приложения с использованием ToggleButton, CheckBox, SeekBar, RadioButton.

21. Разработка простых приложений с использованием элементов управления ToggleButton, CheckBox, SeekBar, RadioButton.

Практика (4 часа): создание приложений с использованием элементов управления ToggleButton, CheckBox, SeekBar, RadioButton.

22. Файлы ресурсов и локализация приложений.

Теория (1 час): локализация строк и изображений.

Практика (1 часа): создание мультязычных приложений.

23. Разработка мультязычного приложения.

Практика (4 часа): создание мультязычных приложений.

24. Элементы управления Spinner и AutoCompliteTextView.

Теория (1 час): предназначение и свойства элементов Spinner и AutoCompliteTextView. Стандартные адаптеры списков.

Практика (1 часа): создание приложений с использованием Spinner и AutoCompliteTextView.

25. Разработка простых приложений с использованием элементов управления Spinner и AutoCompliteTextView.

Практика (4 часа): создание приложений с использованием Spinner и AutoCompliteTextView.

26. Элемент управления ListView. Адаптеры списков.

Теория (1 час): отображение таблиц и списков. Пользовательские адаптеры списков.

Практика (1 часа): создание приложений с использованием стандартных адаптеров и атрибута entries.

27. Разработка приложений с использованием компонента ListView и стандартных макетов.

Практика (4 часа): создание приложений с использованием стандартных макетов и файла строковых ресурсов strings.h

28. Разработка приложений с использованием ListView и пользовательских макетов.

Теория (1 час): класс ArrayAdapter. Принцип работы с ListView с использованием адаптеров.

Практика (3 часа): создание приложений с пользовательскими макетами строк, адаптерами и табличным представлением данных.

29. Многооконные мобильные приложения. Передача данных между окнами.

Теория (1 час): передача данных между окнами. Намерения (intent). Обработка значений, возвращаемых при закрытии окна.

Практика (1 часа): создание многооконных приложений.

30. Разработка многооконных приложений.

Практика (4 часа): создание приложений с передачей данных между окнами.

31. Разработка авторизации с использованием FireBase.

Теория (1 час): подключение к базе данных, запрос данных, добавление данных.

Практика (1 часа): создание экрана авторизации.

32. Основы баз данных.

Теория (1 час): SQL – запросы для добавления, удаления, редактирования и выборки данных.

Практика (3 часа): составление SQL – запросов для добавления, удаления, редактирования и выборки данных.

33. Разработка приложения с использованием базы данных в FireBase.

Практика (4 часа): разработка приложения с использованием базы данных.

34. Сохранение параметров. Создание меню в приложении.

Теория (1 час): методы `onRestoreInstanceState`, `onSaveInstanceState`. Класс `Bundle`. Класс `SharedPreferences`. XML – разметка файлов меню, подключение главного и контекстного меню. Добавление и настройка.

Практика (1 часа): создание приложений с меню, сохранением состояния и использованием класса `SharedPreferences`.

35. Класс для анимации объектов Animation.

Теория (1 час): класс `Animation` и методы работы с ним. Описание анимации в XML – файле.

Практика (1 часа): создание приложений с использованием класса `Animation`.

36. Создание приложения с использованием класса Animation.

Практика (4 часа): создание приложения с использованием класса `Animation`.

37. Компонент для создания анимации SurfaceView.

Теория (1 час): класс `SurfaceView` и методы работы с ним.

Практика (1 часа): создание приложения с анимацией, реализованной с использованием класса `SurfaceView`.

38. Создание анимированного приложения с использованием SurfaceView.

Практика (4 часа): создание анимации различной сложности с использованием класса `SurfaceView`.

39. Создание игрового приложения с использованием SurfaceView.

Практика (4 часа): создание мобильной игры с использованием класса `SurfaceView`.

40. Определение и согласование темы и цели итогового проекта.

Теория (1 час): введение в проектную деятельность. Цели, задачи проекта.

Практика (3 часа): выбор темы, определение целей и задач проекта.

41. Планирование работы над итоговым проектом.

Теория (1 час): этапы работы над проектом. Правила эффективного планирования.

Практика (1 часа): составление плана работы над проектом.

42. Исследование предметной области проекта.

Практика (4 часа): подбор и изучение материалов о выбранной предметной области.

43. Подготовка контента приложения.

Практика (4 часа): подготовка изображений, текстов и другого контента для проекта.

44. Создание макетов приложения.

Практика (4 часа): создание макетов приложения.

45. Демонстрация идей проектов.

Практика (2 часа): защита идей проектов.

46. Программирование логики программ.

Практика (4 часа): программирование функционала программ для итоговых проектов.

47. Тестирование программ и исправление ошибок.

Практика (4 часа): тестирование и исправление ошибок в работе приложения для итогового проекта.

48. Подготовка к защите проектов.

Практика (2 часа): подготовка речи и презентации для защиты проекта.

49. Демонстрация проектов.

Практика (2 часа): защита проектов.

КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

Материально-техническое обеспечение

Комплекс условий реализации программы:

Аппаратное и техническое обеспечение:

а) Рабочее место учащегося

- ноутбук с выходом в сеть Интернет;

б) Рабочее место наставника

- ноутбук с выходом в сеть Интернет;

- технические средства обучения (мультимедийное устройство).

Методическое обеспечение программы

Методы обучения:

- объяснительно-иллюстративный (беседы, объяснения);

- репродуктивный (деятельность обучаемых носит алгоритмический характер, выполняется по инструкциям, предписаниям, правилам в аналогичных, сходных с показанным образцом ситуациях);

- метод проблемного изложения;

- эвристический (метод обучения заключается в организации активного поиска решения выдвинутых в обучении (или самостоятельно сформулированных) познавательных задач в ходе подготовки и реализации творческих проектов);

- исследовательский.

Педагогические технологии: информационные технологии, проектная технология, здоровьесберегающие технологии, технология проблемного обучения.

В процессе обучения учащиеся работают с текстовыми редакторами, тренажерами быстрого набора текста, табличными редакторами, СУБД, средой разработки Android Studio.

Проектная технология дает возможность самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развивает критическое и творческое мышление, создаёт условия для формирования и

развития внутренней мотивации учащихся к более качественному овладению знаниями, повышения мыслительной активности и приобретения навыков логического мышления.

Здоровьесберегающие технологии позволяют создать максимально возможные условия для сохранения, укрепления и развития эмоционального, интеллектуального и физического здоровья, в том числе в условиях работы с компьютерной техникой.

Проблемное обучение — это тип развивающего обучения, содержание которого представлено системой проблемных задач различного уровня сложности, в процессе решения которых учащиеся овладевают новыми знаниями и способами действия, а через это происходит формирование творческих способностей: продуктивного мышления и познавательной мотивации.

Учебно-методические средства обучения

Для реализации программы используется:

- специализированная литература по направлению, подборка журналов,
- образцы программ и систем, выполненные учащимися и педагогом,
- учебно-методические пособия для педагога и учащихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное и видео оборудование.

Применяемое на занятиях дидактическое и учебно-методическое обеспечение включает в себя электронные учебники, справочные материалы и системы используемых Программ, Интернет, рабочие тетради учащихся.

Кадровое обеспечение

Программу реализуют педагоги структурного подразделения Центр цифрового образования детей «IT-куб».

Информационное обеспечение

Для реализации программы планируется использование следующих информационных ресурсов:

- самоучитель по разработке на Java для Android (<https://metanit.com/java/android/>);
- официальная документация по разработке на Android (<https://developer.android.com/docs>);
- клавиатурный тренажер Stamina (<https://stamina-online.com/ru>).

СПИСОК ЛИТЕРАТУРЫ

Список литературы для педагогов

1. Блох, Дж. Java: эффективное программирование / Дж. Блох. - М.: Диалектика, 2019. - 464 с.
2. Гриффитс Дэвид, Гриффитс. Head First. Программирование для Android. 2-е изд. — СПб.: Питер, 2018. — 912 с.
3. Дейтел П., Дейтел Х., Уолд А. Android для разработчиков. 3-е изд. — СПб.: Питер, 2016. — 512 с.
4. Харди Б, Филипс Б., Стюарт К., Марсикано К Android. Программирование для профессионалов 2-е изд.— СПб: Питер, 2016.
5. Шилдт Г. Java. Полное руководство, 10-е изд.: Пер. с англ. – СПб. : ООО «Альфа – книга», 2018. – 1488 с.

Список литературы для учащихся

1. Васильев, А.Н. Программирование на Java для начинающих / А.Н. Васильев. - М.: Эксмо, 2014. - 416 с.
2. МакГрат, М. Программирование на Java для начинающих / М. МакГрат. - М.: Эксмо, 2016. - 192 с.
3. МакМанус Ш. Программист: детская академия/ Шон МакМанус.— М.: Эксмо, 2019. —64 с.
4. Федотенко М. А. Разработка мобильных приложений. Первые шаги/М. А. Федотенко; под ред. В.В. Тарапаты. — М.: Лаборатория знаний, 2019. — 335 с.

Примерные задания для промежуточной аттестации

1. Разработайте приложение для вычисления четвертной оценки. Пользователь вводит оценки, полученные за четверть, через пробел. Программа должна вывести средний балл с точностью до 2 знаков после запятой, итоговую оценку и все введенные пользователем оценки.

2. Разработайте приложение – калькулятор сложения, вычитания, умножения и деления смешанных дробей. Пользователь вводит две дроби и номер операции (номера операций определите сами в приветственном слове пользователю) в формате:

целая_часть_первой_дроби	
числитель_дробной_части_первой_дроби	
знаменатель_дробной_части_первой_дроби	целая_часть_второй_дроби
числитель_дробной_части_второй_дроби	
знаменатель_дробной_части_второй_дроби	номер_операции.

Примерные темы для итогового проекта

Во всех задачах необходимо привести полное обоснованное решение.

1. Викторина
2. Интерактивный учебник
3. Инструкция по сбору головоломки
4. Приложение для обмена книгами
5. Математический тренажер