ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ»

УТВЕРЖДАЮ
Ректор РГАИС
А.О. Аракелова
25 мая 2026 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

«РАЗРАБОТКА ПРИЛОЖЕНИЙ ДЛЯ МОБИЛЬНЫХ УСТРОЙСТВ»

Направление подготовки: 09.04.02 «Информационные системы и технологии»
Профиль: «Информационные системы и технологии»
Квалификация (степень) выпускника: магистр
Форма обучения: очная, очно-заочная, заочная

Москва – РГАИС – 2026

Разработчик: доцент кафедры Информационных технологий В.Б. Трухманов «Разработка приложений для мобильных устройств» // Рабочая программа учебной дисциплины предназначена для обучающихся по направлению подготовки 09.04.02 «Информационные системы и технологии». — М.: Российская государственная академия интеллектуальной собственности (РГАИС), кафедра Информационных технологий.

© ФГБОУ ВО РГАИС, 2026

1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

1.1. Цель и задачи дисциплины

Изучение дисциплины «Разработка приложений для мобильных устройств» направлено на получение знаний в области современных средств разработки Web-приложений, получения представления о том, что представляют собой современные средства разработки Web-приложений для мобильных устройств, каковы их возможности, достоинства и недостатки, как их применять в своей профессиональной деятельности. Изучение дисциплины «Разработка приложений для мобильных устройств» нацелено на понимание основных принципов применения средств разработки Web-приложений, того, какие из средств следует использовать для решения конкретных задач и какие ресурсы для этого требуются.

Целью дисциплины «Разработка приложений для мобильных устройств» является формирование у обучающихся теоретических знаний, практических навыков и умений в области разработки и создания Web-приложений, необходимых в дальнейшей профессиональной деятельности.

Для достижения поставленной цели решаются следующие задачи:

– изучить теоретические основы разработки мобильных приложений, что представляет собой современные системы разработки мобильных приложений, каковы особенности и возможности среды разработки мобильных приложений Android Studio;

– рассмотреть теоретические аспекты среды разработки мобильных приложений Android Studio, как создается проект мобильного приложения, как осуществляется проектирование интерфейса мобильных приложений, какова структура проекта, какие основные компоненты используются в разработке приложений;

– изучить возможности среды разработки мобильных приложений Android Studio, что представляет собой макет приложения, как располагать элементы в окне приложения, какие сервисы и в какой последовательности стоит использовать в разрабатываемом мобильном приложении;

– рассмотреть возможности среды разработки мобильных приложений Android Studio в разработке приложений с использованием баз данных;

– уметь использовать проектную деятельность, как основную форму учебного процесса для формирования необходимых знаний, умений и навыков программирования в будущей профессиональной деятельности.

1.2. Место дисциплины в структуре образовательной программы

Дисциплина «Разработка приложений для мобильных устройств» изучается по направлению подготовки 09.04.02 «Информационные системы и технологии» в части, формируемой участниками образовательных отношений, и реализуется на 2 году обучения (4 семестр).

Место дисциплины «Разработка приложений для мобильных устройств» определено, как наиболее значимое, которая опирается на содержание дисциплин: введение в информационные системы и технологии, информационные технологии в профессиональной деятельности, технологии программирования, программирование на языках высокого уровня, Web-программирование, системы Web-дизайна. В свою очередь, содержание дисциплины «Разработка приложений для мобильных устройств» определяет платформу, на которой развертывается проект выпускной квалификационной работы.

По этой причине дисциплина занимает важное место в области профессиональной подготовки.

2. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ) В ЗАЧЕТНЫХ ЕДИНИЦАХ С УКАЗАНИЕМ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ (АСТРОНОМИЧЕСКИХ) ЧАСОВ ПО ВИДАМ УЧЕБНЫХ ЗАНЯТИЙ

Виды занятий	Объем дисциплины		
	Форма обучения		
	Очная форма обучения	Очно-заочная форма обучения	Заочная форма обучения
Объем зачетных единиц	3	3	3
Общая трудоемкость в часах	108	108	108
Аудиторные занятия	24	34	18
Лекции	8	8	8
Практические занятия (семинары)	16	26	10
Самостоятельная работа	84	74	86
Контроль			4
Форма контроля	Зачет	Зачет	Зачет

3. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ), СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ) С УКАЗАНИЕМ КОМПЕТЕНЦИЙ, ФОРМИРУЕМЫХ В ПРОЦЕССЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

3.1. Учебно-тематический план курса и распределение компетенций по темам занятий

Наименование темы	Формируемые компетенции (или их части)	
	ПК-3	ПК-4
Тема 1. Основы разработки мобильных приложений.	+	
Тема 2. Проектирование интерфейса мобильных приложений.	+	+
Тема 3. Дизайн. Создание текстов и изображений. Обработка касаний и жестов.	+	+
Тема 4. Создание приложений с базами данных.	+	+

3.2. Содержание разделов дисциплины (модуля) и контрольные вопросы для самостоятельной работы (самоконтроля) обучающихся

Тема 1. Основы разработки мобильных приложений.

Запуск Android Studio. Создание проекта. Особенности интерфейса Android Studio. Структура проекта Android Studio. Режим дизайна. Текстовые элементы. Библиотеки ресурсов. Идентификатор. Кнопки. Работа в режиме кода. Сборка проекта. Тестирование приложения. Использование эмулятора. USB – отладка.

Контрольные вопросы:

1. Как создать проект в Android Studio?
2. Что представляет собой интерфейс Android Studio?
3. Какова структура проекта Android Studio?
4. Что представляет собой Режим Дизайна Android Studio?
5. Что такое текстовые элементы?
6. Что такое библиотеки ресурсов?
7. Для чего нужен идентификатор?
8. Для чего используются кнопки в Android Studio?

9. Для чего используется режим кода в Android Studio?
10. Как собрать проект в Android Studio?
11. Как тестировать приложение в Android Studio?
12. Как использовать эмулятор и USB для отладки приложения в Android Studio?

Тема 2. Проектирование интерфейса мобильных приложений.

Макеты. Расположение элементов друг над другом. Линейное расположение элементов по вертикали. Линейное расположение элементов по горизонтали. Сеточное расположение элементов. Строковые ресурсы. Ориентация экрана. Особенности разработки приложений для планшетов. Всплывающие сообщения и уведомления. Звуковые эффекты. Примеры создания простейших приложений.

Контрольные вопросы:

1. Что такое макеты в Android Studio?
2. Каким образом можно располагать элементы друг над другом?
3. Как определяется линейное расположение элементов по вертикали?
4. Как определяется линейное расположение элементов по горизонтали?
5. Как осуществляется сеточное расположение элементов?
6. Что представляют собой строковые ресурсы в Android Studio?
7. Как задается ориентация экрана в Android Studio?
8. В чем состоят особенности разработки приложений для планшетов в Android Studio?
9. Как создаются всплывающие сообщения и уведомления в Android Studio?
10. Как создаются звуковые эффекты в Android Studio?

Тема 3. Дизайн. Создание текстов и изображений. Обработка касаний и жестов.

Дизайн и удобство пользования. Логотип приложения. Виды логотипов. Создание и установка логотипа приложения. Загрузочный экран приложения. Анимация элементов и класс AnimationUtils. Индикатор загрузки. Темы и стили. Меню. Фрагменты. Переключение между несколькими экранами. Работа с текстом. Обработка касаний и жестов. Работа с камерой. Разработка учебных проектов приложений.

Контрольные вопросы:

1. Что такое логотип приложения?

2. Какие существуют виды логотипов в Android Studio?
3. Как создаются и устанавливаются логотипы в Android Studio?
4. Что представляет собой загрузочный экран в Android Studio?
5. Что такое анимация элементов и класс AnimationUtils?
6. Что такое индикатор загрузки? Как он работает?
7. Что представляют собой темы и стили в Android Studio?
8. Как создается меню в Android Studio?
9. Как осуществляется переключение между несколькими экранами в Android Studio?
10. Как создаются и вставляются текстовые фрагменты в Android Studio?
11. Как осуществляется обработка касаний и жестов в Android Studio?
12. Как осуществляется работа с камерой в Android Studio?

Тема 4. Создание приложений с базами данных.

Подключение к Интернет. Поддержка БД SQLite. Создание мини-приложения БД. Подключение БД к проекту Android Studio. Инструмент Firebase. Создание приложения с базой данных SQLite.

Контрольные вопросы:

1. Как подключиться к Интернет в Android Studio?

2. Что представляет собой база данных SQLite? Каковы особенности SQLite?

3. Как создать мини-приложение в Android Studio?

4. Что представляет собой инструмент Firebase?

3.3. Активные и интерактивные формы проведения занятий

В качестве активных форм проведения занятий по дисциплине «Разработка приложений для мобильных устройств» предлагаются четыре формы проведения занятий: лекция-беседа, консультационная работа, практическое занятие и проектная деятельность. Выбор интерактивной формы предоставляется непосредственно преподавателю.

Лекция-беседа предполагает непосредственный контакт преподавателя с аудиторией. В начале занятия обучаемые получают материалы лекции в электронном виде.

Во время занятия преподаватель знакомит обучаемых с учебным материалом, акцентируя внимание на разборе примеров приложений. Обучаемые имеют возможность воспроизвести программы в программе

Android Studio на компьютерах. В процессе рассмотрения учебного материала они могут задавать преподавателю уточняющие вопросы. В свою очередь, преподаватель может вносить добавления, расширяющие и углубляющие содержание учебного материала, а также задавать вопросы. Вопросы преподаватель может адресовать как всей аудитории, так и кому-то конкретно. Они могут быть как простые, способные сосредоточить внимание на отдельных важнейших элементах темы, так и проблемные. Обучающиеся, продумывая ответ на заданный вопрос, получают возможность самостоятельно прийти к тем выводам и обобщениям, которые преподаватель должен был сообщить им в качестве новых знаний, либо понять глубину и важность обсуждаемой проблемы, что повышает интерес и степень восприятия материала.

Консультационная работа преподавателя предполагает два вида консультаций: групповые и индивидуальные. Групповые консультации представляют собой своеобразную форму проведения лекционных занятий, основным содержанием которых является разъяснение отдельных, часто наиболее сложных или практически значимых вопросов изучаемой программы. Групповые консультации проводятся в случаях, когда необходимо подробно рассмотреть практические вопросы, недостаточно или совсем не освещенные в лекциях, или при проведении других видов занятий, а также с целью оказания помощи в самостоятельной работе, в подготовке к выполнению практических занятий, в написании рефератов или выпускных работ, сдаче экзаменов и зачетов. Проведение индивидуальных консультаций проводится преподавателем в специально отведенное время. В этом случае к нему за помощью могут обратиться как те, кто испытывает трудности в изучении данной темы, так и обучающиеся, которые хотели бы более глубоко разобраться в содержании изучаемой темы предмета.

Практическое занятие представляет собой разработку мобильных приложений в программе Android Studio. Главная цель практического занятия - закрепление учебного материала, полученных во время лекционных занятий, формирование умений применять полученные знания на практике в будущей профессиональной деятельности.

Проектная деятельность является формой организации учебного процесса, основной задачей которого является разработка учебного программного проекта и самостоятельного доведение его до конечного результата - готового проекта, например, программного андроид-приложения. Главная цель проектной деятельности — это закрепление полученных знаний умений и навыков в области программирования в

процессе самостоятельной разработки мобильного-приложения в соответствии с техническим заданием. В процессе выполнения проекта на занятии возникает атмосфера творчества, повышающая интерес к учебной дисциплине. На определенной стадии выполнения проекта обучающиеся стремятся расширить свои знания о предметной области изучаемой дисциплины либо в виде консультаций с преподавателем, либо самостоятельно. В проектной деятельности допускается и даже приветствуется усложнения исходного технического задания самими обучающимися в сторону создания более совершенного программного приложения.

4. МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

4.1. Методические рекомендации по самостоятельному изучению курса (дисциплины)

Самостоятельная работа обучающихся – это индивидуальная или коллективная учебная деятельность, осуществляемая без непосредственного руководства преподавателя. Самостоятельная работа есть особо организованный вид учебной деятельности, проводимый с целью повышения эффективности подготовки обучающихся к последующим занятиям, формирования у них навыков самостоятельной отработки учебных заданий, а также овладения методикой организации своего самостоятельного труда в целом.

Являясь необходимым элементом дидактической связи различных методов обучения между собой, самостоятельная работа обучающихся призвана обеспечить более глубокое, творческое усвоение понятийного аппарата дисциплины, знаний возможностей и особенностей современных технологий программирования.

Во время лекций обучающимся необходимо сосредоточить внимание на её прослушивание, уловить то главное, что скажет лектор. Основные положения лекции, отдельные важные факты и выводы из рассматриваемых вопросов обучающиеся получают в электронном виде, отдельные положения важные для обучающихся нужно записывать. Записи следует делать кратко.

Главным определяющим фактором успешной работы обучающихся является его самостоятельная работа. В процессе изучения учебных материалов необходимо самостоятельно разобрать теоретический материал, разобрать примеры в указанной среде программирования и выполнить задания для самостоятельной работы.

Успеха в заочном обучении можно добиться только при правильной организации регулярных занятий. Поэтому обучающимся необходимо систематически заниматься.

Организация самостоятельной работы обучающихся должна строиться по системе поэтапного освоения материала. Метод поэтапного изучения включает в себя предварительную подготовку, непосредственное изучение теоретического содержания источника, обобщение полученных знаний.

Предварительная подготовка включает в себя уяснение цели изучения материала, оценку широты информационной базы анализируемого вопроса,

выяснение его научной и практической актуальности. Изучение теоретического содержания заключается в выделении и уяснении ключевых понятий и положений, выявлении их взаимосвязи и систематизации. Обобщение полученных знаний подразумевает широкое осмысление теоретических положений через определение их места в общей структуре изучаемой дисциплины и их значимости для практической деятельности.

Методические рекомендации по проектной деятельности.

Проектная деятельность работа обучающихся является одним из видов учебной деятельности, которая призвана, прежде всего, сформировать навыки разработки программных приложений в соответствии с техническим заданием. Основной целью проектной деятельности дисциплины «Разработка приложений для мобильных устройств» является закрепление полученных знаний умений и навыков в области программирования в процессе самостоятельной разработки программного приложения.

Ключевым моментом проектной деятельности является разработка технического задания. Проектная деятельность осуществляется в рамках практических занятий, а также самостоятельной работы дома. При разработке технического задания следует ориентироваться на содержание теоретического материала учебной дисциплины и практических занятий. Особое внимание следует уделять разработке структурной схеме программного проекта и взаимосвязи объектов и компонентов. От того насколько точно составлено техническое задание зависит успешность всей проектной деятельности.

Проектная деятельность должна быть построена таким образом, чтобы обучающиеся имели возможность не только довести проект до готового программного приложения, но и усложнить техническое задание в сторону создания более совершенного программного приложения.

Методические рекомендации по работе с литературой.

При самостоятельном изучении основной рекомендованной литературы обучающимся необходимо обратить главное внимание на узловые положения, излагаемые в изучаемом тексте.

Необходимо внимательно ознакомиться с содержанием соответствующего блока информации, структурировать его и выделить в нем центральное звено. Обычно это бывает ключевое определение или совокупность сущностных характеристик рассматриваемого объекта. Для того, чтобы убедиться, насколько глубоко усвоено содержание темы, в конце соответствующих глав и параграфов учебных пособий обычно дается перечень контрольных вопросов, на которые обучающийся должен уметь дать четкие и конкретные ответы.

Работа с дополнительной литературой предполагает умение выделять в ней необходимый аспект изучаемой темы. Дополнительную литературу целесообразно прорабатывать на базе уже освоенной основной литературы, изучать комплексно и всесторонне.

Обязательный элемент самостоятельной работы обучающихся с источниками и литературой – ведение необходимых записей. Основными общепринятыми формами записей являются конспект, выписки, тезисы, аннотации, резюме, план.

Конспект – это краткое письменное изложение содержания источника, статьи, доклада, лекции, включающее в сжатой форме основные положения и их обоснование.

Выписки – это краткие записи в форме цитат (дословное воспроизведение отрывков источника, произведения, статьи, содержащих существенные положения, мысли автора), либо лаконичное, близкое к тексту изложение основного содержания.

Тезисы – это сжатое изложение ключевых идей прочитанного источника или произведения.

Аннотации, резюме – это соответственно предельно краткое обобщающее изложение содержания текста, критическая оценка прочитанного документа или произведения.

В целях структурирования содержания изучаемой работы целесообразно составлять ее план, который должен раскрывать логику построения текста, а также способствовать лучшей ориентации обучающегося в содержании источника.

Самостоятельная работа обучающегося будет эффективной и полезной в том случае, если она будет построена исходя из понимания обучающимися необходимости обеспечения максимально широкого охвата информационных источников, что вполне достижимо при научной организации учебного труда.

4.2. Глоссарий

Android Studio — официальная среда разработки мобильных приложений для устройств с операционной системой Android от компании Google.

ActionBar (*панель действий*) — «шапка» приложения. В ней отображается имя приложения и кнопки вызова меню.

Activity (*активная область экрана*) — каждый отдельно взятый экран приложения.

Alert Dialog (*диалоговое окно*) — один из способов оповещения пользователей — всплывающее окно, в котором запрашивается у пользователя ответ на некоторый вопрос. Обычно содержит вопрос и кнопки с предлагаемыми вариантами ответа.

Android SDK (Software Development Kit) — набор инструментов для разработки программного обеспечения для операционной системы Android. Включает в себя эмуляторы устройств, документацию и все необходимые для разработки пакеты.

Android Studio проект — все файлы приложения, необходимые для его разработки. Хранятся в служебных папках, имеющих сложную структуру, и распознаются Android Studio как единый проект.

AndroidManifest.xml (файл манифеста) — файл проекта, в котором на языке разметки XML описана основная информация о приложении, необходимая для его распознавания операционной системой смартфона или магазином приложений (название приложения, иконка, разрешения, основные компоненты приложения).

AnimationUtils (*средства анимации*) — Java-класс, в котором описаны все методы реализации анимации элементов интерфейса.

AVD Manager (Android Virtual Device Manager) — менеджер виртуальных Android-устройств.

Bottom Navigation Activity (*экран с нижней навигацией*) — шаблон интерфейса, в котором переключение между фрагмента- ми осуществляется с помощью нижней панели навигации.

Design и Text (*дизайн и код*) — два режима работы с файлами xml-разметки; вкладки для переключения между этими режимами.

Firebase — мобильная платформа, которая при интеграции с приложением дает разработчику возможность хранить данные в базе данных, синхронизируемой в реальном времени, подключать приложение к системам аналитики, организовывать обмен сообщениями между пользователями, авторизацию в приложении, рассылку уведомлений и многие другие функции.

float — тип данных Java для работы с дробными числами (числами с плавающей точкой).

FloatingActionButton (*плавающая кнопка действия*) — элемент интерфейса, плавающая кнопка в правом нижнем углу экрана, предназначенная для того, чтобы самые важные функции приложения всегда находились «под рукой» у пользователя.

Fragment (*фрагмент*) — Java-класс, в котором описаны основные методы работы с фрагментами (элементами-контейнерами, которые можно переключать внутри одной Activity).

FragmentManager (менеджер фрагментов) — Java-класс, в котором прописаны методы управления и переключения фрагментов.

FrameLayout (*рамочный макет*) — элемент интерфейса (макет), внутри которого элементы располагаются поверх других элементов.

GestureDetector (*детектор жестов*) — Java-класс, в котором описаны основные методы обработки касаний и жестов.

Gradle — система автоматической сборки. Android Studio выполняет сборку приложения с помощью системы Gradle в фоновом режиме, выполняя Gradle-скрипты без необходимости нашего вмешательства как разработчиков.

Gradle Scripts (*скрипты Gradle*) — папка проекта, в которой хранятся написанные на языке Groovy скрипты системы автоматической сборки Gradle.

GridLayout (*макет-сетка*) — элемент интерфейса (макет), внутри которого элементы располагаются в прямоугольной сетке, похожей на таблицу.

Handler (*обработчик*) — Java-класс, в котором прописаны основные методы для создания и управления фоновыми потоками, выполняющимися в любое заданное время.

ID (*идентификатор*) — уникальный идентификатор элемента для последующего обращения к этому элементу. Идентификатор должен напоминать о типе и содержании элемента.

Image Asset Studio — встроенный инструмент Android Studio, который предназначен для простого и быстрого создания различных значков: для логотипа приложения, для пунктов меню, кнопок и других.

ImageButton (*изображение-кнопка*) — элемент интерфейса, представляющий собой картинку, на которую можно нажать, как на кнопку.

ImageView (*элемент-изображение*) — элемент интерфейса, предназначенный для отображения картинки.

int — тип данных Java для работы с целыми числами.

isChecked() (*поставлена ли галочка*) — метод Java, вызываемый для того, чтобы определить, поставлена ли выбранная галочка (чекбокс) или выбран ли пункт меню.

Java — объектно-ориентированный язык программирования. На нем пишется, чтобы интерфейс приложения стал «живым».

Landscape variation (*горизонтальный вариант*) — копия Activity, создаваемая для горизонтальной ориентации экрана.

Layout (*макет*) — элемент интерфейса для компоновки элементов внутри него.

LinearLayout (horizontal) (*горизонтальный линейный макет*) — элемент интерфейса (макет), внутри которого элементы располагаются друг за другом по горизонтали.

LinearLayout (vertical) (*вертикальный линейный макет*) — элемент интерфейса (макет), внутри которого элементы располагаются друг за другом по вертикали.

ListView (*элемент-список*) — элемент интерфейса, предназначенный для хранения и отображения списков.

Login Activity (*экран авторизации*) — шаблон интерфейса для создания страницы авторизации в приложении.

MainActivity (*главный экран*) — Java-класс, в котором описано все, что относится к главному экрану приложения.

makeText() (*«сделать», то есть задать текст*) — метод Java, который вызывается, чтобы заполнить всплывающее сообщение текстом.

manifests (*манифесты*) — папка проекта, в которой хранятся файлы манифеста.

onClick() (*при нажатии*) — метод Java, в котором описывается обработка нажатия на кнопку (или любой другой элемент интерфейса).

onCreate() (*при создании*) — метод Java, в котором описывается все, что должно происходить при переходе к данному экрану.

onScroll() (*при прокручивании*) — метод Java, в котором прописываются методы обработки жеста «скроллинг» (прокрутка).

Palette (*палитра элементов*) — панель Android Studio. В ее левой части перечислены названия типов элементов, а в правой — все элементы выбранного типа.

Portrait Variation (*вертикальный вариант*) — копия Activity, создаваемая для вертикальной ориентации экрана.

Resources (*ресурсы*) — библиотека ресурсов. Если определить и прописать изображение или какой-нибудь текст как ресурс, в дальнейшем к нему можно будет неоднократно обращаться, указав его уникальный идентификатор id, а не переписывая текст каждый раз заново.

ScrollingActivity (*прокручиваемый экран*) — шаблон интерфейса для создания экрана с длинным прокручиваемым текстом.

setVisibility() (*установить видимость*) — метод Java, который вызывается для изменения типа видимости элемента (видимый, невидимый, скрытый).

show() (*показать*) — метод Java, который вызывается для отображения уведомления после установки всех его параметров.

SoundPool (*звуковой пул, объединение*) — Java-класс, в котором описаны основные методы для реализации возможности загрузки, воспроизведения и работы со звуками в приложении.

Splash Screen (*заставка*) — загрузочный экран приложения, то есть экран-заставка, который демонстрируется пользователю во время загрузки приложения.

String — тип данных языка Java для работы со строками (текстом).

Switch (*переключатель*) — условный оператор языка Java, который в отличие от if и if-else применяется для известного списка значений, а также предусматривает ситуацию по умолчанию (default).

Switch (*переключатель*) — элемент интерфейса, который име-ет два режима — включен и выключен. Служит для переключения между этими параметрами.

TextView (*текстовый элемент*) — элемент интерфейса для отображения на экране текста.

toString() (*в строку*) — метод Java, который вызывается для «конвертации» извлеченного текста в строку.

WebView (*элемент для отображения веб-содержимого*) — элемент интерфейса, с помощью которого можно встраивать в мобильные приложения веб-страницы или их части.

XML — расширяемый язык разметки, созданный для описания данных. В Android Studio на языке XML описываются элементы интерфейса (дизайн) приложения, а также некоторые ресурсы (цвета, строки, стили).

Главное меню — меню, которое вызывается обычно из правого верхнего угла ActionBar. В него обычно выносят такие пункты, как «Настройки», «О приложении», «Оценить приложение» и так далее, то есть моменты, которые несут в себе информацию о приложении, но напрямую не относятся к его основному содержанию.

Исключение — возникшая во время выполнения программы нештатная ситуация (ошибка).

Контекстное меню — меню, которое может вызываться по нажатию на любой элемент приложения, как правило, долгим нажатием. Представляет собой аналог контекстного меню, которое мы вызываем нажатием правой кнопкой мыши при работе за компьютером.

Меню-шторка — основное меню приложения. Оно разворачивается на большую часть экрана и вызывается либо свайпом от края экрана, либо нажатием на кнопку с тремя полосками, которая обычно располагается в левом верхнем углу в ActionBar приложения.

Стиль (в разработке мобильных приложений) — набор настроек отдельно взятого элемента приложения, которые можно отнести к его дизайну (цвет, шрифт, цвет фона и так далее).

Тема (в разработке мобильных приложений) — стиль, который применяется ко всему приложению или, как минимум, ко всей Activity.

Эмулятор — виртуальное устройство, имитирующее работу смартфона, на котором можно запустить и протестировать наши приложения.

5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

Оценка качества освоения обучающимися образовательных программ включает в себя порядок, периодичность, систему оценок и формы проведения текущего контроля успеваемости и промежуточной аттестации обучающихся.

Нормативно-методическое обеспечение текущего контроля успеваемости и промежуточной аттестации обучающихся осуществляется в соответствии с положением ФГБОУ ВО РГАИС «Об осуществлении текущего контроля успеваемости и промежуточной аттестации обучающихся».

Основными задачами текущего контроля успеваемости является систематический мониторинг за формированием компетенций, предусмотренных ФГОС ВО и ООП, повышение качества знаний обучающихся, приобретение и развитие навыков самостоятельной работы, повышение академической активности обучающихся.

Критерии оценки обучающихся

Текущая аттестация (текущий контроль) уровня усвоения содержания дисциплины возможно проводить в ходе всех видов учебных занятий методами устного и письменного опроса (работ), в процессе выступлений обучающихся на практических занятиях, защиты рефератов, а также посредством тестирования.

Качество письменных работ оценивается исходя из того, что обучающиеся:

– выбрали и использовали форму и стиль изложения, соответствующие целям и содержанию дисциплины;

– применили связанную с темой информацию, используя при этом понятийный аппарат специалиста в данной области;

– представили структурированный и грамотно написанный текст, имеющий связное содержание.

Тестовые материалы оцениваются по процентному соотношению правильных вариантов. Количество правильных ответов в пределах от 90 до 100 % - «отлично»; в пределах от 75 до 89 % - «хорошо»; в пределах от 50 до 74 % - «удовлетворительно»; менее 50 % - «неудовлетворительно».

Сдача зачета происходит в устной форме по билетам. В ходе зачета студент должен продемонстрировать знания и умения по предмету учебного

курса. Качество ответов студентов и выполнение заданий оценивается: «зачтено», «зачтено с оценкой» и/или «не зачтено», «не зачтено с оценкой».

«зачтено», «зачтено с оценкой»:

– полные, осознанные знания в рамках курса лекций и дополнительной литературы, логичное и грамотное изложение материала.

«не зачтено» «не зачтено с оценкой»:

– допускаются существенные ошибки в знании курса лекций, при ответе вскрывается ошибочное понимание основных понятий курса.

Сдача экзамена происходит в устной форме по билетам.

Качество ответов на экзамене оцениваются на «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно».

Оценка «отлично» выставляется обучающемуся, если:

– даны исчерпывающие и обоснованные ответы на все поставленные вопросы, правильно решены практические задачи;

– ответы были четкими и краткими, основные мысли излагались в строгой логической последовательности;

– обучающийся продемонстрировал умение самостоятельно анализировать факты, события, явления, процессы в их взаимосвязи и диалектическом развитии.

Оценка «хорошо» выставляется обучающемуся, если:

– даны полные, достаточно обоснованные ответы на поставленные вопросы, правильно решены практические задания;

– в ответах не всегда выделялось главное, при решении практических задач не всегда использовались рациональные методики расчётов;

– ответы в основном были краткими, но не всегда четкими.

Оценка «удовлетворительно» выставляется обучающемуся, если:

– даны в основном правильные ответы на все поставленные вопросы, но без должной глубины и обоснования, при решении практических задач студент использовал прежний опыт и не применял новые методики выполнения расчётов, однако на уточняющие вопросы даны в целом правильные ответы;

– при ответах не выделялось главное;

– ответы были многословными, нечеткими и без должной логической последовательности;

– на отдельные дополнительные вопросы не даны положительные ответы.

Оценка «неудовлетворительно» выставляется обучающемуся, если не выполнены требования, соответствующие оценке «удовлетворительно».

5.1. Список вопросов к зачету

1. Какова структура проекта Android Studio?
2. Что представляет собой Режим Дизайна Android Studio? Что такое текстовые элементы?
3. Что такое библиотеки ресурсов в Android Studio? Какие ресурсы используются наиболее часто в проектах?
4. Для чего используется режим кода в Android Studio?
5. Как собрать проект в Android Studio? Как тестировать приложение?
6. Как использовать эмулятор и USB для отладки приложения в Android Studio?
7. Что такое макеты в Android Studio? Каким образом можно располагать элементы друг над другом?
8. Как определяется линейное расположение элементов по вертикали и горизонтали? Как осуществляется сеточное расположение элементов?
9. Что представляют собой строковые ресурсы в Android Studio? Как задается ориентация экрана в Android Studio?
10. Как создаются всплывающие сообщения и уведомления в Android Studio? Как создаются звуковые эффекты в Android Studio?
11. Что такое макеты в Android Studio?
12. Каким образом можно располагать элементы друг над другом?
13. Как определяется линейное расположение элементов по вертикали?
14. Как определяется линейное расположение элементов по горизонтали?
15. Как осуществляется сеточное расположение элементов?
16. Что представляют собой строковые ресурсы в Android Studio?
17. Как задается ориентация экрана в Android Studio?
18. В чем состоят особенности разработки приложений для планшетов в Android Studio?
19. Как создаются всплывающие сообщения и уведомления в Android Studio?
20. Как создаются звуковые эффекты в Android Studio?
21. Что такое логотип приложения? Как создаются и устанавливаются логотипы в Android Studio?
22. Что представляет собой загрузочный экран в Android Studio?

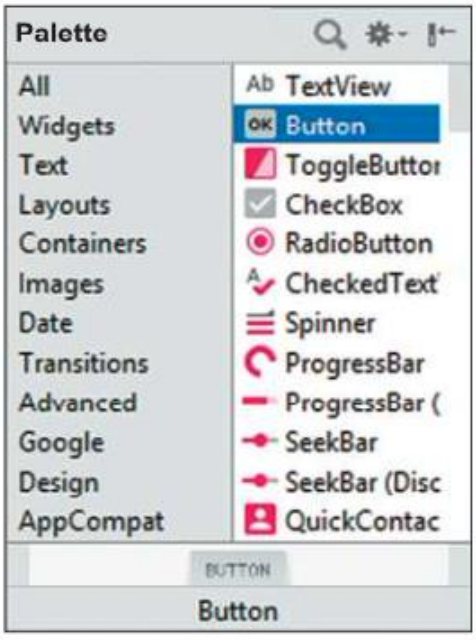
23. Что такое анимация элементов и класс AnimationUtils?
24. Как создается меню в Android Studio?
25. Как осуществляется переключение между несколькими экранами в Android Studio? Как создаются и вставляются текстовые фрагменты в Android Studio?
26. Как осуществляется обработка касаний и жестов в Android Studio?
27. Как осуществляется работа с камерой в Android Studio?
28. Как подключиться к Интернет в Android Studio?
29. Что представляет собой база данных SQLite? Каковы особенности SQLite?
30. Что представляет собой инструмент Firebase?

5.2. Тестовые задания

1. Android Studio – это

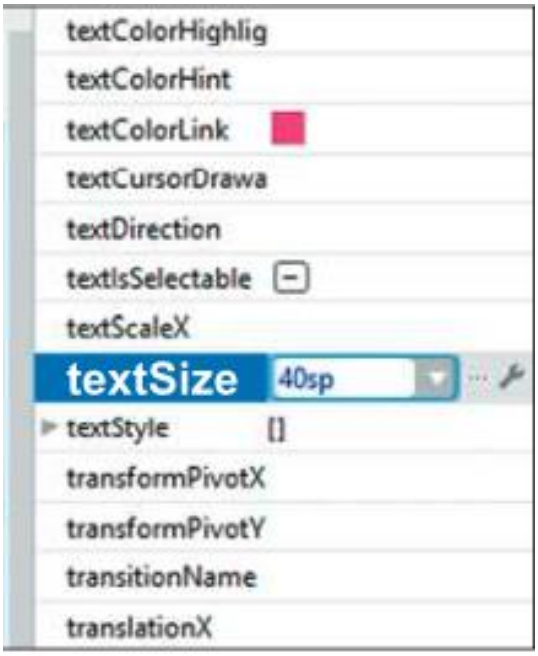
a) официальная среда разработки мобильных приложений для устройств с операционной системой Android от компании Google;
b) бесплатно распространяемая программа для создания Web-приложений;
c) среда разработки web-приложений компании Microsoft.

2. Панель Palette в Android Studio представляет собой:



a) дерево объектов;
b) панель свойств;
c) палитру элементов.

3. Панель Palette в Android Studio представляет собой:



a) дерево объектов;
b) панель свойств;
c) палитру элементов.

4. Элемент TextView предназначен для:

a) отображения текста;
b) отображения рисунка;
c) подключения к базе данных.

5. Свойство visibility элементов отвечает за:

a) видимость – невидимость на экране;
b) порядок слоев наложения;
c) яркость по отношению к фону.

6. Элемент ImageButton – это:

a) рисунок;
b) меню;
c) изображение-кнопка.

7. Файл манифеста содержит:

a) базу данных проекта;
b) основную информацию о проекте ;
c) систему управления проектом.

8. Эмулятор смартфона – это:

a) специальная программа подключения смартфона к компьютеру через разъем USB;
b) виртуальное устройство, имитирующее работу смартфона, на котором будет запускаться приложение;
c) приложение передачи данных с компьютера на смартфон.

9. USB- отладка – это:

a) отладка программы через подключение смартфона к компьютеру с помощью кабеля USB;
b) имитация подключения к кабелю USBс помощью специальной опции отладчика;
c) приложение имитации передачи данных с компьютера на смартфон.

10. Стиль – это:

a) почерк написания кода программы;
b) набор настроек отдельно взятого элемента приложения, которые можно отнести к его дизайну;
c) способ расположения текста и кода программы в одном окне.

11. Исключение – это:

a) повторное использование свойства компонента в одной команде;
b) удаление переменной в ходе выполнения программы;
c) возникшая в ходе выполнения программы нештатная ситуация.

12. FrameLayout – это:

a) добавление элемента в палитру элементов;
b) исключение элемента из палитры элементов;
c) расположение элементов друг за другом (в стопку) в дереве компонентов.

13. LinearLayout (vertical) – это:

a) линейное расположение элементов по вертикали;
b) линейное расположение элементов по горизонтали;
c) изменение размера элемента относительно центра экрана.

14. LinearLayout (horizontal) – это:

a) линейное расположение элементов по вертикали;
b) линейное расположение элементов по горизонтали;

c) изменение размера элемента относительно центра экрана.

15. GridLayout – это:

a) табличный процессор;
b) расположение элементов в прямоугольной сетке;
c) массив.

16. Toast Notification – это:

a) изменение содержимого буфера обмена данными;
b) тестирование работоспособности приложения;
c) всплывающее сообщение.

17. Диалоговое окно используется для:

a) окно коллективного обсуждения проекта при совместной работе над проектом;
b) дополнительное рабочее пространство пользователя для работы над кодом программы;
c) ввода дополнительной информации или запроса на принятие некоторого решения пользователя.

18. Уведомления – это:

a) разновидность программного кода, содержащая явные ошибки;
b) сообщения, которые выводятся за пределами самого приложения для привлечения внимания пользователя;
c) реакция на изменение свойства элемента.

19. Загрузочный экран приложения – это:

a) экран-заставка, которая демонстрируется пользователю во время загрузки приложения;
b) папка, в которой расположены наборы логотипов приложения;
c) область памяти компьютера в которую выгружается приложение в момент проверки на вирусы.

20. Тема – это:

a) диалоговое окно для дискуссии разработчиков в коллективном проекте;
b) предметная область проекта;
c) стиль, который применяется ко всему приложению.

21. В Android Studio для работы с базами данных используется:

a) SQLite;
b) Microsoft Project;
c) Microsof SQL Server.

22. Sring – это:

a) переключатель между опциями в диалоговом окне кода программы;
b) условный оператор языка Java, который в отличие от if и if-else применяется для известного списка значений, а также предусматривает ситуацию по умолчанию;
c) тип данных языка Java для работы со строками.

23. Switch – это:

a) переключатель между опциями в диалоговом окне кода программы;
b) условный оператор языка Java, который в отличие от if и if-else применяется для известного списка значений, а также предусматривает ситуацию по умолчанию;
c) тип данных языка Java для работы со строками.

24. ToString() – это:

a) метод Java предназначенный для конвертации строки в текст;
b) метод Java предназначенный для приведения строк к единому стилю;
c) метод Java предназначенный для конвертации текста в строку.

25. Show – это

a) метод Java предназначенный для отображения пути загрузки приложения
b) метод Java предназначенный для отображения файла подкачки при работе с объемным приложением
c) метод Java предназначенный для отображения уведомления

26. Resources – это

a) библиотека ресурсов
b) область описания переменных
c) список расширений приложения

27. Int – это:

a) тип данных Java для работы с вещественными числами;
b) тип данных Java для работы со смешанными числами;

c) тип данных Java для работы с целыми числами.

28. Fragment – это:

a) часть кода программы предназначенного для немедленного выполнения;
b) часть кода программы предназначенного для удаления;
c) Java-класс, в котором описаны основные методы работы с фрагментами.

29. Float – это:

a) тип данных Java для работы с вещественными числами;
b) тип данных Java для работы со смешанными числами;
c) тип данных Java для работы с целыми числами.

30. ID – это:

a) уникальный идентификатор элемента для последующего обращения к этому элементу;
b) тип данных Java для работы со смешанными числами;
c) тип данных Java для работы с целыми числами.

Ключ
к демоверсии теста по дисциплине
«Разработка приложений для мобильных устройств»

1	2	3	4	5
a	c	b	a	a
6	7	8	9	10
c	b	b	a	b
11	12	13	14	15
c	c	a	b	b
16	17	18	19	20
c	c	b	a	c
21	22	23	24	25
a	c	b	c	c
26	27	28	29	30
a	c	c	a	a

6. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

6.1. Основная и дополнительная учебная литература

Основная литература

1. Пирская, Л. В. Разработка мобильных приложений в среде Android Studio : учебное пособие : [16+] / Л. В. Пирская ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2019. – 125 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=598634. – Библиогр. в кн. – ISBN 978-5-9275-3346-6. – Текст : электронный.

2. Черников В. Н. Разработка мобильных приложений на C# для iOS и Android / В.Н. Черников. - Москва : ДМК Пресс, 2020. - 188 с. - ISBN 978-5-97060-805-0. - URL: https://ibooks.ru/bookshelf/387368/reading . - Текст: электронный.

Дополнительная литература

1. Кузнецова, М. М. Разработка мобильного приложения для медитации на платформе iOS / М. М. Кузнецова ; Санкт-Петербургский государственный университет промышленных технологий и дизайна, Институт полиграфических технологий и оборудования, Кафедра информационных и управляющих систем. – Санкт-Петербург : б.и., 2019. – 73 с. : табл., ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=594424 . – Текст : электронный.

2. Тракимус, Ю. В. Разработка консольных приложений с помощью Microsoft Visual Studio 2017 : учебное пособие : [16+] / Ю. В. Тракимус ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2018. – 64 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=575290 . – Библиогр. в кн. – ISBN 978-5-7782-3763-6. – Текст : электронный.

Библиотечный фонд Академии укомплектован печатной или электронной основной учебной литературой по дисциплинам базовой части всех циклов, изданными за последние 5 лет.

Фонд дополнительной литературы включает в себя официальные справочно-библиографические и периодические издания в расчете не менее одного экземпляра на каждые 100 обучающихся. Каждому обучающемуся обеспечен доступ к комплектам библиотечного фонда и периодическое издание из следующего перечня: Копирайт; wipo magazine; Библиотековедение; Биржа интеллектуальной собственности (БИС); Бюллетень Министерства юстиции Российской Федерации; Вестник гражданского права; Государство и право; Инновации; Интеллектуальная собственность. Авторское право и смежные права; Интеллектуальная собственность. Промышленная собственность; Международное публичное и частное право; Общество: социология, психология, педагогика; Патентный поверенный; Патенты и лицензии. Интеллектуальные права; Уголовное право; Управление проектами и программами; Хозяйство право; Экономическая политика.

7. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, ИНФОРМАЦИОННО-СПРАВОЧНЫХ СИСТЕМ И РЕСУРСОВ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

В процессе реализации образовательной программы в вузе применяются современные интерактивные и мультимедийные средства обучения (компьютеры, мультимедиапроекторы, интерактивные доски и др.), тематические стенды и плакаты, а также электронные информационные образовательные ресурсы.

На основе аппаратно-программного комплекса в РГАИС функционирует и постоянно совершенствуется портал электронного обучения и дистанционных образовательных технологий (ЭОиДОТ), обеспечиваемый преимущественно авторским учебным контентом и методическими разработками профессорско-преподавательского состава Академии.

В РГАИС функционируют читальный зал и электронная библиотека. Сотрудникам и обучающимся обеспечен доступ к электронной библиотечной системе «Университетская библиотека онлайн», насчитывающей более 100 тысяч наименований изданий с доступом в режиме онлайн, а также к объектам Национальной электронной библиотеки (в соответствии с договором с ФГБУ «Российская государственная библиотека»).

Имеется компьютерный класс, возможности которого позволяют каждому из обучающихся работать на компьютере с установленным комплектом лицензионного программного обеспечения не менее 20 часов в год. Академия обеспечена необходимым комплектом лицензионного программного обеспечения

Электронная информационно-образовательная среда Академии обеспечивает:

- доступ к учебным планам, рабочим программам дисциплин (модулей), практик, к изданиям электронных библиотечных систем и электронным образовательным ресурсам, указанным в рабочих программах;

- фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения программы;

- формирование электронного портфолио обучающегося, в том числе сохранение его работ и оценок за эти работы.

- доступ к современным профессиональным базам данных, информационным справочным и поисковым системам, в том числе: справочно-правовой системе «Гарант»: www.garant.ru; справочно-правовой

системе «Консультант плюс»: www.consultant.ru; библиотеке «Книгофонд»: www.knigafund.ru; Университетской библиотеке www.biblioclub.ru.

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Для ведения образовательной деятельности по данной дисциплине Академия располагает материально-технической базой, обеспечивающей проведение всех видов лабораторной, практической и научно-исследовательской работы обучающихся, предусмотренных учебным планом РГАИС, и соответствующей действующим санитарным и противопожарным правилам и нормам.

Для организации и ведения учебного процесса Академия располагает зданием общей площадью 5936,2 кв.м, учебная и учебно-лабораторная площадь составляет 1249,6 кв.м. Для питания сотрудников и обучающихся имеется столовая площадью 130,1 кв.м.

Аудиторные занятия проводятся в специальных помещениях, представляющих собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также в помещениях для самостоятельной работы. Имеются помещения для хранения и профилактического обслуживания учебного оборудования. Специальные помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа имеются наборы демонстрационного оборудования и учебно-наглядных пособий, обеспечивающие тематические иллюстрации, соответствующие примерным программам дисциплин (модулей), рабочим учебным программам дисциплин (модулей).

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

9. ОСОБЕННОСТИ ОБУЧЕНИЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Организация образовательного процесса для лиц с ограниченными возможностями здоровья осуществляется в соответствии с приказом Минобрнауки России от 9 июня 2016 г. № 694 «О внесении изменений в административные регламенты предоставления государственных услуг в части обеспечения условий доступности государственных услуг для инвалидов», «Методическими рекомендациями по организации образовательного процесса для инвалидов и лиц с ограниченными возможностями здоровья в образовательных организациях высшего образования, в том числе оснащенности образовательного процесса» Министерства образования и науки РФ от 08.04.2014 г. № АК-44/05вн.

Академия предоставляет инвалидам и лицам с ограниченными возможностями здоровья (по их заявлению) возможность обучения по образовательной программе, учитывающей особенности их психофизического развития, индивидуальных возможностей и при необходимости, обеспечивающей коррекцию нарушений развития и социальную адаптацию указанных лиц. Для инвалидов и лиц с ограниченными возможностями здоровья Академия устанавливает особый порядок освоения дисциплин (модулей).

Подбор и разработка учебных материалов для обучающихся с ограниченными возможностями здоровья производится с учетом их индивидуальных особенностей.

Предусмотрена возможность обучения по индивидуальному графику.