Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Ганеев Винер Валиахметович
Должность: Директор
Дата подписания: 02.11.2026 14:16:09
Уникальный программный ключ:
fceab25d7092f3bff743e8ad3f8d57fddc1f5e66

Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Уфимский университет науки и технологий»

Бирский филиал

ФАКУЛЬТЕТ ФИЗИКИ И МАТЕМАТИКИ

УТВЕРЖДАЮ:
Декан
подписано ЭЦП Гайсин Ф.Р.
(подпись, инициалы, фамилия)

«26» 02 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Разработка приложений для мобильных устройств
(наименование дисциплины)

ОПОП ВО программа бакалавриата
09.03.03 Прикладная информатика
шифр и наименование направления подготовки (специальности)

направленность (профиль, специализация)

Прикладная информатика в информационной сфере
наименование направленности (профиля, специализации)

форма обучения
очная
(очная, очно-заочная, заочная)

Бирск 2026 г.

Рабочая программа составлена на основании учебного плана основной профессиональной образовательной программы 09.03.03 Прикладная информатика профиль Прикладная информатика в информационной сфере, одобренного ученым советом Бирского филиала Уфимского университета науки и технологий (протокол №7 от 27.02.2026 г.) и утвержденного директором Бирского филиала 27.02.2026.

Зав.кафедрой кафедры информатики и экономики *(наименование кафедры разработчика программы)*	*подписано ЭЦП*	Стовба Е.В.
Разработчик программы	*подписано ЭЦП*	Гилев А.Ю.
Руководитель образовательной программы	*подписано ЭЦП*	Бигаева Л.А.

1. Цель дисциплины. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

1.1. Цель дисциплины

Цель изучения дисциплины: формирование знаний, умений и навыков в области создания и адаптирования программного обеспечения мобильных устройств.

1.2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

Таблица 1. – Результаты обучения по дисциплине

Планируемые результаты освоения основной профессиональной образовательной программы (компетенции, закрепленные за дисциплиной)		Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной
код компетенции	наименование компетенции	
ПК-2	Способен разрабатывать и адаптировать программное обеспечение	ПК-2.1. Демонстрирует знания принципов, методов и средств разработки и адаптирования программного обеспечения на различных программных платформах
		ПК-2.2. Разрабатывает программное обеспечение на различных программных платформах
		ПК-2.3. Применяет методы тестирования и отладки, оценивает качество разработанного или адаптированного ПО по заданным критериям.

2. Структура и трудоемкость дисциплины

Общая трудоемкость (объем) дисциплины составляет 5 зачетные единицы (з.е.), 180 академических часов.

Таблица 2 – Объем дисциплины

Виды учебной работы	Всего, часов	Количество часов в семестре
Общая трудоемкость дисциплины	180	8 семестр - 180
Контактная работа обучающихся с преподавателем по видам учебных занятий (всего)	94	8 семестр - 94
в том числе:		
лекции	36	8 семестр - 36
лабораторные занятия	58	8 семестр - 58
практические занятия	0	
Другие виды работ в соответствии с УП:		
контрольная работа	0	
консультации	1	8 семестр - 1

Виды учебной работы	Всего, часов	Количество часов в семестре
Самостоятельная работа обучающихся (всего)	86	8 семестр - 86
Из них:		
контроль	34.8	
ФКР:		
зачет	0	
зачет с оценкой	0	
курсовая работа (проект)	0	
экзамен	1.2	8 семестр - 1

3 Содержание дисциплины

Таблица 3 – Содержание дисциплины

№ п/п	Раздел (тема) дисциплины	Виды деятельности				Форма текущего контроля успеваемости
		Лек, час.	Лаб, час.	Эк, час.	СРС, час.	
4 курс / 8 семестр						
1	Язык программирования Kotlin Синтаксис и основные алгоритмические конструкции Java. Модификаторы доступа в Java. Классы, наследование классов Java, полиморфизм. Интерфейсы Java. Абстрактные классы в Java. Коллекции Java. Java пакеты (Package), взаимодействие между пакетами. Java модули (Module), взаимодействие между модулями.	8	10		8	Лабораторная работа, Индивидуальный опрос
2	Архитектура Android и инструменты разработчика Архитектура Android и инструменты разработчика	8	8		8	Лабораторная работа
3	Программирование пользовательских интерфейсов Программирование пользовательских интерфейсов	4	8		8	Индивидуальный опрос, Лабораторная работа
4	Процессы и потоки Process и Main Thread. Создание и жизненный цикл Thread. Синхронизация. Coroutines.	4	8		6	Индивидуальный опрос, Лабораторная работа
5	Архитектурные компоненты Жизненный цикл Activity/Fragment. Жизненный цикл JetPack. Что такое LiveData? Концепция ViewModel.	4	8		7	Индивидуальный опрос, Лабораторная работа
6	Работа с сетью. Сохранение данных	4	8		7	Лабораторная

	Работа с сетью. Сохранение данных					работа, Индивидуальный опрос
7	Фоновая работа. Уведомления и оповещения Фоновая работа. Уведомления и оповещения	4	8		6	Лабораторная работа, Индивидуальный опрос
8	Экзамен			1	36	
Итого по 4 курсу 8 семестру		36	58	1	86	
Итого по дисциплине		36	58	1	86	

Таблица 4 – Лабораторные работы

№ п/п	Наименование лабораторных работ	Объем, час.

4. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Критериями оценивания являются баллы, которые выставляются за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины. Баллы, выставляемые за конкретные виды деятельности представлены ниже.

Контрольные задания для проведения текущего контроля успеваемости

Индивидуальный опрос

Индивидуальный опрос

Индивидуальный опрос 1

1. Синтаксис и основные алгоритмические конструкции Java;
2. Модификаторы доступа в Java;
3. Классы, наследование классов Java, полиморфизм.
4. Интерфейсы Java;
5. Абстрактные классы в Java;
6. Коллекции Java;
7. Java пакеты (Package), взаимодействие между пакетами;
8. Java модули (Module), взаимодействие между модулями;

Индивидуальный опрос 2

1. Что такое Android SDK? Какие компоненты он содержит? Какие инструментальные средства можно использовать при разработке приложений на платформе Android?
2. Что такое менеджер пакетов Android? Какие задачи он решает?
3. Какова структура автоматически создаваемого проекта приложения для Android? Какие компоненты создаются и в каких каталогах они размещаются?
4. Что такое файл манифеста? Какова его структура? Какие основные элементы могут встречаться в файле манифеста и для чего они нужны?
5. Что такое ant? Как он используется для сборки приложений? Какие цели содержатся в автоматически сгенерированном файле сборки?

Индивидуальный опрос 2_2

1. Из каких компонентов могут состоять Android-приложения? В чём назначение и характерные особенности каждого из компонентов?
2. Что такое интент? Какую роль играют интенты во взаимодействии компонентов на платформе Android?
3. Как объявить активность в файле манифеста? С какой целью необходимо данное объявление?
4. Что такое жизненный цикл активности? Какие особенности платформы заставляют вводить понятие жизненного цикла?
5. Какие callback-методы жизненного цикла активности могут быть переопределены разработчиком? В какие моменты жизненного цикла они будут вызываться? Каково типичное назначение каждого из этих callback-методов?
6. Назовите два способа вызова активности через интент. В чём заключается различие между ними? Когда используется каждый из этих способов?
7. Что такое «задача» в терминах Android? Как задачи связаны с активностями? Как задачи выглядят с точки зрения пользователя?

8. Как получить данные, переданные из одной активности в другую? Как получить результат вызова активности?
9. . Опишите, как правильно обрабатывать событие поворота экрана пользователем. Что происходит при повороте с точки зрения жизненного цикла активности?
10. Что такое ресурсы? Для решения каких задач разработан данный механизм в Android? Какие преимущества даёт разработчику использование механизма ресурсов?
11. Какие типы ресурсов существуют? Как размещены в проекте файлы ресурсов?
12. Как можно использовать ресурсы в приложении непосредственно из программного кода, а также из других ресурсов?
13. Что такое ресурсы, зависящие от конфигурации? Для чего предназначен данных механизм и как его можно использовать?
14. Как, используя механизм ресурсов, создать главное меню или панель действий Android-приложения?
15. В чём отличия в реализации меню для ранних и поздних версий платформы Android?
16. Как обработать выбор действий из главного меню или панели действий?

Методические материалы, определяющие процедуру оценивания индивидуального опроса

При выставлении оценки преподаватель учитывает правильность ответа по содержанию, его последовательность, самостоятельность суждений и выводов, умение связывать теоретические положения с практикой, в том числе и с будущей профессиональной деятельностью.
Критерии оценки (в баллах):
- 15-12 баллов выставляется студенту, если ответ полный, правильный по содержанию, показана последовательность в ответах, самостоятельность суждений и выводов, умение связывать теоретические положения с практикой;
- 11-9 баллов выставляется студенту, если ответ полный, правильный по содержанию, не всегда последовательный, самостоятельность суждений и выводов, умение связывать теоретические положения с практикой;
- 8- 5 баллов выставляется студенту, если ответ фрагментарный, не показана последовательность в ответах, самостоятельность суждений и выводов вызывают затруднение, умение связывать теоретические положения с практикой;
- менее 5 баллов выставляется студенту, если ответ не полный, содержание фрагментарно правильно, не показана последовательность в ответах, самостоятельность суждений и выводов вызывают затруднение, не умение связывать теоретические положения с практикой;

Лабораторная работа

Лабораторная работа №2.
Основные алгоритмические конструкции Java.Модификаторы доступа в Java.
Цель работы овладение базовыми алгоритмическими конструкциями, модификаторами доступа на языке программирования Java.
Задания для самостоятельного выполнения
1. Написать метод, выводящий значение функции f(x) = sin(x)+cos(x) от введенного аргумента x. Вызвать метод несколько раз, чтобы результат был представлен в виде таблицы в диапазоне -5 до 5 с шагом 0,5.
2. Написать метод, выводящий значение функции f(x) = 1/sin(x)+7*x от введенного аргумента x. Вызвать метод несколько раз, чтобы результат был представлен в виде таблицы в диапазоне -5 до 5 с шагом 0,5.
3. Написать метод fun(x,i), выводящий значение функции от введенного аргумента x и параметра i. Вызвать метод несколько раз, чтобы результат был представлен в виде таблицы в диапазоне -5 до 5 с шагом 0,5. Так

при значении i =1 вычисляется f(x) = sin(2*x)
при значении i =2 вычисляется f(x) = sin(x)

при значении *i* =3 вычисляется f(*x*) = 2*sin(*x*)

1. Написать методfun(*x*,*i*), выводящий значение функции от введенного аргумента *x* и параметра *i*. Вызвать метод несколько раз, чтобы результат был представлен в виде таблицы в диапазоне -5 до 5 с шагом 0,5.Так

при значении *i* =1 вычисляется f(*x*) = cos(2**x*)
при значении *i* =2 вычисляется f(*x*) = cos(*x*)
при значении *i* =3 вычисляется f(*x*) = 2*cos(*x*)

1. Написать метод voidKvadr(***a***), который передает в метод ссылку на аргумент ***a***, а затем возводит это число в квадрат. Вызвать метод более 5 раз.
2. Написать метод voidKoren(***a***), который передает в метод ссылку на аргумент ***a***, а затем, берет квадратный корень из этого числа. Вызвать метод более 5 раз.
3. Написать метод voidMest(***a***,***b***), который передает в метод ссылку на аргументы ***a***,***b***, а затем меняет местами значения этих аргументов не ссылочного типа. Вызвать метод более 5 раз.
4. Написать метод voidMin_Maxt(***a***,***b***,***c***), который передает в метод ссылку на аргументы ***a***,***b***, ***c*** а затем меняет местами значения максимального и минимального из этих аргументов не ссылочного типа. Вызвать метод более 5 раз.
5. Написать метод voidmax(***a***,***b***,***c***) ***a***, ***b*** входные аргументы, с выходной. Параметру ***c*** присваивается максимальное значение из двух введенных параметров. Вызвать созданный метод в методе Main 5 раз.
6. Написать метод voidmin(***a***,***b***,***c***) ***a***, ***b***входные аргументы, свыходной.Параметру ***c*** присваивается минимальное значение из двух введенных параметров. Вызвать созданный метод в методе Main 5 раз.
7. Написать метод voidSfera(***r***,***s***,***v***) ***r***входной аргумент–радиус сферы, ***s***, ***v***выходные – площадь и объем сферы соответственно. Вызвать созданный метод в методе Main 5 раз.
8. Написать метод voidSfera(***r***,***s***,***v***) ***r***входной аргумент–ребро октаэдра, ***s***, ***v***выходные – площадь и объем октаэдра соответственно. Вызвать созданный метод в методе Main 5 раз.
9. Создать метод voidarray_s1(d), который находит сумму нечетных элементов массива, d это массив переменной длины. Вызвать созданный метод в методе Main 5 раз.
10. Создать метод voidarray_s2(d), который находит сумму элементов массива dc четными номерами,d это массив переменной длины. Вызвать созданный метод в методе Main 5 раз.
11. Создать метод voidarray_sort1(d), который сортирует элементы массива d по возрастанию, d это массив переменной длины. Применить сортировку методом выбора. Вызвать созданный метод в методе Main 5 раз.
12. Создать метод voidarray_sort2(d), который сортирует элементы массива d по убыванию, d это массив переменной длины. Применить пузырьковую сортировку. Вызвать созданный метод в методе Main 5 раз.
13. Создать метод voidarray_sort_1(d), который сортирует элементы массива d по возрастанию, d это массив переменной длины (Сортировка производится только над элементами с нечетными номерами). Применить сортировку методом выбора. Вызвать созданный метод в методе Main 5 раз.
14. Создать метод voidarray_sort_2(d), который сортирует элементы массива d по убыванию, d это массив переменной длины. (Сортировка производится только над элементами с нечетными номерами). Применить пузырьковую сортировку. Вызвать созданный метод в методе Main 5 раз.

Методические материалы, определяющие процедуру оценивания выполнения лабораторных работ

При оценивании лабораторной работы следует уделять внимание тому, насколько качественно выполнены задания и студентом демонстрируются владение освоенной тематикой; демонстрируются высокий уровень умения оперировать знаниями, анализировать информацию.
Критерии оценки (в баллах):
- 2 баллов выставляется студенту, если полно и качественно выполнены лабораторные задания; четко и правильно даны определения, раскрыто содержание понятий; верно использованы научные

термины; использованы ранее приобретенные знания; раскрыты причинно-следственные связи; демонстрируются высокий уровень умения оперировать научными категориями, анализировать информацию;
- 1 балла выставляется студенту, если качественно выполнены лабораторные задания с некоторыми недочетами; недостаточно раскрыто основное содержание учебного материала, не последовательно; определения понятий недостаточно четкие; допущены ошибки и неточности в использовании научной терминологии определения понятий; уровень умения оперировать научными категориями, анализировать информацию низкий;
- **0** балла выставляется студенту, если качественно выполнены фрагментарно лабораторные задания; с некоторыми недочетами если не раскрыто содержание учебного материала, изложено фрагментарно, определения понятий не четкие; допущены значительные ошибки в использовании научной терминологии определения понятий; уровень умения оперировать научными категориями, анализировать информацию очень низкий.

Экзаменационные билеты

Экзамен (зачет) является оценочным средством для всех этапов освоения компетенций. Структура экзаменационного билета: в билете указывается кафедра в рамках нагрузки которой реализуется данная дисциплина, форма обучения, направление и профиль подготовки, дата утверждения; билет может включать в себя теоретический(ие) вопрос(ы) и практическое задание (кейс-задание).

Примерные вопросы к экзамену, 4 курс / 8 семестр

1. Kotlin: ООП
2. Kotlin: Коллекции. Исключения
3. Структура Android проекта
4. Структура Layout файла. Атрибуты View элементов. Создание разметки с помощью кода
5. Элемент ListView. Класс ListAdapter. Паттерн ViewHolder. Элемент RecyclerView
6. Использование фрагментов в приложении. Создание фрагментов. Класс FragmentManager. Взаимодействие фрагментов с Activity
7. Process и Main Thread. Создание и жизненный цикл Thread. Синхронизация. Coroutines
8. Жизненный цикл Activity/Fragment. Жизненный цикл JetPack. Что такое LiveData? Концепция ViewModel
9. Создание сетевых запросов. HttpClient. Библиотека Retrofit. Библиотека Picasso, Glide
10. Методы сохранения данных. Saved instance state. SharedPreferences. SQLite, Room
11. Service. Создание Service. Фоновые процессы. Настройка WorkManager
12. Создание уведомлений. Показ уведомлений. Каналы и приоритет вызов. Реакция на действия пользователя

Образец экзаменационного билета

<table>
<tr><td colspan="2">МИНОБРНАУКИ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»
БИРСКИЙ ФИЛИАЛ УУНиТ
Кафедра информатики и экономики</td></tr>
<tr><td>Дисциплина: Разработка приложений для мобильных устройств
очная форма обучения
4 курс 8 семестр</td><td>Курсовые экзамены 20__-20__ г.
Направление 09.03.03 Прикладная информатика
Профиль: Прикладная информатика в информационной сфере</td></tr>
<tr><td colspan="2">Экзаменационный билет № 1</td></tr>
</table>

1. Kotlin: Коллекции. Исключения 2. Жизненный цикл Activity/Fragment. Жизненный цикл JetPack. Что такое LiveData? Концепция ViewModel 3. Решить задачу	
Дата утверждения: __.__._____	Заведующий кафедрой _____

Методические материалы, определяющие процедуру оценивания ответа на экзамене

Экзаменнационный билет состоит из теоретического вопроса и задачи, компонуемой из заданий лабораторных работ.
Критериями оценивания являются баллы, которые выставляются за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины: текущий контроль – максимум 40 баллов; рубежный контроль – максимум 30 баллов, поощрительные баллы – максимум 10.
При оценке ответа на экзамене максимальное внимание должно уделяться тому, насколько полно раскрыто содержание материала, четко и правильно даны определения, раскрыто содержание понятий, верно ли использованы научные термины, насколько ответ самостоятельный, использованы ли ранее приобретенные знания, раскрыты ли раскрыты причинно-следственные связи, насколько высокий уровень умения оперирования научными категориями, анализа информации, владения навыками практической деятельности.
Критерии оценки (в баллах):
- **25-30 баллов** выставляется студенту, если студент дал полные, развернутые ответы на все теоретические вопросы билета, продемонстрировал знание функциональных возможностей, терминологии, основных элементов, умение применять теоретические знания при выполнении практических заданий. Студент без затруднений ответил на все дополнительные вопросы. Практическая часть работы выполнена полностью без неточностей и ошибок;
- **17-24 баллов** выставляется студенту, если студент раскрыл в основном теоретические вопросы, однако допущены неточности в определении основных понятий. При ответе на дополнительные вопросы допущены небольшие неточности. При выполнении практической части работы допущены несущественные ошибки;
- **10-16** баллов выставляется студенту, если при ответе на теоретические вопросы студентом допущено несколько существенных ошибок в толковании основных понятий. Логика и полнота ответа страдают заметными изъянами. Заметны пробелы в знании основных методов. Теоретические вопросы в целом изложены достаточно, но с пропусками материала. Имеются принципиальные ошибки в логике построения ответа на вопрос. Студент не решил задачу или при решении допущены грубые ошибки;
- **1-10 баллов** выставляется студенту, если ответ на теоретические вопросы свидетельствует о непонимании и крайне неполном знании основных понятий и методов. Обнаруживается отсутствие навыков применения теоретических знаний при выполнении практических заданий. Студент не смог ответить ни на один дополнительный вопрос.

Перевод оценки из 100-балльной в четырехбалльную производится следующим образом:
- отлично – от 80 до 110 баллов (включая 10 поощрительных баллов);
- хорошо – от 60 до 79 баллов;
- удовлетворительно – от 45 до 59 баллов;
- неудовлетворительно – менее 45 баллов.

Задания для проведения промежуточной аттестации обучающихся

Промежуточная аттестация по дисциплине проводится в форме:

8 семестр - экзамен.

Проверяемыми на промежуточной аттестации элементами содержания являются темы дисциплины.

Для проверки *знаний* используются вопросы и задания в различных формах.

Умения, навыки (или опыт деятельности) и компетенции проверяются с помощью компетентностно-ориентированных задач (ситуационных, производственных или кейсового характера) и различного вида конструкторов.

Все задачи являются многоходовыми. Некоторые задачи, проверяющие уровень сформированности компетенций, являются многовариантными. Часть умений, навыков и компетенций прямо не отражена в формулировках задач, но они могут быть проявлены обучающимися при их решении.

В каждый вариант включаются задания по каждому проверяемому элементу содержания во всех перечисленных выше формах и разного уровня сложности. Такой формат позволяет объективно определить качество освоения обучающимися основных элементов содержания дисциплины и уровень сформированности компетенций.

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине при использовании модульно-рейтинговой системы

Критериями оценивания при модульно-рейтинговой системе являются баллы, которые выставляются преподавателем за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины:

для экзамена: текущий контроль – максимум 40 баллов; рубежный контроль – максимум 30 баллов, поощрительные баллы – максимум 10.

Шкалы оценивания:

для экзамена:

от 45 до 59 баллов – «удовлетворительно»;

от 60 до 79 баллов – «хорошо»;

от 80 баллов – «отлично».

Рейтинг-план дисциплины

Таблица перевода баллов текущего контроля в баллы рейтинга

	5	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**
0	0	0	0	0	0	0	0	0	0	0
1	5	3	2	2	1	1	1	1	1	1
2		5	4	3	2	2	2	2	2	1
3			5	4	3	3	3	2	2	2
4				5	4	4	3	3	3	2
5					5	5	4	4	3	3
6						5	5	4	4	3
7							5	5	4	4
8								5	5	4
9									5	5
10										5

Рейтинг-план дисциплины представлен в Приложении 1.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

5.1. Основная учебная литература

1. Сильвен, Р. Android NDK. Разработка приложений под Android на C/C++ / Р. Сильвен ; пер. с англ. Киселева А.Н.. — Электрон. дан. — Москва : ДМК Пресс, 2012. — 496 с. — Режим доступа: https://e.lanbook.com/book/9126.
2. Виссер, Д. Разработка обслуживаемых программ на языке Java / Д. Виссер ; пер. с англ. Р. Н. Рагимова. — Электрон. дан. — Москва : ДМК Пресс, 2017. — 182 с. https://e.lanbook.com/book/105834.

5.2. Дополнительная учебная литература

1. Соколова, В.В. Разработка мобильных приложений : учебное пособие / В.В. Соколова ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Томский государственный университет». - Томск : Издательство Томского политехнического университета, 2015. - 176 с.- URL: http://biblioclub.ru/index.php?page=book&id=442808
2. Семакова, А. Введение в разработку приложений для смартфонов на ОС Android / А. Семакова. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 103 с. - URL: http://biblioclub.ru/index.php?page=book&id=429181
3. Введение в разработку приложений для ОС Android / Ю.В. Березовская, О.А. Юфрякова, В.Г. Вологдина и др. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 434 с. URL: http://biblioclub.ru/index.php?page=book&id=428937
4. Ретабоуил, С. Android NDK: руководство для начинающих [Электронный ресурс] : руководство / С. Ретабоуил ; пер. с англ. Киселев А. Н.. — Электрон. дан. — Москва : ДМК Пресс, 2016. — 518 с. — URL: https://e.lanbook.com/book/82810
5. Head First. Программирование для Android : / Д. Гриффитс, Д. Гриффитс .— 2-е изд. — Санкт-Петербург : Питер, 2018 .— Обяз. экз.
6. Тазетдинов, Б.И. Лабораторный практикум по базовому курсу программирования на языке C #: [Электронный ресурс]: учебное пособие / Башкирский государственный университет, Бирский филиал; авт.- сост. Б.И. Тазетдинов. — Бирск: РИЦ БашГУ, 2017. <URL:https://elib.bashedu.ru/dl/local/Tazetdinov_avt-sost_Laboratornyj praktikum_Birsk_2017.pdf> .

5.3. Другие учебно-методические материалы

Перечень рекомендуемых ресурсов информационно-телекоммуникационной сети «Интернет», находящихся в свободном доступе

1. центр справки java: [сайт]. https://www.java.com/ru/download/help/

6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Перечень ресурсов информационно-телекоммуникационной сети «Интернет» и программного обеспечения, необходимых для освоения дисциплины

1. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс]. – Режим доступа: https://elibrary.ru/.
2. Электронная библиотечная система «Лань» [Электронный ресурс]. – Режим доступа: https://e.lanbook.com/.
3. Университетская библиотека онлайн biblioclub.ru [Электронный ресурс]. – Режим доступа: http://biblioclub.ru/.
4. Электронная библиотека УУНиТ [Электронный ресурс]. – Режим доступа: https://elib.bashedu.ru/.
5. Российская государственная библиотека [Электронный ресурс]. – Режим доступа: https://www.rsl.ru/.
6. Национальная электронная библиотека [Электронный ресурс]. – Режим доступа: https://xn--90ax2c.xn--p1ai/viewers/.
7. Национальная платформа открытого образования npoed.ru [Электронный ресурс]. – Режим доступа: http://npoed.ru/.
8. Электронное образование Республики Башкортостан [Электронный ресурс]. – Режим доступа: https://edu.bashkortostan.ru/.
9. Информационно-правовой портал Гарант.ру [Электронный ресурс]. – Режим доступа: http://www.garant.ru/.

Программное обеспечение

1. LIBREOFFICE - Договор №0301400003023000002 от 14.03.2023 (бессрочный)
2. Браузер Яндекс - Бесплатная лицензия https://yandex.ru/legal/browser_agreement/index.html
3. Операционная система Astra Linux - Договор 0301400003023000002 от 14 марта 2023 бессрочно
4. Система дистанционного обучения Moodle - Бесплатная лицензия http://www.gnu.org/licenses/gpl.html
5. Android studio - Бесплатная лицензия https://developer.android.com/studio/terms

7. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Аудитория 231(ФМ)	Лекционная, Для консультаций, Для контроля и аттестации, Для лабораторных занятий	Доска, коммутатор, компьютер, мебель. Программное обеспечение 1. Браузер Яндекс 2. Операционная система Astra Linux 3. Android studio 4. Система дистанционного обучения Moodle
Аудитория 301 Читальный зал (электронный каталог)(ФМ)	Для самостоятельной работы	Компьютер, мебель, принтер, сканер hp scanyet g2410.

		Программное обеспечение 1. LIBREOFFICE 2. Операционная система Astra Linux 3. Браузер Яндекс
Аудитория 311(ФМ)	Лекционная, Для консультаций, Для контроля и аттестации, Для лабораторных занятий	Доска, компьютер, мебель, проектор, экран. Программное обеспечение 1. Браузер Яндекс 2. Операционная система Astra Linux 3. Система дистанционного обучения Moodle 4. Android studio 5. LIBREOFFICE
Аудитория 411(ФМ)	Лекционная, Для консультаций, Для контроля и аттестации, Для лабораторных занятий	Компьютер, мебель, проектор, экран. Программное обеспечение 1. Система дистанционного обучения Moodle 2. Android studio 3. Браузер Яндекс 4. Операционная система Astra Linux 5. LIBREOFFICE
Аудитория 411 а(ФМ)	Для хранения оборудования	Компьютер, мебель. Программное обеспечение 1. LIBREOFFICE 2. Операционная система Astra Linux
Аудитория 420(ФМ)	Для самостоятельной работы	Мебель, нетбук. Программное обеспечение 1. LIBREOFFICE 2. Операционная система Astra Linux
Аудитория 422 а(ФМ)	Для хранения оборудования	Ноутбук. Программное обеспечение 1. LIBREOFFICE 2. Операционная система Astra Linux