

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 42

УТВЕРЖДАЮ

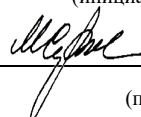
Руководитель направления

Д.Т.Н., проф.

(должность, уч. степень, звание)

М.Б. Сергеев

(инициалы, фамилия)



(подпись)

«07» июня 2023 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«ИТ-модуль "Разработка мобильных приложений"»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.03.01
Наименование направления подготовки/ специальности	Информатика и вычислительная техника
Наименование направленности	Компьютерные технологии, системы и сети
Форма обучения	очная

Санкт-Петербург– 2023

Лист согласования рабочей программы дисциплины

Программу составил (а)

Старший преподаватель
(должность, уч. степень, звание)



06.06.2023

(подпись, дата)

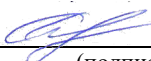
В.А. Ушаков
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 42

«09» июня 2023 г, протокол №9/2022-2023

Заведующий кафедрой № 42

Д.Т.Н., доц.
(уч. степень, звание)



07.06.2023

(подпись, дата)

С.В. Мичурин
(инициалы, фамилия)

Ответственный за ОП ВО 09.03.01(04)

(должность, уч. степень, звание)



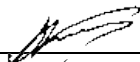
07.06.2023

(подпись, дата)

Д.В. Куртяник
(инициалы, фамилия)

Заместитель директора института №4 по методической работе

доц., к.т.н., доц.
(должность, уч. степень, звание)



07.06.2023

(подпись, дата)

А.А. Ключарев
(инициалы, фамилия)

Аннотация

Дисциплина «ИТ-модуль "Разработка мобильных приложений"» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 09.03.01 «Информатика и вычислительная техника» направленности «Компьютерные технологии, системы и сети». Дисциплина реализуется кафедрой «№42».

Дисциплина не является обязательной при освоении обучающимся образовательной программы и направлена на углубленное формирование следующих компетенций:

УК-6 «Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни».

Содержание дисциплины охватывает круг вопросов, связанных с разработкой мультимедийных мобильных приложений для устройств под управлением операционной системы Android.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции в формате онлайн, практические занятия, самостоятельная работа обучающегося.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме дифференцированного зачета.

Общая трудоемкость освоения дисциплины составляет 7 зачетных единиц, 252 часа.

Язык обучения по дисциплине «русский».

1. Перечень планируемых результатов обучения по дисциплине

1.1. Целью преподавания дисциплины является получение обучающимися необходимых знаний, умений и навыков в области разработки прикладного программного обеспечения на языке программирования Kotlin и приложений для мобильных устройств под управлением операционной системы Android.

1.2. Дисциплина является факультативной дисциплиной по направлению образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Универсальные компетенции	УК-6 Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	УК-6.3.2 знать образовательные Интернет-ресурсы, возможности и ограничения образовательного процесса при использовании цифровых технологий УК-6.У.2 уметь находить информацию и использовать цифровые инструменты в целях самообразования УК-6.В.2 владеть навыками использования цифровых инструментов для саморазвития и самообразования

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Информатика»,
- «Основы программирования»,
- «Учебная практика»

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и могут использоваться при изучении других дисциплин:

- «Производственная технологическая практика»,
- «Производственная преддипломная практика»,
- «Выпускная квалификационная работа».

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам	
		№5	№6
1	2	3	4
Общая трудоемкость дисциплины, ЗЕ/ (час)	7/ 252	3/ 108	4/ 144
Из них часов практической подготовки			

Аудиторные занятия , всего час.	119	51	68
в том числе:			
лекции (Л), (час)	34	17	17
практические/семинарские занятия (ПЗ), (час)	85	34	51
лабораторные работы (ЛР), (час)			
курсовой проект (работа) (КП, КР), (час)			
экзамен, (час)			
Самостоятельная работа , всего (час)	133	57	76
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**)	Зачет, Дифф. Зач.	Зачет	Дифф. Зач.

Примечание: ** кандидатский экзамен

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП (час)	СРС (час)
Семестр 5					
Раздел 1. Введение в Kotlin, отличие от Java	2	2	-	-	3
Раздел 2. Лямбда-выражения	2		-	-	6
Раздел 3. Разработка классов, наследование и полиморфизм	2	4	-	-	6
Раздел 4. Интерфейсы и аннотации	2	4	-	-	6
Раздел 5. Коллекции	2	4	-	-	6
Раздел 6. Функциональное программирование	2	4	-	-	6
Раздел 7. Многопоточное программирование	2	4	-	-	6
Раздел 8. Работа с базами данных	1	4	-	-	6
Раздел 9. Пользовательский интерфейс	1	4	-	-	6
Раздел 10. Веб-приложения	1	4	-	-	6
Итого в семестре:	17	34			57
Семестр 6					
Раздел 11. Введение в Android разработку	2	2	-	-	7
Раздел 12. Намерения, меню и работа с данными	4	7	-	-	7
Раздел 13. СУБД и использование сетевых сервисов	4	4	-	-	7
Раздел 14. Фрагменты (Fragments)	2	4	-	-	7
Раздел 15. Нативный пользовательский интерфейс	4	7	-	-	7
Раздел 16. Проектирование современного программного обеспечения	4	4	-	-	9
Раздел 17. Использование паттернов в нативных мобильных приложениях	4	4	-	-	7
Раздел 18. Работа с многопоточностью в нативных мобильных приложениях	4	4	-	-	7

Раздел 19. Тестирование нативных мобильных приложений	2	4	-	-	7
Раздел 20. Размещение нативных мобильных приложений в магазине приложений	4	4	-	-	7
Итого в семестре:	17	51			76
Итого	34	85	0	0	133

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
1	Языковые лексемы; модули, пакеты, файлы и классы; настройка среды; константы, переменные и функции; строки, строковые шаблоны; система типов; допустимость null; совместимость с Java; создание и вызов функций; именованные аргументы, значения по умолчанию; операторы ветвления; when; автоматическое приведение типов; операторы циклов; диапазоны; пары и триплеты; деструктурирование и мультидекларации; встраивание функций; концепция исключений; исключения Kotlin; использование операторов try, catch и finally; идиома try-with-resource
2	Синтаксис лямбда-выражений; ссылки на методы; функциональные интерфейсы
3	Обзор основных концепций ООП; перечисления; объявление класса; поля, свойства, методы, модификаторы доступа; конструкторы и инициализаторы; data-классы; перегрузка операторов; функции-расширения и свойства-расширения; отложенная инициализация; внутренние классы; вложенные классы; шаблон синглтон; объекты-компаньоны; наследование как механизм повторного использования кода; конструктор при наследовании; преобразование типов; виртуальные методы и позднее связывание; абстрактные классы и методы; open, final, sealed; делегирование как альтернатива
4	Концепция исключений; исключения Kotlin; использование операторов try, catch и finally; идиома try-with-resource
5	Иерархия классов коллекций; концепция параметризованных типов данных; работа с параметризованными методами и интерфейсами; обзор возможностей коллекций в Kotlin
6	Функции высшего порядка; Работа с коллекциями в функциональном стиле; Последовательности.
7	Плюсы и минусы многопоточных приложений; средства для управления многопоточностью; класс Thread и интерфейс Runnable; создание потоков; мониторы и синхронизация потоков; современные средства по управлению потоками (Executors, Fork/Join Framework); новые потоки, безопасные коллекции и классы (ThreadLocalRandom, AtomicInteger и др.); reactive streams; корутины
8	MySQL и PostgreSQL базы данных; JDBC
9	Swing; TornadoFX
10	Сервлеты и JSP; JSF; веб-сервисы

11	Android Studio, эмулятор, компоненты приложения, встроенные базы данных, вызов веб-сервисов, преимущества Android, архитектура Android, особенности платформы Android, основные компоненты Android, безопасность и полномочия (Permissions), установка и настройка компонентов среды разработки, активности (Activity) в Android, создание Активности, жизненный цикл Активности, стеки Активностей, состояния Активностей, отслеживание изменений состояния Активности, ресурсы, отделение ресурсов от кода программы, создание ресурсов, простые значения, визуальные стили и темы, изображения
12	Адаптеры в Android, AdapterView - ListView, GridView, использование Адаптеров для привязки данных, RecyclerView, понятие контекста, намерения в Android, использование Намерений для запуска Активностей, неявные намерения, сохранение состояния и настроек приложения, общие настройки (Shared Preferences), работа с файлами, использование статических файлов как ресурсов, меню в Android, основное и контекстное меню, описание меню с помощью XML, PopupMenu, Bottom Navigation, NavigationDrawer
13	Работа с базами данных в Android, курсоры (Cursor) и ContentValues, работа с СУБД SQLite, работа с СУБД без адаптера, особенности работы с БД в Android, выполнение запросов для доступа к данным, изменение данных в БД, использование SimpleCursorAdapter, использование интернет-сервисов, Picasso, Retrofit2, архитектура приложений с Firebase Cloud Messaging, добавление Firebase Cloud Messaging в проект, посылка и прием сообщений, проверка сетевых соединений, отслеживание состояния соединений
14	Создание Фрагментов, добавление пользовательского интерфейса, добавление фрагментов к Активностям, управление Фрагментами, транзакции с Фрагментами, взаимодействие Фрагментов и Активностей, жизненный цикл Фрагментов, навигация, критика
15	Material Design, использование внешних ресурсов в коде приложения, использование ресурсов внутри ресурсов, локализация приложения, класс Application, обработка событий жизненного цикла приложения, пользовательский интерфейс, представления (View), разметка (Layout), пользовательский интерфейс без разметки, Jetpack Compose, руководство по Material Design, адаптивная верстка
16	Android Jetpack (ViewModel, Room, DataStore, WorkManager, Navigation, CameraX, Compose), чистая архитектура, модульность, работа с сетью (OkHttp, Retrofit), сериализация (Moshi, Kotlin Serialization), использование библиотек для загрузки изображений (Coil, Glide)
17	Архитектурные паттерны для уровня представления (MVVM, MVI), внедрение зависимостей (Hilt, Dagger, Koin)
18	Жизненный цикл процесса, потоки, фоновые потоки, использование AsyncTask, корутины Kotlin, реактивность/управление потоками (Kotlin Coroutines, RxJava), оптимизация R8
19	JUnit 5, Mockk, Espresso, Robolectric
20	Play Feature Delivery, отечественный магазин приложений RuStore, отечественный магазин приложений NashStore, отечественный магазин приложений RuMarket, инструменты для повышения производительности на всем жизненном цикле (Benchmark, Baseline Profiles, App Startup, Firebase Performance Monitoring, JankStats library), обновления в нативных мобильных приложениях, отзывы в нативных мобильных приложениях, безопасность нативных мобильных приложений, логирование (Timber), линтер (Android Lint, Detekt, Ktlint)

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 5					
1	Разработка консольного приложения	Онлайн выполнение практических заданий	2		1
2	Работа с классами	Онлайн выполнение практических заданий	4		3
3	Работа с интерфейсами и аннотациями	Онлайн выполнение практических заданий	4		4
4	Работа с коллекциями	Онлайн выполнение практических заданий	4		5
5	Функциональное программирование при работе с коллекциями	Онлайн выполнение практических заданий	4		6
6	Потоки, Executors, Корутины	Онлайн выполнение практических заданий	4		7
7	Работа с базами данных	Онлайн выполнение практических заданий	4		8
8	Разработка графического приложения	Онлайн выполнение практических заданий	4		9
9	Создание веб-приложений	Онлайн выполнение практических заданий	4		10
Итого в семестре:			34		
Семестр 6					
1	Создание первого приложения под ОС Android	Онлайн выполнение практических заданий	7		11
2	Создание пользовательского интерфейса активностей и локализация приложения с помощью ресурсов	Онлайн выполнение практических заданий	4		15
3	Работа с активностями	Онлайн выполнение практических	4		12

		заданий			
4	Работа с базой данных Preferences	Онлайн выполнение практических заданий	7		12,13
5	Разработка нативного пользовательского интерфейса	Онлайн выполнение практических заданий	4		14,15
6	Использование SQLite	Онлайн выполнение практических заданий	4		13
7	Применение архитектурных компонентов Google	Онлайн выполнение практических заданий	4		17
8	Многопоточные мобильные приложения на Kotlin	Онлайн выполнение практических заданий	6		18
9	Тестирование нативного мобильного приложения	Онлайн выполнение практических заданий	4		19
10	Размещение нативного мобильного приложения в магазине	Онлайн выполнение практических заданий	7		20
Итого в семестре:			51		
Всего			85		

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Учебным планом не предусмотрено				
Всего				

4.5. Курсовое проектирование/ выполнение курсовой работы

Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 5, час	Семестр 6, час
----------------------------	------------	----------------	----------------

1	2	3	4
Изучение теоретического материала дисциплины (ТО)	58	27	31
Оформление практических работ (ПР)	25	10	15
Подготовка к текущему контролю успеваемости (ТКУ)	25	10	15
Подготовка к промежуточной аттестации (ПА)	25	10	15
Всего:	133	57	76

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. 7-11.

6. Перечень печатных и электронных учебных изданий
Перечень печатных и электронных учебных изданий приведен в таблице 8.
Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров в библиотеке (кроме электронных экземпляров)
https://znanium.com/catalog/product/1756155	Скин, Д. Kotlin. Программирование для профессионалов : практическое руководство / Д. Скин, Гринхол Дэвид. - Санкт-Петербург : Питер, 2020. - 464 с.	-
https://znanium.com/catalog/product/1733498	Гриффитс, Д. Head First. Kotlin : практическое руководство / Д. Гриффитс, Д. Гриффитс. - Санкт-Петербург : Питер, 2020. - 464 с.	-
https://znanium.com/catalog/product/1094968	Сомон, П. Волшебство Kotlin : практическое руководство / П. Сомон ; пер. с англ. А. Н. Киселева. - Москва : ДМК Пресс, 2020. - 536 с.	-
https://e.lanbook.com/book/112926	Жемеров, Д. Kotlin в действии / Д. Жемеров, С. Исакова ; перевод с английского А. Н. Киселев. — Москва : ДМК Пресс, 2018. — 402 с.	-
https://e.lanbook.com/book/241007	Коузен, К. Kotlin. Сборник рецептов / К. Коузен ;	-

	перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2021. — 220 с.	
004 П 33	Пирская, Л. В. Разработка мобильных приложений : учебное пособие : в 2 ч. ч. 2 / Л. В. Пирская. - СПб : ГУАП, 2021. - 65 с.	5
004 М 27	Марковский С. Г., Марковская Н. В. Мультимедиа технологии в мобильных системах: лабораторный практикум.- СПб.: ГУАП, 2011. - 90 с.	80
https://urait.ru/bcode/451366	Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебное пособие для вузов / В. В. Соколова. — Москва : Издательство Юрайт, 2020. — 175 с.	-
004 Х 20 https://elibrary.ru/item.asp?id=21558057	Харди Б., Филлипс Б., Стюарт К., Марсикано К. Android. Программирование для профессионалов. 3-е изд.- СПб: Питер, 2017. – 688с.	2
https://www.labirint.ru/books/537853/	Дейтел П., Дейтел Х., Уолд А. Android для разработчиков. 3-е изд.- СПб: Питер, 2016. – 512с.	-
www.biblio-online.ru/book/D80F822D-BA6D-45E9-B83B-8EC049F5F7D9	Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебное пособие для прикладного бакалавриата / В. В. Соколова. — М. : Издательство Юрайт, 2018. — 175 с.	-
004 С 32	Сержантова М. В. Компьютерные сети в электромеханических системах : учебно-методическое пособие / М. В. Сержантова, Г. И. Король. - СПб : ГУАП, 2022. - 84 с.	5
004	Пирская, Л. В. Разработка	5

П 33	мобильных приложений : учебное пособие : в 2 ч. ч. 2 / Л. В. Пирская. - СПб : ГУАП, 2021. - 65 с.	
004 У 93	Ушаков В. А. Мультимедиа в мобильных системах: лабораторный практикум / В.А. Ушаков. – СПб.: ГУАП, 2020. – 97 с.	5
https://publications.hse.ru/books/420071117	Цифровые технологии в российской экономике / К.О. Вишневский, Л. М. Гохберг, В. В. Дементьев и др.; под ред. Л.М. Гохберга. – М.: НИУ ВШЭ, 2021. – 116 с. DOI: 10.17323/978-5-7598-2199-1	
https://issek.hse.ru/news/551331807.html	Цифровая экономика: 2022 : краткий статистический сборник / Г. И. Абдрахманова, С. А. Васильковский, К. О. Вишневский и др. – М.: НИУ ВШЭ, 2022. – 124 с. DOI: 10.17323/978-5-7598- 2599-9	-

7. Перечень электронных образовательных ресурсов
информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
https://docs.cntd.ru/document/1200157208	ГОСТ 7.32-2017
https://kotlinlang.org/	Kotlin
https://kotlinlang.ru/	Руководство по языку Kotlin
https://kotlins.org/	Kotlin
https://metanit.com/kotlin/tutorial/	Руководство по языку Kotlin
https://code.tutsplus.com/ru/categories/kotlin	Kotlin
https://developer.android.com/kotlin	Develop Android apps with Kotlin
https://developer.android.com/guide/	Documentation
http://www.fandroid.info/	Обучение разработке мобильных приложений и игр для ANDROID
http://startandroid.ru	Start Android - учебник

	по Android для начинающих и продвинутых
https://www.youtube.com/channel/UCzE7HcbvyEiS5ea1rVRbPLQ	YouTube-канал Start Android
https://metanit.com/java/android/	Программирование под ОС Андроид
https://stackoverflow.com	Learn, Share, Build

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
1	IntelliJ IDEA Community Edition
2	NetBeans
3	Android Studio
4	Среды Android Studio
5	Средства прототипирования интерфейсов: Figma
6	Руководства по стилям: Google Material Design
7	Genymotion Desktop
8	Ionic Framework
9	Pencil

8.2. Перечень информационно-справочных систем, используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование	Ссылка на ресурс
1	Научная электронная библиотека	https://www.elibrary.ru/defaultx.asp
2	Портал Федеральных государственных образовательных стандартов высшего образования	http://www.fgosvo.ru
3	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru
4	Федеральный центр информационно-образовательных ресурсов	http://fcior.edu.ru
5	Библиотека ГУАП	https://lib.guap.ru/jirbis2/
6	Российская национальная библиотека	http://www.rsl.ru
7	Научная электронная библиотека	https://cyberleninka.ru/
8	Государственная публичная научно-техническая библиотека России	http://www.gpntb.ru
9	Центральная библиотека образовательных ресурсов Минобрнауки РФ	www.edulib.ru
10	Электронно-библиотечная система (ЭБС)	https://www.urait.ru/

	«Юрайт»	
11	Электронно-библиотечная система «Book.ru»	https://www.book.ru/
12	Электронно-библиотечная система «Znanium»	https://new.znanium.com/
13	Электронно-библиотечная система издательства «Лань»	https://e.lanbook.com/
14	Электронно-библиотечная система «IPRbooks»	http://www.iprbookshop.ru

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1.	Система дистанционного обучения ГУАП (СДО ГУАП) – https://lms.guap.ru/	-

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Зачет	Тесты
Дифф. зачет	Тесты

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 – Критерии оценки уровня сформированности компетенций

Оценка компетенции		Характеристика сформированных компетенций
100-балльная шкала	5-балльная шкала	
85 ≤ K ≤ 100	«отлично» «зачтено»	<ul style="list-style-type: none"> – обучающийся глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий.

Оценка компетенции		Характеристика сформированных компетенций
100-балльная шкала	5-балльная шкала	
70 ≤ K ≤ 84	«хорошо» «зачтено»	<ul style="list-style-type: none"> – обучающийся твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий.
55 ≤ K ≤ 69	«удовлетворительно» «зачтено»	<ul style="list-style-type: none"> – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий.
K ≤ 54	«неудовлетворительно» «не зачтено»	<ul style="list-style-type: none"> – обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
	Учебным планом не предусмотрено	

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код индикатора
	Учебным планом не предусмотрено	

Перечень тем для курсового проектирования/выполнения курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для курсового проектирования/выполнения курсовой работы

№ п/п	Примерный перечень тем для курсового проектирования/выполнения курсовой работы
	Учебным планом не предусмотрено

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора
1	<pre>class Person (val name: String)</pre> <ul style="list-style-type: none"> <input type="radio"/> Он package-private <input type="radio"/> Он может быть расширен другими классами <input type="radio"/> Он public <input type="radio"/> У него приватное свойство "name" 	УК-6.3.2 УК-6.У.2 УК-6.В.2
2	<p>Есть ли у Kotlin примитивные типы данных, такие как <code>int</code>, <code>long</code>, <code>float</code>?</p> <ul style="list-style-type: none"> <input type="radio"/> Нет, Kotlin не имеет и не использует примитивные типы данных. <input type="radio"/> Нет, не на уровне языка. Но компилятор Kotlin использует примитивы JVM для лучшей производительности. <input type="radio"/> Да, но Kotlin всегда конвертирует их в не примитивные аналоги. <input type="radio"/> Да, Kotlin в этом отношении похож на Java. 	УК-6.3.2 УК-6.У.2 УК-6.В.2
3	<p>Что такое <code>to</code> в приведенном ниже примере:</p> <pre>val test = 33 to 42</pre> <ul style="list-style-type: none"> <input type="radio"/> Инфиксная функция, создающая пару (33, 42) <input type="radio"/> Ключевое слово Kotlin для создания пары (33, 42) <input type="radio"/> Ключевое слово для создания диапазона от 33 до 42 <input type="radio"/> Опечатка 	УК-6.3.2 УК-6.У.2 УК-6.В.2
4	<p>Какое из объявлений функций является валидным?</p> <ul style="list-style-type: none"> <input type="radio"/> <pre>int sum(int a, int b)</pre> <input type="radio"/> <pre>int sum(a: Int, b: Int)</pre> <input type="radio"/> <pre>function sum(a: Int, b: Int): Int</pre> <input type="radio"/> <pre>fun sum(a: Int, b: Int): Int</pre> 	УК-6.3.2 УК-6.У.2 УК-6.В.2
5	<p>В чем ключевое отличие <code>Iterable<T></code> и <code>Sequence<T></code> в Kotlin?</p> <ul style="list-style-type: none"> <input type="radio"/> <code>Iterable<T></code> работает только с <code>immutable</code> коллекциями, <code>Sequence<T></code> применим к <code>mutable</code> <input type="radio"/> Нет никакой разницы, т. к. <code>Sequence<T></code> аналог <code>Iterable<T></code> <input type="radio"/> Последовательности обрабатываются лениво, итераторы жадно <input type="radio"/> Последовательности обрабатываются по очереди, итераторы параллельно (многопоточно) 	УК-6.3.2 УК-6.У.2 УК-6.В.2

6	<p>Что выведет следующий код?</p> <pre>val listA = mutableListOf(1, 2, 3) val listB = listA.add(4) print(listB)</pre> <p> <input type="radio"/> [1, 2, 3, 4] <input type="radio"/> True <input type="radio"/> Ничего, тут ошибка компиляции <input type="radio"/> Unit </p>	<p>УК-6.3.2 УК-6.У.2 УК-6.В.2</p>
7	<p>В чем разница между a и b?</p> <pre>var a: String? = "KotlinQuiz" var b: String = "KotlinQuiz"</pre> <p> <input type="radio"/> a является volatile, как в Java <input type="radio"/> b является final и не может быть изменено <input type="radio"/> a является final и не может быть изменено <input type="radio"/> b никогда не сможет стать null </p>	<p>УК-6.3.2 УК-6.У.2 УК-6.В.2</p>
8	<p>Как в Kotlin правильно объявить переменную целочисленного типа?</p> <p> <input type="radio"/> <code>var i : int = 42</code> <input type="radio"/> <code>let i = 42</code> <input type="radio"/> <code>int i = 42</code> <input type="radio"/> <code>var i : Int = 42</code> </p>	<p>УК-6.3.2 УК-6.У.2 УК-6.В.2</p>
9	<p>Какой тип у arr?</p> <pre>val arr = arrayOf(1, 2, 3)</pre> <p> <input type="radio"/> <code>Array<Int></code> <input type="radio"/> <code>Int[]</code> <input type="radio"/> <code>int[]</code> <input type="radio"/> <code>IntArray</code> </p>	<p>УК-6.3.2 УК-6.У.2 УК-6.В.2</p>
10	<p>Укажите правильный синтаксис для преобразования строки "42" в long</p> <p> <input type="radio"/> <code>val l: Long = <Long>"42"</code> <input type="radio"/> <code>val l: Long = "42".toLong()</code> <input type="radio"/> <code>val l: Long = (Long)"42"</code> <input type="radio"/> <code>val l: Long = Long.parseLong("42")</code> </p>	<p>УК-6.3.2 УК-6.У.2 УК-6.В.2</p>
11	<p>Какой метод находит объект по id? findId FindViewID</p>	<p>УК-6.3.2 УК-6.У.2 УК-6.В.2</p>

	findById findViewById findViewById	
12	Для описания графического интерфейса используются ресурсы типа: string drawable layout ui	УК-6.3.2 УК-6.У.2 УК-6.В.2
13	Какие из перечисленных ресурсов не используются в Andorid приложениях? font drawable layout raw используются все перечисленные	УК-6.3.2 УК-6.У.2 УК-6.В.2
14	Зачем нужны фрагменты? Позволяют встроить «подстраницу» с кнопками, картинками и всем остальным Это новые страницы приложения с кнопками, картинками и всем остальным Позволяют встроить видео Позволяют встроить картинки	УК-6.3.2 УК-6.У.2 УК-6.В.2
15	Какой класс отвечает за всплывающие подсказки? Hint Text Alert Toast Message	УК-6.3.2 УК-6.У.2 УК-6.В.2
16	Куда необходимо помещать изображения? В любую папку В папку values В папку drawable В папку res В папку layout	УК-6.3.2 УК-6.У.2 УК-6.В.2
17	Что делает это свойство? android:backgroundTint = «@android:color/holo_green_light» Устанавливает тень объекта Устанавливает задний фон объекта Устанавливает задний фон текста Устанавливает цвет текста	УК-6.3.2 УК-6.У.2 УК-6.В.2
18	Какие программы необходимы для работы с Андроид? Только Android Studio JDK, Android Studio и Visual Studio JDK, Android Studio Только JDK	УК-6.3.2 УК-6.У.2 УК-6.В.2
19	Какие аннотации можно использовать для написания тестов? @Before @After @BeforeClass @Test Все перечисленные	УК-6.3.2 УК-6.У.2 УК-6.В.2

20	Как называется библиотека для для создания заглушек вместо реальных объектов для использования их в тестах? Mockito JUnit Все перечисленные	УК-6.3.2 УК-6.У.2 УК-6.В.2
----	--	----------------------------------

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
	Не предусмотрено

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала.

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

- получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;
- получение опыта творческой работы совместно с преподавателем;
- развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.
- появление необходимого интереса, необходимого для самостоятельной работы;
- получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;
- научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);
- получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

- изложение теоретических вопросов, связанных с рассматриваемой темой;
- обобщение изложенного материала.

11.2. Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающаяся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающимся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимся практических занятий:

- закрепление, углубление, расширение и детализация знаний при решении конкретных задач;
- развитие познавательных способностей, самостоятельности мышления, творческой активности;
- овладение новыми методами и методиками изучения конкретной учебной дисциплины;
- выработка способности логического осмысления полученных знаний для выполнения заданий;
- обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

Задание и требования к проведению практических работ

Вариант задания по каждому практическому занятию обучающийся получает в соответствии с номером по журналу группы. Перед проведением практического занятия обучающемуся следует внимательно ознакомиться с методическими указаниями по его выполнению. В соответствии с заданием обучающийся должен подготовить необходимые данные, выполнить указанную последовательность действий, получить требуемые результаты, оформить отчет о практической работе. Отчет о выполнении практической работы загружается в СДО ГУАП.

Структура и форма отчета о практической работе

- 1 Титульный лист;
- 2 Задание к практической работе;
- 3 Листинг с кодом программы (.kt)

или

Листинг с кодом Android-приложения (kt), manifest и Activity (xml);

- 4 Экранные формы с результатами работы программы;
- 5 Выводы по практической работе.

Требования к оформлению отчета о практической работе

При оформлении отчета о практической работе следует пользоваться ГОСТ 7.32-2017 издания 2017 года. Правила оформления текстовых документов по ГОСТ 7.32-2017, а также титульные листы практических работ представлены на сайте ГУАП (<https://guap.ru/standart/doc>).

В случае загрузки отчета о выполнении практической работы после крайнего срока сдачи начисляется уменьшенное количество баллов.

11.3. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Для обучающихся по заочной форме обучения, самостоятельная работа может включать в себя контрольную работу.

В процессе выполнения самостоятельной работы, у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет им развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;
- методические указания по выполнению контрольных работ (для обучающихся по заочной форме обучения).

Материалы для самостоятельной работы представлены с применением электронного обучения, дистанционных образовательных технологий (онлайн-курс).

Курс размещён в системе дистанционного обучения ГУАП (СДО ГУАП).

11.4. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Возможные методы текущего контроля успеваемости обучающихся:

- устный опрос на занятиях;
- систематическая проверка выполнения индивидуальных заданий;
- защита отчётов по практическим работам;
- тестирование;
- контроль самостоятельных работ (в письменной или устной формах);
- контроль выполнения индивидуального задания на практику;
- иные виды, определяемые преподавателем.

Обязательным элементом текущего контроля успеваемости является загрузка в СДО ГУАП всех отчетов о выполнении практических работ, а также прохождение всех предусмотренных тестов.

11.5. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

– зачет – это форма оценки знаний, полученных обучающимся в ходе изучения учебной дисциплины в целом или промежуточная (по окончании семестра) оценка знаний обучающимся по отдельным разделам дисциплины с аттестационной оценкой «зачтено» или «не зачтено».

– дифференцированный зачет – это форма оценки знаний, полученных обучающимся при изучении дисциплины, при выполнении курсовых проектов, курсовых работ, научно-исследовательских работ и прохождении практик с аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Обязательным требованием к получению зачета по дисциплине является просмотр всех лекций, успешное прохождение всех предусмотренных тестов, а также выполнение всех практических работ. Успешным прохождением теста считается правильный ответ не менее, чем на 60 % вопросов теста.

Правила начисления баллов озвучиваются на первой лекции, а также размещаются в СДО ГУАП.

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой