Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Костровец Лариса Борисовна
Должность: директор
Дата подписания: 02.10.2026 00:12:43
Уникальный программный ключ:
6882606104c36dbde41c4ab93a65382136a292d6

Приложение 4
к образовательной программе

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.01.ДВ.01.02 Разработка под ОС Android
(индекс, наименование дисциплины в соответствии с учебным планом)

09.03.03 Прикладная информатика
(код, наименование направления подготовки/специальности)

Прикладная информатика в управлении корпоративными информационными системами
(наименование образовательной программы)

Очная форма обучения
(форма обучения)

Год набора – 2026
Донецк

Автор(ы)-составитель(и) РПД:

Лебезова Элла Михайловна, старший преподаватель кафедры информационных технологий

Заведующий кафедрой:

Брадул Наталья Валерьевна, канд. физ.-мат. наук, заведующий кафедрой информационных технологий

Рабочая программа дисциплины Б1.В.01.ДВ.01.02 Разработка под ОС Android одобрена на заседании кафедры информационных технологий администрирования факультета государственной службы и управления Донецкого филиала РАНХиГС.

Протокол № 7 от «05 » марта 2026 г.

СОДЕРЖАНИЕ

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

2. Объем и место дисциплины в структуре образовательной программы

3. Содержание и структура дисциплины

4. Типы оценочных материалов, показатели и критерии их оценивания

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

7. Методические материалы по освоению дисциплины

8. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

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

Дисциплина Б1.В.01.ДВ.01.02 Разработка под ОС Android обеспечивает формирование у обучающихся следующих общепрофессиональных компетенций*:

ОТФ /ТФ и реквизиты ПС *(при наличии)* **	**Код компетенции ****	**Наименование Компетенции ****	**Код индикатора достижения компетенций ****	**Наименование индикатора достижения компетенций ****	**Образовательный результат ****
-	ПК-1.	Способность адаптировать бизнес-процессы заказчика ИС к возможностям типовой ИС в рамках выполнения работ по созданию (модификации) и сопровождению ИС	ПК-1.5.	Утверждает у заказчика предлагаемые изменения его бизнес-процессов для реализации их автоматизации в типовой ИС	**Знает** Основы современных операционных систем; Основы современных СУБД; Устройство и функционирование современных ИС; Основы архитектуры мультиарендного программного обеспечения; Современные стандарты информационного взаимодействия систем; Программные средства и платформы инфраструктуры информационных технологий организаций. **Умеет** Проводить презентации заинтересованным сторонам в рамках

					выполнения работ по созданию (модификации) и сопровождению ИС.

** Дисциплина может формировать компетенцию полностью или частично.*
***Должно соответствовать Приложению 1 к образовательной программе*

2. Объем и место дисциплины в структуре образовательной программы

Общий объем дисциплины:

2,00 з.е., 72 ак.час

Контактная работа обучающихся с преподавателем по видам учебных занятий: 32 ак. час на контактную работу с преподавателем, из них 28 ак. час на практические занятия и 40 ак. час на самостоятельную работу обучающихся.

Б1.В.01.ДВ.01.02 Разработка под ОС Android реализуется в 7-м семестре 4-го курса после после изучения дисциплин:

- Объектно-ориентированное программирование
- Разработка программных приложений на C#/C++
- Анализ данных на Python
- Информационная безопасность
- Информационные системы и технологии
- Базы данных
- Информатика и программирование

.

3. Содержание и структура дисциплины

3.1. Структура дисциплины

Очнаяформа обучения

№ п/п	Наименование тем и (или) разделов		Объем дисциплины, ак.час												Форма текущего контроля успеваемости, промежуточной аттестации
		ВСЕГО	Контактная работа обучающихся с преподавателем по видам учебных занятий									Самостоятельная работа			
			Период теоретического обучения						Период промежуточной аттестации (сессия)						
			Занятия лекционного типа		Занятия семинарского типа		ИК	КСР	КЭ	Каттэк	Контроль	СРкр	СРэк	СР	
			Л	ВЛ	ЛР	ПЗ									
Раздел 1. Разработка программных приложений для ОС Android на Kotlin															
Тема 1	Введение в разработку для ОС Android. Установка необходимого программного обеспечения. Настройка SDK Manager и создание	10	0	0	0	4	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР1

	эмулятора. Создание проекта и запуск на устройстве.														
Тема 2	Основные элементы управления Android Studio. Работа с разметкой и с ресурсами приложения.	10	0	0	0	4	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР1
Тема 3	Активити и интенты. Множественные активности и интенты.	10	0	0	0	4	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР 1
РАЗДЕЛ 2. Архитектура мобильных приложений															
Тема 4	Фоновые потоки AsyncTask. Управление жизненным циклом активити. Базовые навыки работы с меню.	8	0	0	0	2	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР 2
Тема 5	Адаптеры и списки. ListView и ArrayAdapter. Создание	10	0	0	0	4	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР 2

	простейшего спискового приложения.														
Тема 6	Элемент RecyclerView. Локализация проекта Android. Сохранение настроек приложения.	9	0	0	0	4	0	0	0	0	0	0	0	5	Контрольные вопросы, практические занятия, КР 2
Тема 7	Фрагменты. Передача параметров, динамическая смена. DialogFragment.	11	0	0	0	6	0	0	0	0	0	0	0	5	Контрольные вопросы, практические занятия, КР 2
Промежуточная аттестация		4	0	0	0	0	0	0	0	4	0	0	0	0	Зачёт
Итого		**72**	**0**	**0**	**0**	**28**	**0**	**0**	**0**	**4**	**0**	**0**	**0**	**40**	

Используемые сокращения:

Л – лекции - занятия, предусматривающие преимущественную передачу учебной информации обучающимся педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях,).

ВЛ – видео лекции.

ЛР – лабораторные работы.

ПЗ – практические занятия (за исключением лабораторных работ).

ИК – индивидуальные консультации.

КСР – контроль самостоятельной работы

КЭ – консультации перед экзаменом

Каттэк – контактная работа на аттестацию в период экзаменационных сессий

Контроль - контактная работа на аттестацию в период экзаменационных сессий для заочной формы обучения

СРкр – самостоятельная работа на подготовку курсовой работы/ курсового проекта.

СРэк – самостоятельная работа на подготовку к экзамену.

СР – самостоятельная работа в семестре на подготовку к учебным занятиям.

3.2. Содержание дисциплины

РАЗДЕЛ 1. РАЗРАБОТКА ПРОГРАММНЫХ ПРИЛОЖЕНИЙ ДЛЯ ОС ANDROID НА KOTLIN

Тема 1. Введение в разработку для ОС Android. Установка необходимого программного обеспечения. Настройка SDK Manager и создание эмулятора. Создание проекта на языке Java и запуск на устройстве. ПК-1.5.

Изучаются основные понятия мобильной разработки, особенности платформы Android, версии ОС и их фрагментация. Рассматривается установка Android Studio, настройка JDK и SDK Manager. Осваиваются инструменты: эмулятор устройств (AVD) и подключение реального устройства через USB-отладку. Создаётся первый проект на Java, изучается структура Android-приложения (папки res, manifests, java). Выполняется сборка (build) и запуск приложения на эмуляторе и физическом устройстве.

Тема 2. Основные элементы управления Android Studio. Работа с разметкой и административной панелью и базовые настройки. ПК-1.5.

Изучаются основные компоненты интерфейса Android Studio: редактор кода, дизайнер разметки (Layout Editor), окно проекта, Logcat, Build Variants. Рассматриваются типы разметок (ConstraintLayout, LinearLayout, RelativeLayout) и базовые виджеты: TextView, EditText, Button, ImageView. Осваиваются атрибуты виджетов (layout_width, layout_height, padding, margin, gravity). Изучается работа с ресурсами (strings.xml, colors.xml, dimens.xml) и настройка темы и стилей приложения.

Тема 3. Активити и интенты. Множественные активности и интенты. ПК-1.5.

Рассматривается понятие активности (Activity) как основного экрана приложения. Изучается жизненный цикл активности (onCreate, onStart, onResume, onPause, onStop, onDestroy). Осваиваются интенты (Intent) — механизм перехода между активностями и передачи данных. Изучаются явные (explicit) и неявные (implicit) интенты. Рассматривается передача данных через putExtra() и получение через getIntent(). Изучается запуск активности с ожиданием результата (startActivityForResult) и обработка результата (onActivityResult).

РАЗДЕЛ 2. АРХИТЕКТУРА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Тема 4. Фоновые потоки AsyncTask. Управление жизненным циклом активити. Базовые навыки работы с меню. ПК-1.5.

Изучается проблема выполнения длительных операций в UI-потоке и способы её решения. Рассматривается класс AsyncTask (doInBackground, onPreExecute, onPostExecute, onProgressUpdate). Осваивается управление жизненным циклом активности при смене конфигурации (поворот экрана, свёртывание). Изучается создание и обработка событий меню: options menu, context menu и popup menu. Рассматриваются способы сохранения состояния приложения через Bundle и onSaveInstanceState.

Тема 5. Адаптеры и списки. ListView и ArrayAdapter. Создание простейшего спискового приложения. ПК-1.5.

Рассматривается паттерн «Адаптер» применительно к мобильной разработке. Изучается компонент ListView для отображения прокручиваемых списков. Осваивается ArrayAdapter для связывания массива данных с ListView. Изучается кастомизация адаптера (переопределение getView). Рассматривается обработка нажатий на элементы списка через OnItemClickListener. Создаётся приложение-список (например, список студентов, городов, задач) с возможностью добавления и удаления элементов.

Тема 6. Элемент RecyclerView. Локализация проекта Android. Сохранение настроек приложения. ПК-1.5.

Изучается более гибкий и производительный компонент RecyclerView, пришедший на смену ListView. Рассматриваются его составляющие: адаптер (RecyclerView.Adapter), менеджер компоновки (LayoutManager), холдер (ViewHolder). Осваивается горизонтальная и вертикальная прокрутка. Изучается локализация приложения — создание альтернативных ресурсов для разных языков (values-ru, strings.xml). Рассматриваются способы сохранения простых настроек приложения через SharedPreferences (чтение и запись ключей-значений).

Тема 7. Фрагменты. Передача параметров, динамическая смена. DialogFragment. ПК-1.5.

Изучаются фрагменты (Fragment) как переиспользуемые части интерфейса. Рассматривается жизненный цикл фрагмента и его отличие от активности. Осваивается статическое и динамическое добавление фрагментов в активность. Изучается передача параметров между фрагментами и активностью через Bundle и интерфейсы. Рассматривается создание диалоговых окон через DialogFragment. Изучается адаптивный дизайн для планшетов и телефонов с использованием фрагментов.

4. Типы оценочных материалов, показатели и критерии оценивания

4.1. Оценочные материалы по дисциплине Б1.В.01.ДВ.01.02 Разработка под ОС Android входят в состав оценочных материалов по образовательной программе. Совокупность оценочных материалов по всем дисциплинам (модулям) образовательной программы составляет фонд оценочных средств (далее – ФОС). ФОС используется при проведении текущего контроля успеваемости и промежуточной аттестации обучающихся с целью оценивания достижения обучающимися планируемых результатов обучения.

4.2. ФОС разработан как комплекс проверочных заданий различного типа и уровня сложности, включает критерии и шкалы оценивания, а также «ключи» правильных ответов. ФОС формируется как отдельный документ и хранится в электронном виде, доступ к ФОС предоставлен ограниченному кругу лиц.

4.3. Для самостоятельной работы обучающихся при подготовке к текущему контролю успеваемости и промежуточной аттестации в рабочих программах дисциплин размещены типовые проверочные задания, которые можно условно разделить на задания закрытого, комбинированного и открытого типов.

Задания закрытого типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных.

Задания комбинированного типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных и обосновать свой выбор.

Задания открытого типа – это задания, в которых на каждый вопрос должен быть предложен развернутый обоснованный ответ.

В зависимости от типа задания рекомендованы определенная последовательность выполнения и система оценивания выполнения заданий.

4.4. Типы заданий, сценарии выполнения, критерии оценивания

ТИП ЗАДАНИЯ	ИНСТРУКЦИЯ	СЦЕНАРИИ ВЫПОЛНЕНИЯ	КРИТЕРИИ ОЦЕНИВАНИЯ
Задание закрытого типа с выбором одного правильного ответа из нескольких вариантов предложенных	Прочитайте текст, выберите правильный ответ	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа (например, 3 или В).	Ответ считается верным, если правильно указана цифра или буква
Задание закрытого типа на установление соответствия	Прочитайте текст и установите соответствие	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидаются пары элементов. 2. Внимательно прочитать оба списка: список 1 – вопросы, утверждения, факты, понятия и т.д.; список 2 – утверждения, свойства объектов и т.д. 3. Сопоставить элементы списка 1 с элементами списка 2, сформировать пары элементов. 4. Записать попарно буквы и цифры (в зависимости от задания) вариантов ответа (например, А1 или Б4).	Ответ считается верным, если правильно указаны цифры или буквы

Задание закрытого типа с выбором нескольких правильных ответов из нескольких вариантов предложенных	Прочитайте текст, выберите правильные ответы	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается несколько правильных ответов из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать несколько правильных ответов. 4. Записать только номера (или буквы) выбранного варианта ответа (например, 1 4 или А Г).	Ответ считается верным, если правильно установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого)
Задание закрытого типа на установление последовательнос ти	Прочитайте текст и установите последовательность	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается последовательность элементов. 2. Внимательно прочитать предложенные варианты ответа. 3. Построить верную последовательность из предложенных элементов. 4. Записать буквы/цифры (в зависимости от задания) вариантов ответа в нужной последовательности (например, БВА или 135).	Ответ считается верным, если правильно указана вся последовательность цифр

Задание комбинированного типа с выбором одного правильного ответа из предложенных и обоснованием выбора	Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные варианты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа. 5. Записать аргументы, обосновывающие выбор ответа (например, 4 текст обоснования).	Ответ считается верным, если правильно указана цифра или буква и приведены корректные аргументы, используемые при выборе ответа
Задание открытого типа с развернутым ответом	Прочитайте текст и запишите развернутый обоснованный ответ	1. Внимательно прочитать текст задания и понять суть вопроса. 2. Продумать логику и полноту ответа. 3. Записать ответ, используя четкие компактные формулировки. 4. В случае расчетной задачи, записать решение и ответ	Ответ считается верным: 1. Отсутствие фактических ошибок. 2. Раскрытие объема используемых понятий (полнота ответа). 3. Обоснованность ответа (наличие аргументов). 4. Логическая последовательность излагаемого материала.

4.5. Общая шкала оценивания результатов текущего контроля успеваемости и промежуточной аттестации обучающихся с применением БРС

Оценка по шкале ECTS	**Сумма баллов за все виды учебной деятельности**	**Оценка по государственной шкале**	**Определение**
A	90 – 100	«Отлично»	отличное выполнение с незначительным количеством неточностей
B	80 – 89	«Хорошо»	в целом правильно выполненная работа с незначительным количеством ошибок (до 10%)
C	75 – 79		в целом правильно выполненная работа с незначительным количеством ошибок (до 15%)
D	70 – 74	«Удовлетворительно»	неплохо, но со значительным количеством недостатков
E	60 – 69		выполнение удовлетворяет минимальные критерии
FX	35 – 59	«Не удовлетворительно»	с возможностью повторной сдачи
F	0 – 34		с обязательным повторным изучением дисциплины (выставляется комиссией)

Соотношение баллов за текущий контроль успеваемости и промежуточную аттестацию, а также повторную промежуточную аттестацию:

Максимальная сумма баллов за текущий контроль успеваемости	Максимальная сумма баллов за промежуточную аттестацию	Максимальная итоговая балльная оценка	Максимальная сумма баллов за повторную промежуточную аттестацию
100 баллов	100 баллов	100 баллов	100 баллов

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

5.1. В ходе реализации дисциплины Б1.В.01.ДВ.01.02 Разработка под ОС Android используются следующие формы текущего контроля успеваемости обучающихся (в том числе, задания к контрольным точкам):

Контрольные вопросы для проведения опроса, задания открытого типа на практических занятиях, контрольные задания

Таблица 5.1.

Распределение баллов по видам учебной деятельности (БРС)

Раздел/Темы	Формы текущего контроля		КЗР
	УО	ПЗ	
Р-1. / Т-1	3	7	15
Р-1. / Т-2	3	7	
Р-1. / Т-3	3	7	
Р-2. / Т-4	3	7	15
Р-2. / Т-5	3	7	
Р-2. / Т-6	3	7	
Р-2. / Т-7	3	7	
Итого: 100 б	18	7	30

УО – устный опрос;
ТЗ – тестовое задание;
КЗ – контрольные задания;
ПЗ – практическое занятие;
Д – доклад;
КЗР – контрольные работы по разделу.

Критерии оценивания опроса:

Балы	Описание критерия
3	Обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка.
2	Обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
1	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
0	Обучающийся обнаруживает незнание вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

0* - в журнал академической группы не выставляется

Критерии оценивания практических занятий:

Балы	Описание критерия	
3	Свыше 90% правильных ответов.	Обучающийся демонстрирует глубокое познание в освоенном материале.
2	Свыше 70% правильных ответов.	Обучающимся материал освоен полностью, без существенных ошибок.
1	Реализовано более 50% поставленных задач	Обучающимся материал освоен не полностью, имеются значительные пробелы в знаниях.
0	Реализовано менее 30% поставленных задач.	Обучающимся материал не освоен, знания обучающегося ниже базового уровня.

0* - в журнал академической группы не выставляется

Критерии оценивания контрольных заданий:

Балы	Описание критерия
12-15	Обучающимся задание выполнено без ошибок и в полном объеме.
8-11	Обучающимся в целом задание выполнено, имеются отдельные неточности или недостаточно полные ответы, не содержащие ошибок.
5-7	Обучающимся допущены отдельные ошибки при выполнении задания
0-4	У обучающегося отсутствуют ответы на большинство вопросов задачи, задание не выполнено или выполнено не верно.

0* - в журнал академической группы не выставляется

5.2. Типовые оценочные материалы для текущего контроля успеваемости обучающихся (вне контрольных работ):

РАЗДЕЛ 1. РАЗРАБОТКА ПРОГРАММНЫХ ПРИЛОЖЕНИЙ ДЛЯ ОС ANDROID НА KOTLIN

Тема 1. Введение в разработку для ОС Android. Установка необходимого программного обеспечения. Настройка SDK Manager и создание эмулятора. Создание проекта на языке Java и запуск на устройстве.

Контрольные вопросы:

1. Что такое Android SDK и для чего он нужен?
2. Какие компоненты входят в состав Android Studio?
3. Как включить режим разработчика и USB-отладку на реальном устройстве?
4. Что такое AVD (Android Virtual Device) и как его создать?
5. Из каких основных папок состоит проект Android?
6. Что содержит файл AndroidManifest.xml?
7. Как выполнить сборку (build) и запуск приложения на эмуляторе?

Практические задания:

1. Установить Android Studio, JDK и настроить SDK Manager (установить SDK Platform для последней стабильной версии Android).
2. Создать эмулятор устройства (AVD) с параметрами: Pixel 4, Android API 33+.
3. Создать новый проект с именем «FirstApp» на языке Java с пустой активностью (Empty Views Activity).
4. Изучить структуру проекта: папки java (MainActivity), res/layout (activity_main.xml), res/values, manifests.
5. Запустить приложение на эмуляторе и убедиться, что отображается фраза «Hello World!».
6. Подключить реальное Android-устройство через USB-отладку и запустить приложение на нём.
7. Модифицировать приложение: изменить текст «Hello World!» на «Моё первое приложение» через файл strings.xml.

Тема 2. Основные элементы управления Android Studio. Работа с разметкой и административной панелью и базовые настройки.

Контрольные вопросы:

1. Какие основные типы разметок (layouts) существуют в Android?
2. Чем отличается ConstraintLayout от LinearLayout?
3. Какие атрибуты отвечают за ширину и высоту элемента?
4. Для чего используются файлы strings.xml, colors.xml, dimens.xml?
5. Что такое ID виджета и как к нему обратиться в коде?
6. Какие основные виджеты используются для ввода и отображения текста?
7. Как настроить обработчик нажатия на кнопку (Button)?

Практические задания:

1. Создать новый проект «UILayouts». В activity_main.xml создать интерфейс с использованием LinearLayout (вертикальная ориентация), содержащий:
   - TextView («Введите имя»)
   - EditText (поле ввода)
   - Button («Отправить»)
   - TextView для вывода результата
2. Повторить тот же интерфейс с использованием ConstraintLayout.
3. Вынести все строковые литералы в strings.xml, цвета в colors.xml.
4. Настроить обработчик нажатия на кнопку: при нажатии текст из EditText выводится в TextView результата.

5. Добавить вторую кнопку «Очистить», которая очищает поле ввода и результат.

6. Настроить тему приложения (изменить цветовую схему в themes.xml).

Тема 3. Активити и интенты. Множественные активности и интенты

Контрольные вопросы:

1. Что такое активность (Activity) в Android?
2. Перечислите основные методы жизненного цикла активности в порядке их вызова.
3. Что такое Intent? Чем отличаются явные и неявные интенты?
4. Как передать данные из одной активности в другую с помощью putExtra()?
5. Как получить переданные данные в целевой активности?
6. Что такое startActivityForResult() и для чего он используется?
7. Как обработать результат возврата из дочерней активности?

Практические задания:

1. Создать проект «IntentsDemo» с двумя активностями: MainActivity и SecondActivity.
2. В MainActivity разместить:
   - EditText для ввода текста
   - Button «Перейти на второй экран»
   - TextView для отображения ответа
3. При нажатии на кнопку передать введённый текст во SecondActivity через Intent (putExtra).
4. Во SecondActivity отобразить полученный текст и добавить кнопку «Назад».
5. Реализовать возврат данных из SecondActivity в MainActivity: пользователь вводит ответ в EditText во второй активности, нажимает «Ответить», и этот текст отображается в TextView первой активности.
6. Добавить неявный Intent: кнопка «Открыть браузер» — открывает сайт (например, https://developer.android.com).

РАЗДЕЛ 2. АРХИТЕКТУРА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Тема 4. Фоновые потоки AsyncTask. Управление жизненным циклом активити. Базовые навыки работы с меню.

Контрольные вопросы:

1. Почему нельзя выполнять длительные операции в UI-потоке?
2. Что такое AsyncTask и какие методы он содержит?
3. В каком методе AsyncTask выполняются фоновые операции?
4. Как обновить интерфейс после завершения фоновой задачи?
5. Какие проблемы могут возникнуть при использовании AsyncTask?
6. Как сохранить состояние активности при повороте экрана?
7. Какие виды меню существуют в Android (OptionsMenu, ContextMenu, PopupMenu)?

Практические занятия:

1. Создать проект «AsyncTaskDemo». Добавить ProgressBar (невидимый) и TextView.
2. Создать класс-наследник AsyncTask, который в doInBackground имитирует длительную операцию (Thread.sleep(3000)).
3. В onPreExecute показать ProgressBar, в onPostExecute скрыть его и вывести текст «Завершено».
4. Добавить в активность OptionsMenu (три точки) с пунктами: «Старт», «О программе».
5. Реализовать обработку выбора пункта меню: «Старт» запускает AsyncTask, «О программе» показывает Toast.
6. Сохранять состояние при повороте экрана: в Bundle сохранять текст из TextView и восстанавливать его в onCreate.

Тема 5. Адаптеры и списки. ListView и ArrayAdapter. Создание простейшего спискового приложения.

Контрольные вопросы:

1. Что такое адаптер (Adapter) в Android?
2. Для чего используется ListView?
3. Как связать массив данных с ListView с помощью ArrayAdapter?
4. Что делает метод getView() в кастомном адаптере?
5. Как обработать нажатие на элемент списка?
6. Какие есть способы динамического добавления/удаления элементов из списка?
7. Что такое ViewHolder и зачем он нужен?

Практические занятия:

1. Создать проект «SimpleList» с ListView.
2. Создать массив строк (например, список городов: «Москва», «Санкт-Петербург», «Казань», «Новосибирск», «Екатеринбург»).
3. С помощью ArrayAdapter вывести этот массив в ListView.

4. Добавить обработчик OnItemClickListener: при нажатии на элемент показывать Toast с названием выбранного города.

5. Добавить кнопку «Добавить город» и EditText — новый город добавляется в список (в массив и уведомление адаптера об изменении).

6. Добавить долгое нажатие (OnItemLongClickListener) для удаления элемента из списка.

7. Реализовать кастомный адаптер: каждый элемент списка должен содержать TextView (название) и ImageView (иконка).

Тема 6. Элемент RecyclerView. Локализация проекта Android. Сохранение настроек приложения.

Контрольные вопросы:

– В чем преимущество RecyclerView перед ListView?

– Из каких трёх основных компонентов состоит RecyclerView?

– Что такое ViewHolder в RecyclerView и зачем он нужен?

– Что делает LayoutManager? Какие виды LayoutManager существуют?

– Как добавить горизонтальную прокрутку в RecyclerView?

– Как добавить поддержку нескольких языков в Android-приложение?

– Для чего используется SharedPreferences? Как записать и прочитать данные?

Практические занятия:

1. Создать проект «RecyclerDemo». Отобразить список имён студентов с помощью RecyclerView (вертикальный LinearLayoutManager).

2. Создать кастомный адаптер и ViewHolder. Каждый элемент содержит: аватар (ImageView), имя (TextView), возраст (TextView).

3. Реализовать горизонтальный RecyclerView для отображения категорий (например, «Игры», «Музыка», «Кино»).

4. Добавить локализацию: создать папку values-ru с strings.xml. Перевести все строки приложения на русский и английский языки. Добавить возможность переключения языка через меню (или через настройки системы).

5. Сохранять последнюю выбранную позицию в списке с помощью SharedPreferences. При следующем запуске приложения выделять этот элемент (например, менять фон).

Тема 7. Фрагменты. Передача параметров, динамическая смена. DialogFragment.

Контрольные вопросы:

1. Что такое фрагмент (Fragment)? Чем он отличается от активности?
2. Каков жизненный цикл фрагмента?
3. Как добавить фрагмент статически (через XML) и динамически (через код)?
4. Как передать данные из фрагмента в активность?
5. Как передать данные из активности во фрагмент?
6. Что такое DialogFragment? Какие преимущества у него перед обычным диалогом?
7. Как создать адаптивный интерфейс для телефонов и планшетов с использованием фрагментов?

Практические занятия:

1. Создать проект «FragmentsDemo» с одной активностью MainActivity.
2. Создать два фрагмента: FirstFragment (с EditText и кнопкой) и SecondFragment (с TextView).
3. Разместить фрагменты статически (два фрагмента в одной активности) при горизонтальной ориентации и динамически (замена) при вертикальной.
4. Реализовать передачу данных: при нажатии на кнопку в FirstFragment текст из EditText передаётся во SecondFragment (через интерфейс или FragmentManager).
5. Создать DialogFragment с полем ввода и кнопками «OK» и «Отмена». При нажатии на кнопку в активности показать этот диалог, а после ввода текста отобразить его в активности.
6. Реализовать динамическую замену фрагментов: два фрагмента, которые меняются при нажатии на кнопки внизу экрана (как табы).

Дополнительное задание повышенной сложности (для самостоятельной работы)

Разработать приложение «Заметки» (Notes), объединяющее все изученные темы:

- Список заметок (RecyclerView)
- Добавление новой заметки через DialogFragment (заголовок + текст)
- Редактирование и удаление заметок через контекстное меню
- Сохранение заметок в SharedPreferences или локальную БД (SQLite)

- Локализация (русский/английский)
- Переход на отдельный экран для просмотра полной заметки (Intent + Activity или Fragment)
- Сохранение состояния при повороте экрана

5.3. Один или несколько тематических блоков дисциплины завершаются контрольной работой по разделу (далее – КР). Текущий контроль успеваемости по дисциплине предусматривает не менее 2 (двух) и не более 10 (десяти) КР в течение периода освоения дисциплины.

Максимальное количество баллов за любой тип работ в рамках КР составляет 100 (сто) баллов.

Распределение весовых коэффициентов по КР в рамках текущего контроля успеваемости по дисциплине и формулы расчета:

Наименование контрольной работы	Максимальное количество баллов за работу в рамках КР, которое может набрать студент	Коэффициент веса контрольной работы	Результат контрольной работы, участвующий в формировании итоговой балльной оценки по дисциплине
КР 1	100	0,15	15
КР 2	100	0,15	15
Итого:	x	0,30	30

Формула расчета результата контрольной работы:

Результат контрольной работы = Количество баллов за работу в рамках КР X Коэффициент веса контрольной работы.

5.4. Формы текущего контроля успеваемости обучающихся в рамках КР и типовые оценочные материалы:

КР-1

Раздел 1. Разработка программных приложений для ОС Android на Kotlin

Теоретические вопросы:

1. Разработчик создал приложение, которое при повороте экрана теряет введённые пользователем данные в EditText. Опишите механизм

сохранения состояния активности в Android. Какие методы жизненного цикла за это отвечают?

2. В чём разница между match_parent и wrap_content? Приведите примеры использования каждого атрибута.

3. Что произойдёт, если в UI-потоке выполнить операцию Thread.sleep(5000)? Как правильно выполнить длительную операцию в Android?

4. Создано приложение с двумя активностями. При переходе из MainActivity во SecondActivity через startActivity() и последующем нажатии кнопки «Назад» какой метод жизненного цикла вызывается в MainActivity?

5. Опишите структуру XML-файла разметки для ConstraintLayout, содержащего два элемента: TextView в центре экрана и Button в правом нижнем углу.

6. Что такое неявный Intent? Приведите пример открытия веб-страницы в браузере по умолчанию. Как проверить, существует ли приложение, способное обработать этот Intent?

7. Перечислите основные папки в ресурсах (res/) Android-проекта и укажите, какие типы файлов в них хранятся.

Практическое задание:

Условие:

Разработать приложение «Конвертер валют», которое соответствует следующим требованиям.

Требования:

Интерфейс (LinearLayout или ConstraintLayout):

Поле для ввода суммы (EditText, тип ввода — число).

Spinner (выпадающий список) для выбора исходной валюты (USD, EUR, RUB).

Spinner для выбора целевой валюты (USD, EUR, RUB).

Кнопка «Конвертировать».

TextView для вывода результата.

Кнопка «Очистить».

Логика конвертации (фиксированные курсы):

1 USD = 0.92 EUR, 1 USD = 95 RUB.

1 EUR = 1.09 USD, 1 EUR = 103 RUB.

1 RUB = 0.0105 USD, 1 RUB = 0.0097 EUR.

Дополнительные требования:

При повороте экрана введённые данные и результат должны сохраняться.

При вводе некорректных данных (пустое поле, не число) показывать Toast с ошибкой.

Результат выполнения: предоставить архив проекта (или скриншоты + код основных файлов).

КР-2

Раздел 2. Архитектура мобильных приложений

Теоретические вопросы:

1. В чём преимущество RecyclerView перед ListView? Назовите три основных компонента RecyclerView и их назначение.
2. Что произойдёт, если AsyncTask будет ссылаться на активность и активность будет уничтожена (поворот экрана)? Как избежать утечки памяти?
3. Опишите процесс локализации Android-приложения на русский язык. Какие папки и файлы необходимо создать?
4. Как сохранить строковое значение и целое число в SharedPreferences? Приведите пример кода для записи и последующего чтения.
5. Чем отличается статическое добавление фрагмента от динамического? В каких случаях применяется каждый способ?
6. Как передать данные из фрагмента в активность? Опишите два способа.
7. Что такое DialogFragment? В чём его преимущество перед AlertDialog.Builder?

Практическое задание:

Условие:

Разработать приложение «Список фильмов с оценками» (Movie Rating List).

Требования:

Главный экран:

RecyclerView для отображения списка фильмов.

Каждый элемент содержит: название фильма, жанр, рейтинг (1–5 звёзд).

FloatingActionButton для добавления нового фильма.

Добавление фильма:

Использовать DialogFragment.

Поля: название (EditText), жанр (Spinner: «Боевик», «Комедия», «Драма», «Фантастика»), рейтинг (RatingBar).

Кнопка «Сохранить».

Удаление:

По долгому нажатию на элемент списка удалить фильм с подтверждением.

Редактирование:

По короткому нажатию открывать тот же DialogFragment с предзаполненными данными для редактирования.

Сохранение данных:

Список фильмов сохраняется в SharedPreferences (сериализация через Gson или ручное преобразование).

Дополнительное требование:

В OptionsMenu добавить пункт «Сортировка по рейтингу» (по убыванию).

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

6.1. Промежуточная аттестация по дисциплине «Разработка под ОС Android» проводится в форме зачёта в седьмом семестре в письменной форме. Обучающийся получает два теоретических вопроса и одно практическое задание.

6.2. Типовые оценочные материалы промежуточной аттестации.

Вопросы к зачёту:

Раздел 1. Основы разработки для Android на Kotlin (темы 1–3)

1. Что такое Android SDK? Какие компоненты входят в его состав?
2. Структура Android-проекта: назначение папок java, res/layout, res/values, manifests.
3. Что такое AndroidManifest.xml? Какие обязательные элементы он содержит?
4. Виды разметок (layouts) в Android: LinearLayout, ConstraintLayout, RelativeLayout. Сравнение.
5. Основные виджеты: TextView, EditText, Button, ImageView. Основные атрибуты.
6. Что такое активность (Activity)? Жизненный цикл активности (все методы).
7. Что такое Intent? Явные и неявные интенты: определение, примеры.
8. Передача данных между активностями через putExtra() и getIntent().
9. Запуск активности с ожиданием результата: startActivityForResult() и onActivityResult().
10. Сохранение состояния активности при повороте экрана: onSaveInstanceState() и Bundle.

Раздел 2. Архитектура мобильных приложений (темы 4–7)

11. Почему нельзя выполнять длительные операции в UI-потоке? Последствия.
12. AsyncTask: назначение, основные методы (doInBackground, onPreExecute, onPostExecute).
13. Какие проблемы существуют у AsyncTask? (deprecated, утечки памяти).

14. Виды меню в Android: OptionsMenu, ContextMenu, PopupMenu. Примеры использования.

15. Что такое адаптер (Adapter)? Роль адаптера при отображении списков.

16. ListView и ArrayAdapter: принцип работы, связывание данных.

17. Паттерн ViewHolder: зачем нужен, как реализуется.

18. RecyclerView: преимущества перед ListView. Три основных компонента.

19. LayoutManager: виды (Linear, Grid, StaggeredGrid), как задать горизонтальную прокрутку.

20. Локализация приложения: создание альтернативных ресурсов (values-ru).

21. SharedPreferences: назначение, запись и чтение данных.

22. Фрагменты (Fragment): определение, отличие от активности.

23. Жизненный цикл фрагмента в сравнении с активностью.

24. Статическое и динамическое добавление фрагментов.

25. Передача данных между фрагментом и активностью (интерфейсы, Bundle).

26. DialogFragment: преимущества перед стандартным AlertDialog.

27. Адаптивный дизайн: использование фрагментов для телефонов и планшетов.

28. Современные альтернативы AsyncTask (ThreadPool, Coroutines, RxJava — общее понятие).

29. Gradle в Android: зависимость, версии SDK, компиляция.

30. Отладка приложений: Logcat, точки останова, USB-отладка.

Пример практического задания

Условие: Разработать приложение «Список дел с приоритетами» (ToDo List with Priorities), удовлетворяющее требованиям ниже.

Вариант выдачи: Студент получает индивидуальное задание с дополнительным условием (например, «добавить поиск по задачам» или «сделать тёмную тему»). Ниже приведён базовый вариант.

Базовое задание

Функциональные требования:

1. Главный экран содержит:
   - RecyclerView для отображения списка задач.
   - Каждая задача отображает: название, приоритет (высокий/средний/низкий), статус выполнения (CheckBox).
   - FloatingActionButton для добавления новой задачи.

2. Добавление/редактирование задачи реализуется через DialogFragment:

- Поле ввода названия задачи (EditText).
- Выбор приоритета (Spinner или RadioGroup).
- Кнопки «Сохранить» и «Отмена».

3. Удаление задачи:

- По долгому нажатию на задачу появляется диалог подтверждения удаления.

4. Изменение статуса:

- При нажатии на CheckBox задача перечёркивается (или меняет цвет фона) и сохраняется статус.

5. Сохранение данных:

- Все задачи сохраняются в SharedPreferences (или SQLite — по уровню сложности).
- При перезапуске приложения список восстанавливается.

6. Сортировка (дополнительное требование к зачёту):

- В OptionsMenu добавить пункты: «По приоритету», «По дате добавления».

Минимальные требования к интерфейсу:

- Использовать Material Design (CardView для элементов списка).
- Адаптация под тёмную и светлую тему (по выбору пользователя или системная).

6.3. Критерии и шкала оценивания на основе БРС.

Соответствие государственной шкалы оценивания академической успеваемости и шкалы ECTS при зачёте

Оценка по шкале ECTS	**Сумма баллов за все виды учебной деятельности**	**Оценка по государственной шкале**	**Определение**
A	90 – 100	«Отлично»	отличное выполнение с незначительным количеством неточностей
B	80 – 89	«Хорошо»	в целом правильно выполненная работа с незначительным количеством ошибок (до 10%)

C	75 – 79		в целом правильно выполненная работа с незначительным количеством ошибок (до 15%)
D	70 – 74	«Удовлетворительно»	неплохо, но со значительным количеством недостатков
E	60 – 69		выполнение удовлетворяет минимальные критерии
FX	35 – 59	«Не удовлетворительно»	с возможностью повторной сдачи
F	0 – 34		с обязательным повторным изучением дисциплины (выставляется комиссией)

6.4. Описание дополнительных материалов и оборудования, необходимых для выполнения проверочных заданий

Компьютер с операционной системой RedOS или Windows с устойчивым Интернет-соединением для работы с удалёнными web-сервисами, программные продукты с открытой лицензией: OpenServer/Denver/XAMPP, Android Studio, эмулятор OS Android.

7. Методические материалы по освоению дисциплины

Получение углубленных знаний по изучаемой дисциплине достигается за счет дополнительных часов к аудиторной работе самостоятельной работы студентов. Выделяемые часы целесообразно использовать для знакомства с дополнительной научной литературой по проблематике дисциплины, анализа научных концепций и современных подходов к осмыслению рассматриваемых проблем. К самостоятельному виду работы студентов относится работа в библиотеках, в электронных поисковых системах и т.п. по сбору материалов, необходимых для проведения практических занятий или выполнения конкретных заданий преподавателя по изучаемым темам. Студенты могут установить диалог с преподавателем, получать консультации по выполнению заданий. В качестве оценочных средств на протяжении семестра используются практические задания.

Обучение по дисциплине «Разработка под ОС Android» предполагает изучение курса на аудиторных занятиях (лекции, практические занятия) и самостоятельную работу студентов. Практические занятия дисциплины предполагают их проведение в различных формах с целью выявления полученных знаний, умений, навыков и компетенций с проведением контрольных мероприятий. С целью обеспечения успешного обучения студент должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

– знакомит с новым учебным материалом;
– разъясняет учебные элементы, трудные для понимания;
– систематизирует учебный материал;
– ориентирует в учебном процессе.

Работа обучающегося на лекции:

Слушание и запись лекций – сложный вид вузовской аудиторной работы. Внимательное слушание и конспектирование лекций предполагает интенсивную умственную деятельность обучающегося. Краткие записи лекций, их конспектирование помогает усвоить учебный материал. Конспект является полезным тогда, когда записано самое существенное, основное и сделано это самим обучающимся.

Подготовка к практическим занятиям:

Подготовку к каждому практическому занятию каждый обучающийся должен начать с ознакомления с планом, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованную к данной теме. Если программой дисциплины предусмотрено выполнение практического задания, то его необходимо выполнить с учетом предложенной инструкции. Все новые понятия по изучаемой теме необходимо внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности обучающегося свободно ответить на теоретические вопросы практического занятия, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий и контрольных работ.

Структура практического занятия:

В зависимости от содержания и количества отведенного времени на изучение каждой темы может практическое занятие состоять из четырех-пяти частей:

1. Устный опрос.

2. Обсуждение теоретических вопросов, определенных программой дисциплины.

3. Выполнение практических заданий с последующим разбором полученных результатов или обсуждение практического задания, выполненного дома.

4. Подведение итогов занятия.

Работа с литературными источниками:

В процессе подготовки к практическим занятиям, обучающимся необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме. Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем, что позволяет обучающимся проявить свою индивидуальность в рамках выступления на занятиях, выявить широкий спектр мнений по изучаемой проблеме.

8. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет

8.1. Основная литература

1. Петросян, Л. Э. Разработка мобильных приложений на Kotlin : учебное пособие / Л. Э. Петросян, Н. А. Приходько. — Москва : РТУ МИРЭА, 2024. — 101 с. — ISBN 978-5-7339-2215-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/421091 (дата обращения: 17.05.2026). — Режим доступа: для авториз. пользователей.

2. Митяков, Е. С. Современная разработка мобильных приложений на языке Kotlin: Практикум : учебное пособие / Е. С. Митяков, М. С. Поздняков. — Москва : РТУ МИРЭА, 2025. — 117 с. — ISBN 978-5-7339-2726-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/513041 (дата обращения: 17.05.2026). — Режим доступа: для авториз. пользователей.

3. Петросян, Л. Э. Разработка мобильных приложений на языке Kotlin : учебное пособие для вузов / Л. Э. Петросян, К. В. Гусев. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 104 с. — ISBN 978-5-507-52328-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/448577 (дата обращения: 17.05.2026). — Режим доступа: для авториз. пользователей.

8.2. Дополнительная литература

4. Льюис, Ш. Нативная разработка мобильных приложений : руководство / Ш. Льюис, М. Данн ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2020. — 376 с. — ISBN 978-5-97060-845-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/179491 (дата обращения: 17.05.2026). — Режим доступа: для авториз. пользователей..

5. Чернов, Е. А. Проектирование и разработка мобильных приложений. Часть 1 : учебное пособие / Е. А. Чернов. — Москва : РТУ МИРЭА, 2025. — 69 с. — ISBN 978-5-7339-2652-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/507485 (дата обращения: 17.05.2026). — Режим доступа: для авториз. пользователей..

8.3.Нормативные правовые документы и иная правовая информация

1. Конституция Российской Федерации. – Текст : электронный // Сайт Президента Российской Федерации. – URL: http://www.kremlin.ru/acts/constitution

8.4 Интернет-ресурсы

1. Информационно-правовой портал ГАРАНТ.РУ. – URL: https://www.garant.ru/
2. Научная электронная библиотека eLIBRARY.RU. – URL: https://elibrary.ru/
3. Научная электронная библиотека «КиберЛенинка». – URL: https://cyberleninka.ru
4. Электронно-библиотечная система «Лань». – URL: http://e.lanbook.com
5. Документация по Java – URL: https://w3schoolsrus.github.io/java/index.html
6. База знаний по разработке на Android Studio – URL: https://metanit.com/java/android/1.1.php

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

Материально-техническое обеспечение дисциплины включает в себя:

- лекционные аудитории, оборудованные видеопроекционным оборудованием для презентаций, средствами звуковоспроизведения, экраном;
- помещения для проведения практических занятий, оборудованные учебной мебелью.

Дисциплина поддержана соответствующими программными продуктами с открытой лицензией: OpenServer/Denver/XAMPP, Android Studio, эмулятор OS Android.

Вуз обеспечивает каждого обучающегося рабочим местом в компьютерном классе в соответствии с объемом изучаемых дисциплин, обеспечивает выход в сеть Интернет.

Помещения для самостоятельной работы обучающихся включают следующую оснащенность: столы аудиторные, стулья, доски аудиторные, компьютеры с подключением к локальной сети института (для компьютерных аудиторий) и Интернет. Для изучения учебной дисциплины используются автоматизированная библиотечная информационная система и электронные библиотечные системы.