

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Курганский государственный университет»

Кафедра «Программное обеспечение автоматизированных систем»



УТВЕРЖДАЮ:

Первый проректор

Т.Р. Змызгова

«31» августа 2021 г.

Рабочая программа учебной дисциплины

РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

образовательной программы высшего образования –
программы бакалавриата

09.03.03 Прикладная информатика

Направленность

Интеллектуальные информационные системы и технологии


Форма обучения: очная, заочная

Курган 2021

Рабочая программа дисциплины «Разработка мобильных приложений» составлена в соответствии с учебным планом программы бакалавриата: «Прикладная информатика», (Интеллектуальные информационные системы и технологии), утвержденным для очной и заочной форм обучения 30 августа 2021 г.

Рабочая программа дисциплины одобрена на заседании кафедры «Программное обеспечение автоматизированных систем» 30.08.2020 года, протокол № 1.

Рабочую программу составил:
доцент кафедры ПОАС



_____ /А.М. Семахин/

Согласовано:

Заведующий кафедрой
«Программное обеспечение
автоматизированных систем»


_____ /В. К. Волк/

Специалист
по учебно-методической работе
Учебно-методического отдела


_____ /Г. В Казанкова/

Начальник
Управления образовательной
деятельности


_____ /С.Н. Синицын/

1. ОБЪЕМ ДИСЦИПЛИНЫ

Всего: 3 зачетных единиц трудоемкости (108 академических часа)

Очная форма обучения

Вид учебной работы	На всю дисциплину	Семестр
		7
Аудиторные занятия (контактная работа с преподавателем), всего часов	48	48
в том числе:		
Лекции	16	16
Лабораторные работы	32	32
Аудиторные занятия в интерактивной форме, часов	-	-
Самостоятельная работа, всего часов	60	60
в том числе:		
Контрольная работа	18	18
Подготовка к зачёту	18	18
Другие виды самостоятельной работы (самостоятельное изучение тем (разделов) дисциплины)	24	24
Вид промежуточной аттестации	Зачёт	Зачёт
Общая трудоемкость дисциплины и трудоемкость по семестрам, часов	108	108

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина «Разработка мобильных приложений» относится к вариативной части, формируемой участниками образовательных отношений, блока 1.

Изучение дисциплины базируется на результатах обучения, сформированных при изучении следующих дисциплин:

- Информатика.
- Основы программирования.
- Алгоритмы и структуры данных.
- Технологии разработки web-приложений.

Результаты изучения дисциплины используются при освоении профильных дисциплин, включенных в модули «Программное и аппаратное обеспечение информационно-коммуникационных систем», «Технологии разработки программных систем» и «Системы интеллектуальной обработки данных».

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Целью освоения дисциплины «Разработки мобильных приложений» является формирование знаний и практических навыков разработке мобильных приложений.

Задачи дисциплины:

1) изучение:

- архитектура iOS/Android и нативные API;
- архитектура кроссплатформенных фреймворков;
- архитектура мобильных приложений;
- процесс разработки мобильных приложений и документация;
- создание интерфейсов с помощью макетов и виджетов;
- базы данных и библиотека компонентов архитектуры Room;
- библиотека WorkManager;
- Веб-серфинг и WebView.

2) практическое освоение:

- платформа Android Studio;
- язык программирования Kotlin;
- Git;
- библиотеки Glide (используется для загрузки изображений) и Retrofit (применяется для получения данных из сети);

Компетенции, формируемые в результате освоения дисциплины:

- способность разрабатывать и проводить установку, настройку, оптимизацию функционирования сетевого и прикладного программного обеспечения (ПК-5);

- владение навыками использования операционных систем, сетевых технологий, систем управления базами данных. Способность настраивать, эксплуатировать и сопровождать информационные системы и сервис (ПК-8);

- способность проводить тестирование компонентов программного обеспечения информационных систем, осуществлять разработку, отладку, проверку работоспособности и рефакторинг программного кода (ПК-10).

В результате изучения дисциплины обучающийся должен:

Знать:

- установку, настройку, оптимизацию функционирования сетевого и прикладного программного обеспечения (ПК-5);

- операционные системы, сетевые технологии, системы управления базами данных, настройку, эксплуатацию и сопровождение информационных систем и сервисы (ПК-8);

- тестирование компонентов программного обеспечения информационных систем, методику разработки, отладку, проверку работоспособности и рефакторинг программного кода (ПК-10).

Уметь:

- разрабатывать и проводить установку, настройку, оптимизацию функционирования сетевого и прикладного программного обеспечения (ПК-5);

- настраивать, эксплуатировать и сопровождать информационные системы и сервис (ПК-8);

- проводить тестирование компонентов программного обеспечения информационных систем, осуществлять разработку, отладку, проверку работоспособности и рефакторинг программного кода (ПК-10).

Владеть:

- навыками разработки и установки, настройки, оптимизации функционирования сетевого и прикладного программного обеспечения (ПК-5);
- навыками использования операционных систем, сетевых технологий, систем управления базами данных, настройки, эксплуатации и сопровождения информационных систем и сервисов (ПК-8);
- навыками тестирования компонентов программного обеспечения информационных систем, осуществления разработки, отладки, проверки работоспособности и рефакторинг программного кода (ПК-10).

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Учебно-тематический план. Очная форма обучения. Семестр 7

Рубеж	Номер раздела, темы	Наименование раздела, темы	Количество часов контактной работы с преподавателем	
			Лекции	Лабораторные работы
Рубеж 1	1	Архитектура и базовые сведения о платформе Android	2	2
	2	Основы разработки мобильных приложений	2	2
	3	Android и модель MVC	2	4
	4	Неявные интенты, интенты при работе с камерой	2	4
		Рубежный контроль №1	-	2
Рубеж 2	5	Создание интерфейсов с использованием макетов и виджетов	2	4
	6	Базы данных и Room Library	2	4
	7	Модульное тестирование и воспроизведение звуков	2	4
	8	Классы Looper и HandlerThread	2	4
		Рубежный контроль №2	-	2
Всего:			16	32

4.2 Содержание лекционных занятий

Семестр 7

Тема 1. Архитектура и базовые сведения о платформе Android

Нативные и кроссплатформенные инструменты разработки. Архитектура iOS/Android. Нативный iOS. Нативный Android. Нативный Windows UWP. Архитектура Android. Уровень ядра. Уровень библиотек. Уровень каркаса приложений. Компоненты Android-приложения. Activity. Service. Broadcast Receiver. Архитектуры кроссплатформенных фреймворков: PhoneGap, ReactNative, Qt, Flutter, Xamarin, Xamarin.Forms.

Тема 2. Основы разработки мобильных приложений

Создание проекта Android. Навигация в Android Studio. Создание макета пользовательского интерфейса. Иерархия представления. Атрибуты виджетов. Создание строковых ресурсов. Предварительный просмотр макета. От разметки XML к объектам View. Ресурсы и идентификаторы ресурсов. Разработка виджетов. Установка ссылок на виджеты. Назначение слушателей. Уведомления. Выполнение в эмуляторе. Процесс сборки Android-приложений. Инструменты сборки.

Тема 3. Android и модель MVC

Создание нового класса. Архитектура «Модель–Представление–Контроллер» и Android. Преимущества MVC. Обновление уровня представления. Обновление уровня контроллера. Добавление ресурсов в проект. Ссылка на ресурсы в XML. Запуск на устройстве. Многослойный MVVM. Декомпозиция по слоям. Связи внутри слоёв. Связи между слоями. Структуры данных на основе UI.

Тема 4. Неявные интенты, интенты при работе с камерой

Использование неявных интентов. Строение неявного интента. Отправка отчёта. Запрос контакта у Android. Проверка реагирующих activity. Хранилище файлов. Использование FileProvider. Использование интента камеры. Масштабирование и отображение растровых изображений.

Тема 5. Создание интерфейсов с использованием макетов и виджетов

ConstraintLayout. Использование графического инструмента макетов. Освобождение пространства. Добавление виджетов. Внутренние механизмы ConstraintLayout. Редактирование свойств. Динамическое поведение элементов списка. Стили, темы, атрибуты тем.

Тема 6. Базы данных и Room Library

Библиотека компонентов архитектуры Room. Создание базы данных. Отделение сущности. Создание класса базы данных. Определение объекта доступа к данным. Доступ к базе данных с помощью шаблона репозитория. Тестирование запросов. Загрузка тестовых данных. Поток приложения. Фоновые потоки. Использование LiveData.

Тема 7. Модульное тестирование и воспроизведение звуков

Создание объекта SoundPool. Доступ к активам. Воспроизведение звука. Зависимости от тестирования. Создание класса теста. Подготовка теста. Настройка тестируемых объектов. Написание тестов. Взаимодействия тестируемых объектов. Обратные вызовы привязки данных. Выгрузка звуков.

Тема 8. Классы Looper и HandlerThread

Подготовка RecyclerView. Подготовка к загрузке через URL. Множественные загрузки. Создание фонового потока. Запуск и остановка HandlerThread. Сообщения и обработки сообщений. Структура сообщений. Структура обработчика. Использование обработчиков. Прослушивание жизненного цикла представления. Сохранённые фрагменты.

4.3 Лабораторные занятия. Очная форма обучения. Семестр 7

Номер раздела, темы	Наименование раздела, Темы	Наименование лабораторной работы	Норматив времени, час.
1	Архитектура и базовые сведения о платформе Android	Основные принципы создания проектов Android	2
2	Основы разработки мобильных приложений	Разработка игры Tetris	2
3	Android и модель MVC	Разработка приложения для хранения данных	4
4	Неявные интенты, интенты при работе с камерой	Воспроизведение мультимедийного контента	4
		Рубежный контроль №1	2
5	Создание интерфейсов с использованием макетов и виджетов	Нестандартный лаунчер	4
6	Базы данных и Room Library	Загрузка и отображение фотографий	4
7	Модульное тестирование и воспроизведение звуков	Графическое приложение обработки событий касания и создание нестандартных представлений	4
8	Классы Looper и HandlerThread	Приложение с анимацией	4
		Рубежный контроль №2	2
Всего:			32

4.4 Контрольная работа

Контрольная работа посвящена разработке сайта, по вариантам задания, согласно методических рекомендаций, указанных в разделе 8.

4.4.1 Назначение, цели и задачи контрольной работы

Контрольная работа выполняется по вариантам заданий или по теме, предложенной студентом, и согласованной с преподавателем.

В ходе выполнения контрольной работы студент проектирует и реализует мобильное приложение.

Основная учебная цель выполнения контрольной работы – закрепление теоретических знаний, полученных в процессе изучения дисциплины «Разработка мобильных приложений», и приобретение практических навыков в разработке мобильных приложений.

Основные задачи, решаемые студентом в процессе выполнения контрольной работы:

- создание проекта в Android Studio;

- создание пользовательского интерфейса;
- добавление активности, навигации и действий;
- выполнение тест-драйва приложения в эмуляторе;
- оформление документации.

4.4.2 Требования к контрольной работе

4.4.2.1 Требования к функциональным характеристикам

Функциональные требования к приложению:

- роли пользователей: какие уровни доступа должны быть у разных пользователей, например у гостя и авторизованного пользователя;
- форматы данных: как будет реализован обмен данными в приложении;
- интеграция: должно ли приложение поддерживать совместную работу с другими сервисами, например с платежными системами и почтовыми серверами;
- интерфейсы доступа: как приложение будет обмениваться данными с внешними сервисами;
- дополнительные функции: должно ли приложение уметь что-то еще, например работать с файлами или библиотеками шифрования;
- конфигурация и администрирование: с помощью каких элементов администратор будет управлять приложением;
- состав системы: из чего состоит мобильное приложение, то есть экраны, пуш-уведомления, система аутентификации и т.д.

4.4.2.2 Требования к эксплуатационным характеристикам

- Модульность.
- Расширяемость.
- Кроссплатформенность.
- Отказоустойчивость.

4.4.2.3 Требования к программному обеспечению

- Язык программирования Kotlin;
- Платформа Android.

4.4.2.4 Требования к содержанию контрольной работы

К защите контрольной работы должен быть представленны мобильное приложение и пояснительная записка:

- экран загрузки;
- регистрация и авторизация;

- основной экран;
- меню;
- поиск;
- уведомления.

4.4.3 Варианты заданий контрольной работы

- 1 Мобильное приложение «Поиск вакансий».
- 2 Интернет-магазина по продаже смартфонов.
- 3 Мобильное приложение «Cash Organizer»
- 4 Мобильное приложение «Прогноз погоды».
- 5 Мобильное приложение «Тренажёр для развития памяти».
- 6 Мобильное приложение «Калькулятор ОСАГО».

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Лекционный курс основывается на методе обучения, использующем технологию, при которой студенты конспектируют теоретический материал, участвуют в опросах и дискуссиях. В этом случае задействованы зрительная, слуховая, моторная и ассоциативная виды памяти.

При прослушивании лекций рекомендуется в конспекте отмечать все важные моменты, на которых заостряет внимание преподаватель, в частности те, которые направлены на качественное выполнение соответствующей лабораторной работы.

Преподавателем запланировано использование при чтении лекций технологии учебной дискуссии. Поэтому рекомендуется фиксировать для себя интересные моменты с целью их активного обсуждения на дискуссии в конце лекции.

Залогом качественного выполнения лабораторных работ является самостоятельная подготовка к ним накануне путем повторения материалов лекций. Рекомендуется подготовить вопросы по неясным моментам и обсудить их с преподавателем в начале занятия.

Лабораторные работы выполняются с применением Android, Kotlin и новых версий этих программных продуктов.

Преподавателем запланировано применение на лабораторных занятиях технологий развивающейся кооперации, коллективного взаимодействия, разбора конкретных ситуаций.

Для текущего контроля успеваемости по очной форме обучения преподавателем используется балльно-рейтинговая система контроля и оценки академической активности. Поэтому настоятельно рекомендуется тщательно прорабатывать материал дисциплины при самостоятельной работе, участвовать во всех формах обсуждения и взаимодействия, как на лекциях, так и на

лабораторных занятиях в целях лучшего освоения материала и получения высокой оценки по результатам освоения дисциплины.

Выполнение самостоятельной работы подразумевает самостоятельное изучение разделов дисциплины, подготовку к лабораторным работам, к рубежным контролям, выполнение контрольной работы, подготовку к зачёту.

Рекомендуемая трудоемкость самостоятельной работы для очной формы обучения представлена в таблице:

**Рекомендуемый режим самостоятельной работы
Очная форма**

Наименование вида самостоятельной работы	Рекомендуемая трудоемкость, акад. час.
	Семестр 7
Самостоятельное изучение тем дисциплины:	4
Жизненный цикл activity	0,8
Отладка Android-приложений	0,8
Версии Android SDK и совместимость	0,8
UI-фрагменты и FragmentManager	0,8
Вывод списков и RecyclerView	0,8
Подготовка к лабораторным занятиям (по 1 часа на каждое занятие)	16
Подготовка к рубежным контролям (по 2 часа на каждый рубеж)	4
Выполнение контрольной работы	18
Подготовка к зачёту	18
Всего:	60

**6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ДЛЯ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ**

6.1. Перечень оценочных средств

- 1 Балльно-рейтинговая система контроля и оценки академической активности студентов в КГУ.
- 2 Отчеты студентов по лабораторным работам.
- 3 Банк заданий к рубежным контролям № 1, № 2.
- 4 Банк заданий к зачёту.
- 5 Контрольная работа.

6.2. Система балльно-рейтинговой оценки работы студентов по дисциплине

Очная форма обучения

№	Наименование	Содержание						
1	Распределение баллов за семестры по видам учебной работы, сроки сдачи учебной работы (<i>доводятся до сведения студентов на первом учебном занятии</i>)	Распределение баллов, 7 семестр						
		Вид учебной работы:	Посещение лекций	Выполнение и защита отчетов по лабораторным работам	Рубежный контроль №1	Рубежный контроль №2	Контрольная работа	Зачёт
	Балльная оценка:	26*8=166	56*8=406	5	5	4	30	
2	Критерий пересчета баллов в традиционную оценку по итогам работы в семестре и зачёта	60 и менее баллов – незачтено; 61... 73 – зачтено; 74... 90 – зачтено; 91... 100 – зачтено						
3	Критерии допуска к промежуточной аттестации, возможности получения автоматического зачёта по дисциплине, возможность получения бонусных баллов	<p>Для допуска к промежуточной аттестации (зачёту) студент должен набрать не менее 50 баллов и выполнить все лабораторные работы, рубежные контроли и контрольную работу.</p> <p>Для получения автоматического зачёта студенту необходимо набрать 61 балл.</p> <p>По согласованию с преподавателем студенту, могут быть добавлены дополнительные (бонусные) баллы за активность на консультациях, активное участие в научной и методической работе, оригинальность принятых решений в ходе выполнения лабораторных и контрольных работ, за участие в значимых учебных и внеучебных мероприятиях кафедры.</p>						
4	Формы и виды учебной работы для неуспевающих (восстановившихся на курсе обучения) студентов для получения недостающих баллов в конце семестра	<p>В случае если к промежуточной аттестации (зачёту) набрана сумма менее 50 баллов, студенту необходимо набрать недостающее количество баллов за счет выполнения дополнительных заданий, до конца последней (зачетной) недели семестра. При этом необходимо проработать материал всех пропущенных лабораторных и работ для зачёта.</p> <p>Формы дополнительных заданий (назначаются преподавателем):</p> <ul style="list-style-type: none"> - выполнение и защита пропущенной лабораторной работы для зачёта (при невозможности дополнительного проведения лабораторной работы преподаватель устанавливает форму дополнительного задания по тематике пропущенной лабораторной работы для зачёта самостоятельно) – до 8 баллов. <p>Ликвидация академических задолженностей, возникших из-за разности в учебных планах при переводе или восстановлении, проводится путем выполнения дополнительных заданий, форма и объем которых определяется преподавателем.</p>						

6.3 Процедура оценивания результатов освоения дисциплины

Рубежные контроли проводятся в виде ответов на вопросы в письменной форме. Зачёт проводится в виде ответов на вопросы билета в устной форме.

Перед проведением рубежного контроля преподаватель прорабатывает с обучающимися основной материал соответствующих разделов дисциплины в форме краткой лекции-дискуссии.

На подготовку к рубежному контролю обучающемуся отводится 2 часа самостоятельной работы. На выполнение тестовых заданий рубежных контролей обучающемуся отводится 2 часа на практических занятиях.

Варианты заданий для рубежных контролей № 1, № 2 состоят из 20 вопросов. Для определения баллов при проверке рубежных контролей используются интервальные оценки, представленные в таблице

Количество правильных ответов	1-5	6-8	9-11	12-14	15-17	18-20
Количество баллов	0	1	2	3	4	5

Преподаватель оценивает в баллах результаты рубежного контроля каждого обучающегося по количеству правильных ответов и заносит в ведомость учета текущей успеваемости.

Билет зачёта содержит 1 вопрос. Вопросы к зачёту доводятся до обучающегося на последней лекции в семестре. На подготовку ответа обучающему отводится 1 астрономический час.

Результаты текущего контроля успеваемости и зачёта заносятся преподавателем в зачётную ведомость, которая сдается в организационный отдел института в день зачёта, а также выставляются в зачетную книжку обучающегося.

6.4 Примеры оценочных средств для рубежных контролей и зачёта

6.4.1 Примеры заданий для рубежного контроля №1 Очная форма обучения, 7 семестр

Вариант 1_1

1 Кроссплатформенные приложения – приложения, разрабатываемые под конкретную аппаратно-программную платформу на языках, созданных для данной платформы?

1 Да.

*2 Нет.

2 Какие преимущества нативных приложений?

- *1 Высокая производительность.
- 2 Простое и быстрое развёртывание.
- 3 Покрывают широкую аудиторию.
- 4 Один код доступен для повторного использования на других платформах.

3 Какие недостатки нативных приложений?

- *1 Значительные финансовые и временные затраты на разработку.
- 2 Отсутствие гибкости.
- *3 Несовместимость с другими мобильными операционными системами.
- 4 Несоответствие UI в различных платформах.

4 Какие преимущества кроссплатформенных приложений?

- 1 Высокая производительность.
- *2 Простое и быстрое развёртывание.
- *3 Покрывают широкую аудиторию.
- *4 Один код доступен для повторного использования на других платформах.

5 Какие недостатки кроссплатформенных приложений?

- 1 Значительные финансовые и временные затраты на разработку.
- *2 Отсутствие гибкости.
- 3 Несовместимость с другими мобильными операционными системами.
- *4 Несоответствие UI в различных платформах.

6.4.2 Примеры заданий для рубежного контроля №2
Очная форма обучения, 7 семестр

Вариант 2_1

1 Какие существуют типы контейнера ViewGroup?

- 1 LinearLayout
- 2 RelativeLayout
- 3 FrameLayout
- 4

2 ConstraintSet программно создаёт привязки, цепочки, барьеры и настраивает их?

- *1 Да
- 2

3 С какой базой данных работает библиотека Room?

- 1
- 2
- 3

*4 SQLite

4 Какие компоненты имеет библиотека Room?

- *1 Entity
- *2 Dao

*3 Database

4

5 Что такое LiveData?

1 хранилище данных, работающее по принципу паттерна Mediator (посредник)

2 хранилище данных, работающее по принципу паттерна Iterator (итератор)

3 хранилище данных, работающее по принципу паттерна Visitor (посетитель)

*4 хранилище данных, работающее по принципу паттерна Observer (наблюдатель)

6.4.3 Примерный перечень вопросов для зачёта

1 Архитектура и базовые сведения о платформе Android.

2 Архитектуры кроссплатформенных фреймворков.

3 Основы разработки мобильных приложений.

4 Многослойный MVVM. Декомпозиция по слоям. Связи внутри слоёв. Связи между слоями. Структуры данных на основе UI.

5 Использование неявных интенгов. Строение неявного интенга. Отправка отчёта. Запрос контакта у Android.

6.5. Фонд оценочных средств

Полный банк заданий для текущего, рубежных контролей и промежуточной аттестации по дисциплине, показатели, критерии, шкалы оценивания компетенций, методические материалы, определяющие процедуры оценивания образовательных результатов, приведены в учебно-методическом комплексе дисциплины.

7. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА

7.1. Основная учебная литература

1 Васильев, Н. П. Введение в гибридные технологии разработки мобильных приложений : учебное пособие для вузов / Н. П. Васильев, А. М. Заяц. – 2-е изд., стер. – Санкт-Петербург : Лань, 2021. – 160 с. – ISBN 978-5-8114-8181-1. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/173103>.

2 Соколова, В. В. Разработка мобильных приложений : учебное пособие / В. В. Соколова. – Томск : ТПУ, 2014. – 176 с. – ISBN 978-5-4387-0369-3. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/82830>.

3 Льюис, Ш. , Данн М. Нативная разработка мобильных приложений / Льюис Ш. , Данн М. , пер. с англ. А. Н. Киселева. – Москва : ДМК Пресс, 2020. – 376 с. – ISBN 978-5-97060-845-6. – Текст : электронный // ЭБС "Кон-

сультант студента" : [сайт]. – URL :
<https://www.studentlibrary.ru/book/ISBN9785970608456.html>.

4 Сомон, П. Волшебство Kotlin : практическое руководство / П. Сомон ; пер. с англ. А. Н. Киселева. – Москва : ДМК Пресс, 2020. – 536 с. – ISBN 978-5-97060-801-2. – Текст : электронный. – URL: <https://znanium.com/catalog/product/1094968>.

7.2. Дополнительная учебная литература

1 Жемеров, Д. Kotlin в действии / Д. Жемеров, С. Исакова ; перевод с английского А. Н. Киселев. – Москва : ДМК Пресс, 2018. – 402 с. – ISBN 978-5-97060-497-7. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/112926>.

2 Хабитуев, Б. В. Программирование на языке Java: практикум : учебное пособие / Б. В. Хабитуев. – Улан-Удэ : БГУ, 2020. – 94 с. – ISBN 978-5-9793-1548-5. – Текст : электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/171791>.

3 Аделекан И. Kotlin: программирование на примерах: Пер с англ. – СПб.: БХВ-петербург, 2020. – 432 с.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

1. Семахин А.М. Разработка мобильных приложений. Методические указания к выполнению лабораторных и контрольных работ для студентов направлений подготовки 09.03.03 «Прикладная информатика», 09.03.04 «Программная инженерия». Курган, КГУ, 2021. – 48 с. (электронный).

4. Семахин А.М. Разработка мобильных приложений: учебное пособие. – Курган : Изд-во КГУ, 2021 – 76 с. (электронный).

9. РЕСУРСЫ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1. Федеральный портал «Российское образование» URL: <http://www.edu.ru/>

2. Сайт дистанционного обучения в НОУ «ИНТУИТ». URL: <http://www.intuit.ru/>

10. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

При чтении лекций используются слайдовые презентации.

Минимальные требования к операционной системе и программному обеспечению компьютера, используемого при показе слайдовых презентаций: Windows, Foxit Reader.

11. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Материально-техническое обеспечение включает в себя учебные лаборатории и классы, оснащенные современными компьютерами (рабочими станциями локальной вычислительной сети) с доступом в Интернет, мультимедийное оборудование (переносной персональный компьютер, мультимедийный проектор, мультимедийный экран).

Программные средства обеспечения учебного процесса включают лицензионное программное обеспечение: операционную систему Windows XP, язык программирования Kotlin, платформа Android.

Аннотация к рабочей программе дисциплины
«Разработка мобильных приложений»

образовательной программы высшего образования –
программы бакалавриата

09.03.03 – Прикладная информатика
Направленность:

Интеллектуальные информационные системы и технологии

Трудоемкость дисциплины: 3 ЗЕ (108 академических часа)
Семестр: 7 (очная форма обучения)

Содержание дисциплины

Нативные и кроссплатформенные инструменты разработки. Архитектура iOS/Android. Нативный iOS. Нативный Android. Нативный Windows UWP. Архитектура Android. Уровень ядра. Уровень библиотек. Уровень каркаса приложений. Компоненты Android-приложения. Activity. Service. Broadcast Receiver. Архитектуры кроссплатформенных фреймворков: PhoneGap, ReactNative, Qt, Flutter, {amarin, {amarin.Forms.

Создание проекта Android. Навигация в Android Studio. Создание макета пользовательского интерфейса. Иерархия представления. Атрибуты виджетов. Создание строковых ресурсов. Предварительный просмотр макета. От разметки XML к объектам View. Ресурсы и идентификаторы ресурсов. Разработка виджетов. Установка ссылок на виджеты. Назначение слушателей. Уведомления. Выполнение в эмуляторе. Процесс сборки Android-приложений. Инструменты сборки.

Создание нового класса. Архитектура «Модель–Представление–Контроллер» и Android. Преимущества MVC. Обновление уровня представления. Обновление уровня контроллера. Добавление ресурсов в проект. Ссылка на ресурсы в XML. Запуск на устройстве. Многослойный MVVM. Декомпозиция по слоям. Связи внутри слоёв. Связи между слоями. Структуры данных на основе UI.

Использование неявных интенгов. Строение неявного интенга. Отправка отчёта. Запрос контакта у Android. Проверка реагирующих activity. Хранилище файлов. Использование FileProvider. Использование интенга камеры. Масштабирование и отображение растровых изображений.

ConstraintLayout. Использование графического инструмента макетов. Освобождение пространства. Добавление виджетов. Внутренние механизмы

ConstrainLayout. Редактирование свойств. Динамическое поведение элементов списка. Стили, темы, атрибуты тем.

Библиотека компонентов архитектуры Room. Создание базы данных. Отделение сущности. Создание класса базы данных. Определение объекта доступа к данным. Доступ к базе данных с помощью шаблона репозитория. Тестирование запросов. Загрузка тестовых данных. Поток приложения. Фоновые потоки. Использование LiveData.

Создание объекта SoundPool. Доступ к активам. Воспроизведение звука. Зависимости от тестирования. Создание класса теста. Подготовка теста. Настройка тестируемых объектов. Написание тестов. Взаимодействия тестируемых объектов. Обратные вызовы привязки данных. Выгрузка звуков.

Подготовка RecyclerView. Подготовка к загрузке через URL. Множественные загрузки. Создание фонового потока. Запуск и остановка HandlerThread. Сообщения и обработчики сообщений. Структура сообщений. Структура обработчика. Использование обработчиков. Прослушивание жизненного цикла представления. Сохранённые фрагменты.