

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
ФГБОУ ВО «Уральский государственный экономический университет»

Одобрено
на заседании педагогического совета
колледжа

23 апреля 2020 г.
протокол № 9

Директор колледжа



А.Э. Чечулин

Утверждено
советом по учебно-методическим вопросам
и качеству образования

20 мая 2020 г.
протокол № 9



Д.А. Карх

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА МЕЖДИСЦИПЛИНАРНОГО КУРСА

Наименование междисциплинарного курса	Разработка мобильных приложений
Наименование специальности	09.02.07 Информационные системы и программирование
Форма обучения	Очно-заочная
Год набора	2020

Разработано
преподавателем

В.В. Плещевым

Екатеринбург
2020

1. ОБЩАЯ ХАРАКТЕРИСТИКА

Контрольно-оценочные средства (КОС) предназначены для контроля и оценки образовательных достижений обучающихся.

КОС включают контрольные материалы для проведения текущего контроля и промежуточной аттестации.

Результатом освоения междисциплинарного курса является формирование у обучающихся следующих профессиональных компетенций:

ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.6	Разрабатывать модули программного обеспечения для мобильных платформ

В результате освоения междисциплинарного курса обучающийся должен иметь:

Практический опыт	Умения	Знания
- в разработке мобильных приложений	- создавать программу по разработанному алгоритму как отдельный модуль; - осуществлять разработку кода программного модуля на современных языках программирования;	- основные этапы разработки программного обеспечения; - основные принципы технологии структурного и объектно-ориентированного программирования

2. МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ

Проверочные тесты

1. Какие методы отражают жизненный цикл Activity:

onCreate() (+)

onRestart()(+)

onResume()(+)

onRunning()

onClose()

2. Подробная информация о приложении содержится в файле:

AndroidManifest.xml (+)

main.xml

R.java

default.properties

3. Для создания всплывающего уведомления необходимо инициализировать объект:

Toast (+)

Message

TextView

MessageBox

4. Единица измерения dp или dip - это 1/72 дюйма, определяется по физическому размеру экрана дюйм, определяется по физическому размеру экрана абстрактная ЕИ, позволяющая приложениям выглядеть одинаково на различных экранах и разрешениях (+) физический элемент матрицы дисплея

5. Тип верстки при котором позиционирование элементов происходит относительно друг друга и относительно главного контейнера

AbsoluteLayout
FrameLayout
LinearLayout
RelativeLayout (+)

6. Какой класс можно использовать для перехода между Activity?

ActivityChanger
Activity Intent (+)
Switcher

7. От какого класса наследуются все элементы управления?

Control
Controls
Element
View (+)

8. Для чего используется класс R?

Класс ресурсов (+) Класс для управления элементами управления Класс для работы с потоками Класс, предназначенный только для доступа к идентификаторам элементов

9. Как программно получить доступ к элементам управления через идентификатор?

getViewById(id)
findViewById(id) (+)
findViewById(id)
getView(id)

10. В какой библиотеке находится класс Activity?

android
android.Activities
android.app (+)
java.app

11. На основе какой операционной системы была разработана система Android:

Linux (+)
MiniX
Windows
iOS

12. Как называется режим использования Android-системы с максимальными правами (аналог администратора Windows):

Суперпользователь (+)
Root-Home пользователь
Android user
Up-User

13. Как называется событие-клик объекта-кнопки:

OnClick (+)
OnClicked
Click
ClickEvent

14. Для добавления в приложение более одного экрана взаимодействия необходимо создание:

Activity (+)
View

Layout

XML-файла

15. Как называется класс для работы с SQLite-базами в Android:

SQLiteOpenHelper (+)

SQLProvider

SQLiteProvider

SQLiteContext

16. Как называется язык разметки Window Phone приложений:

XAML (+)

XML

Java

C#

17. С помощью какого класса можно вывести сообщение в Window Phone приложении:

Toast

MessageBox (+)

Message

ShowClass

18. Какой элемент предоставляет вариант разметки, элементы, помещенные в который, располагаются в стек горизонтально или вертикально:

StackPanel (+)

Pivot

Panorama

Canvas

19. Какой элемент позволят позиционировать элементы внутри себя максимально гибко предоставляя возможность размещать элементы, используя строки и столбцы.

Grid (+)

Pivot

Panorama

Canvas

20. Какой элемент предоставляет наиболее простой вариант разметки. Он может быть использован для абсолютного позиционирования элементов с использованием координат.

Grid

Pivot

Panorama

Canvas (+)

21) На базе какого ядра сделано ядро Android?

1. Windows NT

2. Linux

3. FreeBSD

4. VxWorks

22) Какое название носит виртуальная Java-машина Android?

1. Dalvik

2. B ionic

3. HotSpot

23) Как изменился набор консольных утилит Linux в Android?

1. Существенно уменьшился

2. Остался без изменений

3. Существенно увеличился

4. В Android нет консольных утилит
- 24) Для чего предназначен Binder?
1. Связывает файлы ресурсов с приложением
 2. Обеспечивает межпроцессное взаимодействие
 3. Обмен информацией между приложениями
- 25) Является ли logcat полным аналогом gdb?
1. Да
 2. logcat — полный аналог gdbserver
 3. Нет
- 26) Из-за чего glibc была заменена на Bionic?
1. Смена бренда
 2. Несовместимость glibc со свободными лицензиями
 3. glibc невозможно скомпилировать для ARM
 4. Уменьшение размеров объектного кода
- 27) Какой байт-код используется в Dalvik?
1. Обычный
 2. Исключительно свой собственный
 3. Свой собственный, но обычный может быть в него сконвертирован
- 28) Из чего состоит Java API в Android?
1. Часть стандартных библиотек Java
 2. Стандартные библиотеки Java и пакет javax.
 3. Набор стандартных, популярных и собственных пакетов
 4. Исключительно собственные библиотеки Android
- 29) Какую функциональность обеспечивает пакет android.view?
1. Реализация GUI
 2. Работа с камерой
 3. Темы рабочего стола
- 30) Какой способ беспроводной связи есть в пакете android.net?
1. Bluetooth
 2. WiFi
 3. NFC
 4. DRM

Вопросы для самоконтроля

1. Архитектура платформы Android. Уровень ядра. Уровень библиотек.
2. Архитектура платформы Android. Dalvik Virtual Machine.
3. Архитектура платформы Android. Уровень каркаса приложений. Уровень приложений.
4. Среда разработки для Android. Eclipse IDE. Плагин ADT. Android Virtual Device.
5. Android SDK. Версии SDK и Android API Level.
6. Структура проекта Android-приложения в Eclipse. Каталоги ресурсов. Файл R.java.
7. Графический интерфейс пользователя в Android-приложениях. XML-разметка интерфейса.
8. Архитектура платформы Android.
9. XML-разметка интерфейса пользователя.
10. XAML-разметка интерфейса пользователя.
11. Базовые элементы управления.
12. Ресурсы в Android-приложениях.
13. Ресурсы в Windows Phone-приложениях.

14. Активности и интен­ты.
15. Обработка пользовательского ввода. Касания, ввод текста.
16. Типы компоновок графического интерфейса. FrameLayout, LinearLayout, TableLayout, RelativeLayout.
17. Базовые элементы управления. TextView. EditText. Тип ввода текста. Параметры отображения клавиатуры. ImageView.
18. Диалоговые окна. AlertDialog. ProgressDialog. DatePickerDialog. TimePickerDialog. Создание пользовательских диалоговых окон.
19. Многопоточные приложения в Android и Windows Phone. Использование системных таймеров и системного времени.
20. Процессы в Android. Объекты Activity. Состояния Activity.
21. Использование объектов Intent. Intent-фильтры.
22. Использование ресурсов. Ссылки на ресурсы. Загрузка простых типов из ресурсов. Загрузка файлов произвольного типа.
23. Файловая система Android. Чтение и запись файлов.
24. Адаптеры данных. Отображение данных в компонентах ListView, GridView, AutoCompleteTextView, MultiAutoCompleteTextView.
25. Пользовательские настройки. Использование SharedPreferences. Виды настроек.
26. Работа с графикой. Drawable и Canvas.
27. Работа с анимацией. Tween Animation и Frame Animation. Описание анимации в XML и в коде программы.
28. Службы в Android. Компонент Service.
29. Датчики мобильных устройств. Управление датчиками в приложении. Виды датчиков и особенности их использования.
30. Программный доступ к дисплею устройства. Менеджер окон. Параметры дисплея.

3. МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Формы промежуточной аттестации: 5-й семестр - **зачет**; 6-й семестр – **дифференцированный зачет**.

Примеры практико-ориентированных заданий

Задание 1. Скачать Android SDK + Eclipse (Eclipse ADT Bundle) <http://developer.android.com/sdk/index.html>, а также последнюю версию Java (JDK) <http://www.oracle.com/technetwork/java/index-jsp-138363.html> (или просто набрать в уа­dex "JDK" и пройти по ссылке на oracle.com). Установить всё это. Создать новый про­ект, зайти в Android SDK и скачать какую-нибудь версию Android SDK (например, lol­lipop - 5.0.1), выделив все инструменты для работы с ней; сконфигурировать эмулятор (желательно эмулировать своё собственное мобильное устройство, включая версию ан­дроид на нём в качестве Target SDK). Изменить TextView с надписью "Hello world" на какую-нибудь другую надпись по желанию. Запустить проект на эмуляторе и убедиться, что всё работает.

Задание 2. Создать новый проект, написать программу, которая выводит в эле­мент TextView надпись, введенную пользователем в текстовом поле EditText после нажатия на кнопку Button. Помимо этого в Activity должен быть TextView с ФИО сту­дента и группой. Запустить на эмуляторе и убедиться, что всё работает.

Задание 3. Создать приложение, которое состоит из нескольких activities. Первое activity содержит элемент TextView с названием или номером activity, текстовое поле EditText для ввода какой-то информации, кнопку Button с названием "Next" или "Пе­

рейти на 2 activity/экран/окно" или просто "2". Помимо этого в 1 activity должен быть TextView с ФИО студента и группой. После нажатия на эту кнопку происходит переход на второе activity, где содержится TextView с названием или номером activity, TextView с надписью что-то вроде "В первом окне вы напечатали:" и под ним - ещё один TextView с содержимым EditText с первого activity, и, разумеется, кнопка "1" или "Вернуться на 1 экран" или "Вернуться к вводу текста", нажав на которую пользователь может перейти обратно к 1 activity. Запустить на эмуляторе и убедиться, что всё работает.

Задание 4. В новом проекте написать приложение, работающее с разными темами/стилями. Сначала создать свой стиль и применить его к какому-нибудь интерфейвному элементу, затем - свою тему, которая применяется ко всем интерфейсным элементам. Приложение при этом должно выглядеть нестандартно, запустить на эмуляторе и убедиться, что всё работает. При возникновении ошибок открыть лог (CatLog) внизу, найти первую красную надпись и породить свою тему от той, которая требуется в этом красном сообщении.

Задание 5. Создать пользовательский (свой) список. Например, получить доступ в приложении к контактам (Permissions-закладка в AndroidManifest.xml) и скопировать контакты телефона в свой список, который отобразить после запуска приложения. Или создать свой список в виде твиттера (картинка+текст), элементы которого просто статически задать в массиве (как и картинки).

Задание 6. Создать приложение, содержащее анимированные интерфейсные элементы (например, увеличивающиеся при клике на них кнопки, вращающиеся TextView и т.д.).

Задание 7. Создать приложение, отображающее после запуска карты Google или какие-нибудь другие карты.

Задание 8. Создать собственный виджет с настройками. Например, виджет, который открывает какой-то сайт, адрес которого можно поменять в настройках.

Задание 9. Создать приложение, использующее опциональное меню (меню настроек) и контекстное меню для какого-нибудь интерфейсного элемента. Естественно, выбор пунктов меню должен что-то менять в интерфейсных элементах или их отображении! Например, очистить поле ввода через контекстное меню, или отобразить невидимые интерфейсные элементы через установку галок в опциональном меню.

Задание 10. Создать приложение, отображающее после некоторых действий (нажатия на кнопку, например, или проверки корректности ввода текста в EditText) диалоговое окно, свидетельствующее об ошибке/информирующее/предупреждающее пользователя о чём-то.

Задание 11. Создать приложение, помещающее по нажатию на кнопку какое-то сообщение со звуком в панель уведомлений/статус-панель на эмуляторе.

Задание 12. Создать приложение, по нажатию кнопки в котором проигрывается какой-то звук.

Задание 13. Создать приложение, при запуске которого проигрывается какое-то видео.

Задание 14. Создать приложение, при запуске которого активируется фотокамера телефона, производится снимок, и этот снимок помещается в Image View интерфейса приложения.

Задание 15. Создать приложение, работающее с SharedPreferences и сохраняющее настройки, а также работающее с БД SQLite - заполняющее БД по нажатию кнопки 1 с помощью EditText, и выводящее все записи этой БД в какой-нибудь интерфейсный элемент ниже с помощью кнопки 2 (в виде списка, datagrid или просто правильно настроенного Text View).

Вопросы к зачету/дифференцированному зачету

1. История возникновения мобильных операционных систем
2. Основные этапы становления рынка мобильных приложений
3. Современное состояние рынка мобильных приложений
4. Классификация видов мобильных приложений
5. Преимущества использования мобильных приложений в сравнении с веб-приложениями
6. Недостатки использования мобильных приложений в сравнении с десктопными приложениями
7. Инструментальные среды разработки мобильных приложений для операционной системы Apple iOS
8. Инструментальные среды разработки мобильных приложений для операционной системы Android
9. Инструментальные среды разработки мобильных приложений для операционной системы Windows Phone
10. Структура операционной системы iOS
11. Структура операционной системы Android
12. Структура приложения iOS
13. Структура приложения Android
14. Основные требования к интерфейсу приложений iOS
15. Основные требования к интерфейсу приложений Android
16. Android-манифест
17. Взаимодействие Android-приложения с сетью
18. Работа Android-приложения с локальной базой данных
19. Считывание информации Android-приложением с XML-файла
20. Вызов приложения из другого приложения в ОС Android
21. Проблемы безопасности мобильных операционных систем
22. Бизнес-модели распространения мобильных приложений
23. Стратегия размещения приложения на Google Play
24. Стратегия размещения приложения на AppStore
25. Сравнительная характеристика современных мобильных операционных систем
26. HTML5 и мобильные приложения
27. Проблемы совместимости мобильных приложений со старыми версиями операционных систем
28. Проектирование мобильных приложений с использованием C++
29. Технологии фреймворков в проектировании мобильных приложений
30. Проблемы масштабирования СУБД в мобильных приложениях
31. Основные технологии виртуализации в инструментальных средах при создании мобильных приложений
32. Проблемы обеспечения безопасности в платных мобильных приложениях
33. Перспективы развития рынка мобильных приложений в России

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ МЕЖДИСЦИПЛИНАРНОГО КУРСА

Критерии оценки (недифференцированной)	Критерии оценки (дифференцированной)
Оценка «зачтено» выставляется студенту, проявившему знания основного про-	«Отлично» - теоретическое содержание курса освоено полностью, без пробелов,

<p>граммного материала в объеме, необходимом для последующего обучения и предстоящей практической деятельности, знакомому с основной рекомендованной литературой, но допустившему неполные или слабо аргументированные ответы, испытывающему затруднения.</p> <p>Оценка «не зачтено» выставляется студенту, обнаружившему значительные пробелы в знании программного материала, допустившему принципиальные ошибки при применении теоретических знаний, которые не позволяют ему продолжить обучение или приступить к практической деятельности без дополнительной подготовки по данной дисциплине</p>	<p>умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p> <p>«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p> <p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки</p>
--	--