

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Шарифуллин Федеральное государственное бюджетное образовательное учреждение высшего образования

Должность: Директор «Информационный центр» «РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПРАВОСУДИЯ»

Дата подписания: 14.10.2024 10:18:54

Уникальный программный ключ:

65fd6cbdf7eae29c01b701aabc1fbc13d72d7bd0b08b122e44091c482448eba9

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

(наименование дисциплины в соответствии с учебным планом)

Специальность:

09.02.07 Информационные системы и программирование

(код и наименование)

Рабочая программа разработана в соответствии с требованиями ФГОС.

Разработчик (-):

Квачко В.Ю., к.ф.м.н., доцент

(ФИО, ученая степень, ученое звание)

Зав. кафедрой

Ловцов Д.А., д.т.н., профессор

(ФИО, ученая степень, ученое звание)

(подпись)

ПРОТОКОЛ ИЗМЕНЕНИЙ
рабочей программы дисциплины (модуля)

«РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»
(наименование дисциплины в соответствии с учебным планом)

Краткое содержание изменения	Дата и номер протокола заседания кафедры

Актуализация выполнена: Квачко В.Ю., к.ф.м.н., доцент
(ФИО, ученая степень, ученое звание) (подпись)

Зав. кафедрой Ловцов Д.А., д.т.н., профессор
(ФИО, ученая степень, ученое звание) (подпись)

ОГЛАВЛЕНИЕ

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ.....	4
1. ЦЕЛИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ	5
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП	6
3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ	6
4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	6
4.1. ТЕКСТ РАБОЧЕЙ ПРОГРАММЫ ПО ТЕМАМ	6
4.2. РАЗДЕЛЫ И ТЕМЫ ДИСЦИПЛИНЫ, ВИДЫ ЗАНЯТИЙ (ТЕМАТИЧЕСКИЙ ПЛАН)	7
4.3. САМОСТОЯТЕЛЬНОЕ ИЗУЧЕНИЕ ОБУЧАЮЩИМИСЯ РАЗДЕЛОВ ДИСЦИПЛИНЫ.....	8
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	8
5.1. УЧЕБНО-МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ.....	8
5.2. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ.....	13
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ.....	14
7. КАРТА ОБЕСПЕЧЕННОСТИ ЛИТЕРАТУРОЙ.....	15
8. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ.....	17
8.1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ.....	17
8.2. ОЦЕНОЧНЫЕ СРЕДСТВА	17

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

«РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

(наименование дисциплины в соответствии с учебным планом)

Разработчик (-и): Квачко В.Ю., к.ф.м.н., доцент
(ФИО, ученая степень, ученое звание)

Цель изучения дисциплины (модуля)	Изучение принципов создания приложений, поддерживающих требования интерфейса операционной среды, типовых приемов организации и конструирования пакетов программ сложной структуры, этапов процесса проектирования программного обеспечения, создание прикладных программ с высокой степенью автоматизации управления.
Место дисциплины в структуре ООП	Дисциплина П.01.03 РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ относится к дисциплинам профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем»
Компетенции, формируемые в результате освоения дисциплины (модуля)	<p>ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.</p> <p>ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.</p> <p>ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях.</p> <p>ОК 04. Эффективно взаимодействовать и работать в коллективе и команде.</p> <p>ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.</p> <p>ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных российских духовно-нравственных ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.</p> <p>ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.</p> <p>ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.</p> <p>ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.</p> <p>ПК 1.2. Разрабатывать программные модули в соответствии</p>

	с техническим заданием. ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.
Содержание дисциплины (модуля)	Тема 1. Мобильные приложения и технологии Тема 2. Создание и тестирование модулей для мобильных приложений Тема 3. Разработка мобильных приложений под Android. Тема 4. Разработка мобильных приложений под iOS. Тема 5. Разработка мобильных приложений под WindowsPhone.
Общая трудоемкость дисциплины (модуля)	Общая трудоемкость дисциплины составляет 85 часов.
Форма промежуточной аттестации	6 семестр – контрольное задание; 7 семестр – Дифференцированный зачёт.

1. ЦЕЛИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

Целью изучения дисциплины (модуля) является освоение компетенций (индикаторов достижения компетенций), предусмотренных рабочей программой.

В совокупности с другими дисциплинами ООП дисциплина обеспечивает формирование следующих компетенций:

Таблица 1

№ п/п	Код компетенции	Название
1	ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.
2	ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.
3	ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях.
4	ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде.
5	ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
6	ОК 06.	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных российских духовно-нравственных ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.
7	ОК 07.	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.
8	ОК 08.	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.
9	ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.

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

Планируемые результаты освоения дисциплины в части каждой компетенции указаны в картах компетенций по ООП.

В рамках дисциплины осуществляется воспитательная работа, предусмотренная рабочей программой воспитания, календарным планом воспитательной работы.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Учебная дисциплина «Разработка мобильных приложений» относится к дисциплинам профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем»

3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Таблица 2
очная форма обучения

Вид учебной работы	Трудоемкость			
	зач. ед.	час.	по семестрам	
			6	7
Общая трудоемкость дисциплины по учебному плану		85	57	28
Контактная работа				
Занятия лекционного типа		32	18	14
Занятия семинарского типа				
Занятия семинарского типа с практической подготовкой		52	38	14
Самостоятельная работа		1	1	
Форма промежуточной аттестации			<i>КЗ</i>	<i>ДЗ.</i>

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. ТЕКСТ РАБОЧЕЙ ПРОГРАММЫ ПО ТЕМАМ

Тема 1. Мобильные приложения и технологии.

Основные платформы мобильных приложений, сравнительная характеристика.

Нативные приложения, веб-приложения, гибридные и кроссплатформенные приложения, их области применения.

Основные языки для разработки мобильных приложений (Java, Objective-C и др.)

Инструменты разработки мобильных приложений (JDK/ AndroidStudio/ WebView/ Phonegap и др.)

Установка инструментария и настройка среды для разработки мобильных приложений.

Установка среды разработки мобильных приложений с применением виртуальной машины

Тема 2. Создание и тестирование модулей для мобильных приложений

Инструментарий среды разработки мобильных приложений

Структура типичного мобильного приложения

Элементы управления и контейнеры

Работа со списками

Способы хранения данных

Создание эмуляторов и подключение устройств

Тестирование и оптимизация мобильного приложения.

Создание эмуляторов. Подключение устройств. Настройка режима терминала. Создание многоэкранного приложения Использование шаблона проектирования MVVM. Работа с JSON, XML, сжатие данных

Обработка событий: подсказки. Обработка событий: цветовая индикация. Подготовка стандартных модулей. Обработка событий: переключение между экранами

Тема 3. Разработка мобильных приложений под Android.

Платформа Android

Создание первого приложения под Android. Основы разработки интерфейсов мобильных приложений.

Создание тем для упрощения работы с элементами. Применение DDMS для отладки приложения.

Создание лога. Списки.

Работа с ориентацией экрана, применение различных layouts.

Анимация. Рисование. Меню. Кнопки. Диалоговые окна. Сообщения. Мультимедиа.

Работа с базами данных в Android. Основные приемы работы с инструментами разработки

Тема 4. Разработка мобильных приложений под iOS.

Платформа iOS.

Создание нового проекта. Изучение кода. Комментирование кода. Изменение элементов дизайна

Создание приложения, которое состоит из нескольких activities.

Написание приложения, работающее с разными темами/стилями.

Тема 5. Разработка мобильных приложений под WindowsPhone.

Приложения под WindowsPhone

Создание собственного виджета с настройками.

Создание приложения, отображающее после запуска карты Google или какие-нибудь другие карты.

Шаблоны проектов, структура проектов. Элементы управления. Разработка пользовательского элемента управления.

4.2. РАЗДЕЛЫ И ТЕМЫ ДИСЦИПЛИНЫ, ВИДЫ ЗАНЯТИЙ (ТЕМАТИЧЕСКИЙ ПЛАН)

Таблица 3

Тематический план

очная форма обучения

№	Раздел дисциплины, тема	Код компетенции	Общая трудоёмкость дисциплины	в том числе			Наименование оценочного средства
				Контактная работа	Занятия лекционного типа	Занятия семинарского типа с практической подготовкой	
				час.	час.	час.	
1	Мобильные приложения и технологии.	ОК 01-ОК 09	16	16	12	4	Вопросы по теме, реферат

2	Создание и тестирование модулей для мобильных приложений	ОК 01-ОК 09	24	24	16	8	Вопросы по теме, реферат
3	Разработка мобильных приложений под Android.	ПК 1.2 ПК 1.6	18	18	2	16	Фонд тестовых заданий, практические задания
4	Разработка мобильных приложений под iOS.	ПК 1.2 ПК 1.6	18	18	2	18	Фонд тестовых заданий, практические задания
5	Разработка мобильных приложений под WindowsPhone.	ПК 1.2 ПК 1.6	9	8	2	6	Фонд тестовых заданий, практические задания
ВСЕГО			85	84	32	52	

4.3. САМОСТОЯТЕЛЬНОЕ ИЗУЧЕНИЕ ОБУЧАЮЩИМИСЯ РАЗДЕЛОВ ДИСЦИПЛИНЫ

Таблица 4

№ раздела (темы) дисциплины (модуля)	Вопросы, выносимые на самостоятельное изучение	Кол-во часов
1	Графический дизайн и пользовательские интерфейсы.	1

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. УЧЕБНО-МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ

Актуальность изучения учебной дисциплины обусловлена внедрением в юридическую практику крупномасштабных автоматизированных информационных систем (АИС), таких как Государственная автоматизированная система (ГАС) РФ «Правосудие», ГАС «Выборы», «АСБР-Москва» Банка России и др., а также увеличением масштабов общественно-производственной деятельности (включая юридическую) и, в связи с этим, объективной необходимостью уменьшения сроков рассмотрения различных юридических коллизий, возникающих в обществе, государстве и на мировой арене.

Для того, чтобы такие материалы можно было легко использовать для создания на основе их информационных сервисов полезных для бизнеса и граждан, их необходимо публиковать в машиночитаемом формате, обеспечивающим автоматическую обработку опубликованных данных. Размещённые в таком формате материалы считаются открытыми данными. Они становятся пригодными для действительно свободного использования в любых соответствующих закону целях любыми лицами, что способствует расширению спроса на эту информацию с целью достижения социального и экономического эффекта от её применения.

Согласно стратегии развития отрасли информационных технологий в Российской Федерации на 2014-2020 годы от 1 ноября 2013 года «Развитию отрасли информационных технологий содействует реализация концепции открытых данных и организация процесса их использования для создания сервисов, способствующих развитию социально значимых сфер». Работать с открытыми данными с помощью мобильных приложений пользователям гораздо легче, чем самостоятельно анализировать информацию на

порталах. Поэтому создание социально-ориентированных сервисов и приложений, которые позволяют значительно упростить повседневную жизнь, тем самым, повышая доступность открытых данных для конечных пользователей, становится все более актуальным.

Структура и отчетность – дисциплина включает последовательное изучение пяти тем, обязательное самостоятельное конспектирование трёх основных нормативно-правовых актов и заканчивается дифференцированным зачётом.

Список основной нормативной литературы для конспектирования:

1. *Закон об информации* – Федеральный закон от 29 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации» // Российская газета. – 2006. – 29 июля;

2. *Закон об электронной подписи* – Федеральный закон от 6 апреля 2011 г. № 63-ФЗ «Об электронной подписи» // Российская газета. – 2011. – 8 апр.

3. *Статьи 160, ч. 2; 434, ч. 2 ГК РФ* – Гражданский кодекс РФ, часть первая от 30 ноября 1994 г. с изменениями, внесёнными Федеральным законом от 10 января 2003 г. № 15-ФЗ // СЗ РФ. – 1994. – № 32. – Ст. 3301; 2003. – № 2. – Ст. 167.

Основные направления правовой информатизации:

- внедрение крупномасштабных АИС;
- широкое использование современных информационно-компьютерных систем, средств, ресурсов и технологий;
- развитие отрасли информационного права.

Операционные системы для которых создаются мобильные приложения.

Android - операционная система для мобильных устройств: смартфонов, планшетных компьютеров, КПК. В настоящее время именно Android является самой

Таблица 1. Мировые продажи смартфонов конечным пользователям, распределение по ОС

Операционная система	Продано (тыс.ед.) III кв. 2019	Доля рынка (%) III кв. 2019	Продано (тыс.ед.) III кв. 2018	Доля рынка (%) III кв. 2018
Android	205022,7	81,9	124552,3	72,6
iOS	30330,0	12,1	24620,3	14,3
Microsoft	8912,3	3,6	3993,6	2,3
BlackBerry	4400,7	1,8	8946,8	5,2
Bada	633,3	0,3	4454,7	2,6
Symbian	457,5	0,2	4401,3	2,6
Другие	475,2	0,2	683,7	0,4
Общее кол-во:	250231,7	100,0	171652,7	100,0

широко используемой операционной системой для мобильных устройств.

Подтверждение этого факта можно найти в таблице 1, составленной по данным аналитической компании Gartner. Внимательное изучение таблицы позволяет увидеть подавляющую популярность смартфонов под управлением ОС Android в мире, доля таких устройств не первый год превышает половину от общего числа купленных смартфонов. Кроме всего прочего, эта популярность продолжает расти. Очевидно, что армия пользователей смартфонов под управлением Android будет искать дополнительные приложения для своих устройств, в связи с этим умение разрабатывать эти самые приложения может принести много пользы своему владельцу. Например, можно разрабатывать для себя полезные, интересные, занимательные (нужное подчеркнуть) приложения, а можно, разведав обстановку и осмотревшись, сделать разработку

мобильных приложений своей профессиональной деятельностью, основной или дополнительной.

Виды приложений и их структура.

Приступая к разработке мобильных приложений хорошо бы иметь представление о том, какие виды приложений существуют. Дело в том, что если удастся определить к какому типу относится приложение, то становится понятнее на какие моменты в процессе его разработки необходимо обращать основное внимание. Можно выделить следующие виды приложений:

Приложения переднего плана выполняют свои функции только, когда видимы на экране, в противном же случае их выполнение приостанавливается. Такими приложениями являются, например, игры, текстовые редакторы, видеопроигрыватели. При разработке таких приложений необходимо очень внимательно изучить жизненный цикл активности, чтобы переключения в фоновый режим и обратно проходили гладко (бесшовно), т. е. при возвращении приложения на передний план было незаметно, что оно вообще куда-то пропало. Для достижения этой гладкости необходимо следить затем, чтобы при входе в фоновый режим приложение сохраняло свое состояние, а при выходе на передний план восстанавливало его. Еще один важный момент, на который обязательно надо обратить внимание при разработке приложений переднего плана, удобный и интуитивно понятный интерфейс.

Фоновые приложения после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии. Примерами таких приложений могут служить, службы экранирования звонков, SMS-автоответчики. В большинстве своем фоновые приложения нацелены на отслеживание событий, порождаемых аппаратным обеспечением, системой или другими приложениями, работают незаметно. Можно создавать совершенно невидимые сервисы, но тогда они будут неуправляемыми. Минимум действий, которые необходимо позволить пользователю: санкционирование запуска сервиса, настройка, приостановка и прерывание его работы при необходимости.

Смешанные приложения большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки. Обычно взаимодействие с пользователем сводится к уведомлению о каких-либо событиях. Примерами таких приложений могут служить мультимедиа проигрыватели, программы для обмена текстовыми сообщениями (чаты), почтовые клиенты. Возможность реагировать на пользовательский ввод и при этом не терять работоспособности в фоновом режиме является характерной особенностью смешанных приложений. Такие приложения обычно содержат как видимые активности, так и скрытые (фоновые) сервисы, и при взаимодействии с пользователем должны учитывать свое текущее состояние. Возможно потребуется обновлять графический интерфейс, если приложение находится на переднем плане, или же посылать пользователю уведомления из фонового режима, чтобы держать его в курсе происходящего. И эти особенности необходимо учитывать при разработке подобных приложений.

Виджеты - небольшие приложения, отображаемые в виде графического объекта на рабочем столе. Примерами могут служить, приложения для отображения динамической информации, такой как заряд батареи, прогноз погоды, дата и время. Разумеется, сложные приложения могут содержать элементы каждого из рассмотренных видов. Планируя разработку приложения, необходимо определить способ его использования, только после этого приступать к проектированию и непосредственно разработке.

Безопасность

Обратим внимание на организацию исполнения приложений в ОС Android. Как уже было отмечено приложения под Android разрабатываются на языке программирования Java, компилируется в файл с расширением .apk, после этот файл используется для установки приложения на устройства, работающие под управлением Android. После

установки каждое Android приложение "живет" в своей собственной безопасной "песочнице", рассмотрим, как это выглядит:

- операционная система Android является многопользовательской ОС, в которой каждое приложение рассматривается как отдельный пользователь;
- по умолчанию, система назначает каждому приложению уникальный пользовательский ID, который используется только системой и неизвестен приложению;
- система устанавливает права доступа ко всем файлам приложения следующим образом: доступ к элементам приложения имеет только пользователь с соответствующим ID;
- каждому приложению соответствует отдельный Linux процесс, который запускается, как только это необходимо хотя бы одному компоненту приложения, процесс прекращает работу, когда ни один компонент приложения не использует его или же системе требуется освободить память для других (возможно, более важных) приложений;
- каждому процессу соответствует отдельный экземпляр виртуальной машины Dalvik, в связи с этим код приложения выполняется изолировано от других приложений.

Перечисленные идеи функционирования приложения в ОС Android реализуют принцип минимальных привилегий, т. е. каждому приложению, по умолчанию, разрешен доступ только к компонентам, необходимым для его работы и никаким больше. Таким образом, обеспечивается очень безопасная среда функционирования приложений.

Однако в случае необходимости приложения могут получить доступ к данным других приложений и системным сервисам (услугам). В случае, когда двум приложениям необходимо иметь доступ к файлам друг друга, им присваивается один и тот же пользовательский ID. Для экономии системных ресурсов такие приложения запускаются в одном Linux процессе и делят между собой один и тот же экземпляр виртуальной машины, в этом случае приложения также должны быть подписаны одним сертификатом. В случае же, когда приложению требуется доступ к системным данным, например, контактам, SMS сообщениям, картам памяти, камере, Bluetooth и т. д., пользователю необходимо дать приложению такие полномочия во время установки его на устройство.

Архитектура приложения, основные компоненты

Вот и пришла пора поговорить непосредственно о внутренней организации приложений под Android: обсудить их архитектуру и основные компоненты.

Архитектура Android приложений основана на идее многократного использования компонентов, которые являются основными строительными блоками. Каждый компонент является отдельной сущностью и помогает определить общее поведение приложения.

Система Android выстроена таким образом, что любое приложение может запускать необходимый компонент другого приложения. Например, если приложение предполагает использование камеры для создания фотографий, совершенно необязательно создавать в этом приложении активность для работы с камерой. Наверняка на устройстве уже есть приложение для получения фотографий с камеры, достаточно запустить соответствующую активность, сделать фотографию и вернуть ее в приложение, так что пользователь будет считать, что камера часть приложения, с которым он работает.

Когда система запускает компонент, она запускает процесс приложения, которому принадлежит компонент, если он еще не запущен, и создает экземпляры классов, необходимых компоненту. Поэтому в отличие от большинства других систем, в системе Android приложения не имеют единой точки входа (нет метода `main()`, например). В силу запуска каждого приложения в отдельном процессе и ограничений на доступ к файлам, приложение не может напрямую активировать компонент другого приложения. Таким образом, для активации компонента другого приложения необходимо послать системе сообщение о намерении запустить определенный компонент, система активирует его.

Можно выделить четыре различных типа компонентов, каждый тип служит для достижения определенной цели и имеет свой особый жизненный цикл, который определяет способы создания и разрушения соответствующего компонента.

Рассмотрим основные компоненты Android-приложений.

Активности (Activities). Активность - это видимая часть приложения (экран, окно, форма), отвечает за отображение графического интерфейса пользователя. При этом приложение может иметь несколько активностей, например, в приложении, предназначенном для работы с электронной почтой, одна активность может использоваться для отображения списка новых писем, другая активность – для написания, и еще одна - для чтения писем. Несмотря на то, что для пользователя приложение представляется единым целым, все активности приложения не зависят друг от друга. В связи с этим любая из этих активностей может быть запущена из другого приложения, имеющего доступ к активностям данного приложения. Например, приложение камеры может запустить активность, создающую новые письма, чтобы отправить только что сделанную фотографию адресату, указанному пользователем.

Сервисы (Services). Сервис - компонент, который работает в фоновом режиме, выполняет длительные по времени операции или работу для удаленных процессов. Сервис не предоставляет пользовательского интерфейса. Например, сервис может проигрывать музыку в фоновом режиме, пока пользователь использует другое приложение, может загружать данные из сети, не блокируя взаимодействие пользователя с активностью. Сервис может быть запущен другим компонентом и после этого работать самостоятельно, а может остаться связанным с этим компонентом и взаимодействовать с ним.

5.2. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

Информационные, в том числе электронные ресурсы Университета, а также иные электронные ресурсы, необходимые для изучения дисциплины (модуля):

№ п./п.	Наименование	Адрес в сети Интернет
1	ZNANIUM.COM	http://znanium.com Основная коллекция и коллекция издательства Статут
2	ЭБС ЮРАЙТ	www.biblio-online.ru
3	ЭБС «BOOK.ru»	www.book.ru коллекция издательства Проспект Юридическая литература; коллекции издательства Кнорус Право, Экономика и Менеджмент
4	East View Information Services	www.ebiblioteka.ru Универсальная база данных периодики (электронные журналы)
5	НЦР РУКОНТ	http://rucont.ru/ Раздел Ваша коллекция – РГУП-периодика (электронные журналы)
6	Информационно-образовательный портал РГУП	www.op.raj.ru электронные версии учебных, научных и научно-практических изданий РГУП
7	Система электронного обучения «Фемида»	www.femida.raj.ru Учебно-методические комплексы, Рабочие программы по направлению подготовки
8	Правовые системы	Гарант, Консультант
9	Судебный департамент при ВС РФ	www.cdep.ru

Основная и дополнительная литература указана в Карте обеспеченности литературой.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Для материально-технического обеспечения дисциплины (модуля) используются специальные помещения. Специальные помещения представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования.

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, обеспечивающие тематические иллюстрации, соответствующие рабочим программам дисциплин. Демонстрационное оборудование представлено в виде мультимедийных средств. Учебно-наглядные пособия представлены в виде экранно-звуковых средств, печатных пособий, слайд-презентаций, видеофильмов, макетов и т.д., которые применяются по необходимости в соответствии с темами (разделами) дисциплины (модуля).

Для самостоятельной работы обучающихся помещения оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Предусмотрены помещения для хранения и профилактического обслуживания учебного оборудования.

Перечень специальных помещений ежегодно обновляется и отражается в справке о материально-техническом обеспечении основной образовательной программы.

Состав необходимого комплекта лицензионного программного обеспечения ежегодно обновляется, утверждается и отражается в справке о материально-техническом обеспечении основной образовательной программы.

№ п/п	Наименование дисциплины в соответствии с учебным планом	Наименование специальных помещений и помещений для самостоятельной работы
1	РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ	Лаборатория организации и принципов построения информационных систем; (ИЛК-1 (помещение 1001)-17 (304))

7. КАРТА ОБЕСПЕЧЕННОСТИ ЛИТЕРАТУРОЙ

Кафедра информационного права, информатики и математики

Специальность: 09.02.07 Информационные системы и программирование

Дисциплина: Разработка мобильных приложений

Курс: 3,4

№ п/п	Полное библиографическое описание
Основная литература	
1	Гагарина Л. Г. Технология разработки программного обеспечения: учебное пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Сидорова-Виснадул; под ред. Л. Г. Гагариной. — Москва: ФОРУМ : ИНФРА-М, 2021. — 400 с. — (Среднее профессиональное образование). — ISBN 978-5-8199-0812-9.
2	Маркин А. В. Программирование на SQL: учебное пособие для среднего профессионального образования / А. В. Маркин. — Москва: Издательство Юрайт, 2021. — 435 с. — (Профессиональное образование). — ISBN 978-5-534-11093-7.
3	Огнева М. В. Программирование на языке C++: практический курс: учебное пособие для среднего профессионального образования / М. В. Огнева, Е. В. Кудрина. — Москва: Издательство Юрайт, 2021. — 335 с. — (Профессиональное образование). — ISBN 978-5-534-05780-5.
4	Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2024. — 160 с. — (Профессиональное образование). — ISBN 978-5-534-16868-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/542342
5	Черников, В. Н. Разработка мобильных приложений на C# для iOS и Android : практическое руководство / В. Н. Черников. - Москва : ДМК Пресс, 2020. - 188 с. - ISBN 978-5-97060-805-0. - Текст : электронный. - URL: https://znanium.com/catalog/product/1094956
Дополнительная литература	
1	Волк, В. К. Информатика : учебное пособие для среднего профессионального образования / В. К. Волк. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 226 с. — (Профессиональное образование). — ISBN 978-5-534-18452-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/535033
2	Кедрова Г. Е. Информатика для гуманитариев: учебник и практикум для СПО. – М.: Издательство Юрайт, 2020. - 439 с.
3	Зимин В. П. Информатика. Лабораторный практикум в 2 ч. Часть 1: учебное пособие для СПО - 2-е изд., испр. и доп. – М.: Издательство Юрайт, 2020. - 126 с.
Дополнительная литература для углубленного изучения дисциплины	

1	Борисов Р.С., Скотченко А.С. Информатика: учебное пособие для СПО. – М.: Российский государственный университет правосудия, 2023. - 334 с. - ISBN: 978-5-00209-051-8 - URL: http://op.raj.ru https://znanium.ru/catalog/document?id=441348
---	---

Зав. библиотекой _____ Астраханцева О.В.

Зав. кафедрой _____ Ловцов Д.А.

8. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

8.1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

№ п/п	Раздел дисциплины (модуля), тема	Код компетенции	Наименование оценочного средства
1.	Мобильные приложения и технологии.	ОК 01- ОК 09	Вопросы по теме, реферат
2.	Создание и тестирование модулей для мобильных приложений	ОК 01- ОК 09	Вопросы по теме, реферат
3.	Разработка мобильных приложений под Android.	ПК 1.2 ПК 1.6	Фонд тестовых заданий, практические задания
4.	Разработка мобильных приложений под iOS.	ПК 1.2 ПК 1.6	Фонд тестовых заданий, практические задания
5.	Разработка мобильных приложений под WindowsPhone.	ПК 1.2 ПК 1.6	Фонд тестовых заданий, практические задания

8.2. ОЦЕНОЧНЫЕ СРЕДСТВА

ПРИМЕРЫ КОНТРОЛЬНЫХ ЗАДАНИЙ

1. Перечень компетенций (части компетенции), проверяемых оценочным средством (наименование, код):

№ п/п	Код компетенции	Название
1	ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.
2	ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.
3	ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях.
4	ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде.
5	ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
6	ОК 06.	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных российских духовно-нравственных ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.
7	ОК 07.	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.
8	ОК 08.	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.
9	ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.

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

Пример 1.

Разработайте с помощью MS Visual Studio .NET серверное веб-приложение, использующее управляющий ASP.NET Mobile элемент: <MOBILE:CALENDAR>. Добавьте для этого элемента обработчик события OnSelectionChanged, который должен отображать выбранный пользователем мобильного устройства день календаря с помощью элемента:<MOBILE:LABEL>. Создайте необходимый виртуальный каталог на веб-сервере и проверьте работу приложения с помощью эмулятора мобильного устройства.

Пример 2.

Создайте на простейшей веб-странице, отображающей на экране мобильного устройства сообщение “Hello world!”.

Пример 3.

Создать игру, реализованную как midp приложение. Создать меню игры, содержащее пункты: Запуск игры Уровень сложности и другие настройки. В игре должны подсчитываться набранные очки и лучший результат сохраняется в течение игры.

Пример 4.

Создать сервер обмена текстовыми сообщениями. Создать сервер, которые будет получать данные от клиента в виде строк и выводить на экран мидлета. Сервер должен быть многопоточным, т.е. Обслуживать одновременно с несколько клиентов. Для завершения работы клиент должен послать строку «exit». Создать клиента для сервера обмена сообщениями. Клиентское приложение должно иметь поле для ввода данных для отправки.

По выбору студента приложение может выполнять функции и сервера, и клиента одновременно, либо могут быть реализованы, как два отдельных приложения.

Пример 5.

Разработать приложение calculator с одним activity. Графическое представление приложения calculator реализовать с кнопками цифр, математических операций (сложения, вычитания, умножения, деления и др.), получения результата.

Пример 6.

Разработать приложение Taxi, состоящее из трех Activity (рисунок). В первом Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку Registration для запуска второго Activity. При нажатии на кнопку Registration выполнить явный вызов второго Activity с передачей данных о пользователе (телефон, имя и фамилия).

Во втором Activity создать два текстовых поля (TextView) для вывода переданной информации о пользователе (имя+фамилия, телефон), пустое по умолчанию текстовое поле (TextView) для вывода маршрута движения, кнопку Set path для ввода этого маршрута, кнопку вызова такси Call Taxi (недоступна, пока не введен маршрут движения). При нажатии на кнопку Set path выполнить неявный вызов третьего Activity с помощью метода startActivityForResult.

В третьем Activity создать шесть редактируемых текстовых полей (EditText) для ввода параметров маршрута движения, кнопку ОК для возврата во второе Activity. При нажатии на кнопку ОК реализовать возврат во второе Activity с передачей в качестве результата параметров маршрута движения.

После возврата во второе Activity в текстовое поле вывести информация о маршруте движения и предложение вызвать такси, кнопку вызова такси Call taxi сделать доступной. При нажатии на кнопку Call Taxi вывести всплывающее сообщение об

успешной отправке такси. Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки Registration должно программно меняться на Log in.

Пример 7.

Разработать приложение MiniShop, состоящее из двух Activity (рисунок).

В первом Activity создать список ListView с Header и Footer. В Footer разместить текстовое поле (TextView) для ввода количества активированных пользователем товаров, кнопку Show Checked Items для перехода в корзину товаров. Реализовать кастомизированный список ListView с помощью собственного адаптера, наследующего класс BaseAdapter. В каждом пункте списка отобразить следующую информацию о товаре: идентификационный номер, название, стоимость, чек-бокс для возможности выбора товара пользователем.

В текстовом поле (TextView) Footer списка динамически отображать общее текущее количество активированных товаров. При нажатии на кнопку Show Checked Items реализовать переход во второе Activity с корзиной товаров. Корзину товаров реализовать в виде нового кастомизированного списка с выбранными товарами. Продемонстрировать работу приложения MiniShop на эмуляторе или реальном устройстве.

Пример 8.

Разработать приложение MyNotes представляющее собой View Pager. Поместить в View Pager четыре фрагмента: FragmentShow, FragmentAdd, FragmentDel, FragmentUpdate. В View Pager добавить верхнее меню вкладок (PagerTabStrip) с заголовками Show, Add, Del, Update. Во фрагменте FragmentShow реализовать кастомизированный список заметок ListView с помощью собственного адаптера. В каждом пункте списка отобразить следующую информацию о заметке пользователя: номер, описание заметки. Хранение, а также предоставление информации о заметках адаптеру реализовать с помощью базы данных SQLite.

Во фрагменте FragmentAdd реализовать функционал добавления новой заметки посредством ввода описания заметки в поле EditText и добавления информации в базу данных SQLite по нажатию на кнопку Add.

Во фрагменте FragmentDel реализовать функционал удаления новой заметки посредством ввода ее номера в поле EditText и удаления информации из базы данных SQLite по нажатию на кнопку Del.

Во фрагменте FragmentUpdate реализовать функционал обновления существующей заметки посредством ввода ее номера в поле EditText, ввода нового описания в поле EditText и обновления информации в базе данных SQLite по нажатию на кнопку Update.

Предусмотреть обработку исключительной ситуации отсутствия заметки по указанному номеру посредством вывода пользователю всплывающего сообщения соответствующего содержания. Продемонстрировать работу приложения MyNotes на эмуляторе или реальном устройстве.

Пример 9.

Требуется разработать приложение с графическим пользовательским интерфейсом, поддерживающее создание/редактирование/удаление/поиск заметок. Два варианта хранения заметок: А) в базе SQLite. Б) С использованием файловой системы.

Пример 10.

Создать приложение с графическим пользовательским интерфейсом с функциями: Определение местоположения пользователя на карте Google Map; Определение скорости и направления движения пользователя; Масштабирование карты. Программа должна быть конфигурируемой. Настройки: Режим определения местоположения (через GPS либо по сотам); Включение/отключение режима поиска.

Пример 11.

Разработать приложение-таймер с использованием датчика ориентации в виде песочных часов. Каждый раз для того чтобы активировать таймер, необходимо перевернуть экран мобильного устройства вверх ногами. Используйте анимацию для показа «перетекающего песка» и переворота песочных часов. Для задания времени перетекания песка требуется разработать push-notification сервер. Через форму ввода на сервере можно отправлять клиенту (приложение-таймер) в указанное время (числовой ввод).

Пример 12.

Разработка программы для обмена мгновенными сообщениями. Требуется разработать приложение для обмена мгновенными сообщениями через Wi-Fi/Bluetooth. Поддерживаемые режимы: 1. Активный режим. Приложение занимает весь экран, содержит поля для отправки сообщений и список принятых сообщений. 2. Режим уведомлений. Приложение через уведомления показывает принятые сообщения.

Пример 13.

Разработка мобильного сайта с адаптивным дизайном. Требуется разработать сайт, пригодный для просмотра на смартфонах и планшетах, с использованием принципов адаптивного дизайна: 1) Гибкая сетка. 2) Пропорциональные шрифты. 3) Масштабируемые изображения. 4) Медиа-запросы.

2. Критерии оценивания:

Критерии	Баллы
На теоретические вопросы даны практически полные ответы. Задание выполнено полностью без ошибок, могут присутствовать отдельные недочеты.	3 – 6 зачет
Получены неполные ответы. Задание не выполнено полностью.	0 – 3 незачет

ТЕМЫ РЕФЕРАТОВ (ЭССЕ, ДОКЛАДОВ, СООБЩЕНИЙ)

1. Перечень компетенций (части компетенции), проверяемых оценочным средством (наименование, код):

№ п/п	Код компетенции	Название
1	ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.
2	ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.
3	ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях.
4	ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде.
5	ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
6	ОК 06.	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных российских духовно-нравственных ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.
7	ОК 07.	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.
8	ОК 08.	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.
9	ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.

2. Перечень тем рефератов (эссе, докладов, сообщений):

№ п/п	Тема	Код компетенции (части) компетенции
1.	Описать конструктор для разработки мобильных приложений на Android	ОК 01-ОК 09
2.	Анализ мобильных приложений систем мониторинга	ОК 01-ОК 09
3.	Описать этапы разработки мобильного приложения	ОК 01-ОК 09
4.	Теоретические основы разработки мобильных приложений.	ОК 01-ОК 09
5.	Идея нового приложения мобильного	ОК 01-ОК 09
6.	Обзор и сравнение мобильных приложений для голосового общения	ОК 01-ОК 09
7.	Обзор приложений для управления «умным домом», бытовой техникой. Примеры автоматизации дома	ОК 01-ОК 09
8.	Перечислить сенсоры (датчики), которые присутствуют в современных телефонах, найти приложения, которые их используют.	ОК 01-ОК 09

9.	Сравнение приложений для пост-обработки фотографий (эффекты, публикация), не менее 5 популярных приложений.	ОК 01-ОК 09
10.	Операционные системы для мобильных устройств (появление, особенности, доли рынка на текущий момент).	ОК 01-ОК 09
11.	Разработка мобильного приложения «Расписание студента»	ОК 01-ОК 09
12.	Разработка мобильного приложения «Зачетная книжка студента»	ОК 01-ОК 09
13.	Разработка мобильного приложения «Студенческая группа»	ОК 01-ОК 09
14.	Разработка мобильного приложения для интернет-магазина	ОК 01-ОК 09
15.	Разработка мобильного приложения для книжного магазина учебной литературы	ОК 01-ОК 09
16.	Разработка мобильного приложения для агентства недвижимости	ОК 01-ОК 09
17.	Разработка мобильного приложения для управляющей компании ЖКХ	ОК 01-ОК 09
18.	Разработка мобильного приложения для кредитной организации	ОК 01-ОК 09

3. Критерии оценивания:

Критерии	Баллы
Тема реферата раскрыта полностью. Материал изложен уверенно и грамотно. Присутствует презентационный материал по теме.	9-10
Тема реферата раскрыта не полностью. Материал изложен. Присутствует презентационный материал.	7-8
Тема реферата раскрыта не полностью. Материал изложен не уверенно и без грамотно. Присутствует презентационный материал не по теме.	5-6
Тема реферата не раскрыта. Присутствует презентационный материал не по теме.	3-4
Реферат по теме не выполнен.	0-2

4. Методические рекомендации по написанию

В часы, отведённые для НИРС, целесообразно подготовить реферат или научную статью для апробации на Ежегодной итоговой научной конференции студентов Российского государственного университета правосудия (традиционно проводится в апреле месяце) на заседании секции «Информационное право». Лучшие работы могут быть рекомендованы для публикации в сборнике конференции.

Рефераты (проекты статей) должны быть выполнены на компьютере, оформлены в соответствии с методическими рекомендациями по оформлению письменных работ и в *обязательном* порядке должны содержать титульный лист, рубрики: содержание (оглавление), введение, основную часть, заключение (выводы), список литературы (включая обязательно литературу кафедры и академии согласно УМК по учебной дисциплине), содержащий не менее трёх наименований со *ссылками* в тексте). Объём реферата: от 5 до 12 страниц машинописного текста (1800 знаков на странице, гарнитура *Times New Roman*).

На все литературные источники в тексте реферата (статьи) должны быть ссылки в виде: [N], где N – номер источника в библиографии (списке литературы). Список литературы должен соответствовать требованиям ГОСТ 7.1–2003 – «Библиографическое описание».

ТЕСТОВЫЕ ЗАДАНИЯ

СОДЕРЖАНИЕ БАНКА ТЕСТОВЫХ ЗАДАНИЙ

№ п/п	Раздел дисциплины (модуля), тема	Код компетенции
1	Разработка мобильных приложений под Android.	ПК 1.2 ПК 1.6
2	Разработка мобильных приложений под iOS.	ПК 1.2 ПК 1.6
3	Разработка мобильных приложений под WindowsPhone.	ПК 1.2 ПК 1.6

1. S: Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется:
 - +: Android SDK
 - : JDK
 - : плагин ADT
 - : Android NDK
2. S: С какой целью был создан Open Handset Alliance?
 - : писать историю развития ОС Android
 - : продавать смартфоны под управлением Android
 - : рекламировать смартфоны под управлением Android
 - +: разрабатывать открытые стандарты для мобильных устройств
3. S: С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) System Analyzer используется в среде разработки Intel* Beacon Mountain?
 - +: позволить разработчикам оптимизировать загрузку системы при использовании процедур OpenGL
 - : для ускорения работы эмулятора в среде разработки
 - : для оптимизированной обработки данных и изображений
 - : позволить разработчикам эффективно распараллелить C++ мобильные приложения
4. S: Библиотеки, реализованные на базе PacketVideo OpenCORE:
 - +: Media Framework
 - : SQLite
 - : FreeType
 - : 3D библиотеки
5. Какой движок баз данных используется в ОС Android?
 - : InnoDB
 - : DBM
 - : MyISAM
 - +: SQLite
6. S: С какой целью инструмент Intel* Integrated Performance Primitives (Intel* IPP) используется в среде разработки Intel* Beacon Mountain?
 - +: для оптимизированной обработки данных и изображений
 - : позволить разработчикам оптимизировать загрузку системы при использовании процедур OpenGL
 - : для ускорения работы эмулятора в среде разработки

- : позволить разработчикам эффективно распараллелить C++ мобильные приложения
- 7. S: Intel XDK поддерживает разработку под:
 - : JavaFX Mobile
 - : Apple iOS, BlackBerry OS
 - : MtkOS, Symbian OS. Microsoft Windows 8
 - +: Android. Apple iOS, Microsoft Windows 8, Tizen
- 8. S: Каждый приемник широковещательных сообщений является наследником класса
 - : ViewReceiver
 - : IntentReceiver
 - : ContentProvider
 - +: BroadcastReceiver
- 9. S: Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (ТЛ), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий?
 - : GUI
 - +: View
 - : UIComponent
 - : Widget
- 10. S: Какой слушатель используется для отслеживания события касания экрана устройства?
 - : OnPressListener
 - +: onTouchListener
 - : OnClickListener
 - : OnInputListener
- 11. S: В какой папке необходимо разместить XML файлы, которые определяют все меню приложения?
 - : res/value
 - : res/items
 - : res/layout
 - +: res/menu
- 12. S: Фоновые приложения ...
 - +: после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии
 - : выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями
 - : небольшие приложения, отображаемые в виде графического объекта на рабочем столе
 - : большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки
- 13. S: Полный иерархический список обязательных файлов и папок проекта можно увидеть на вкладке ...
 - +: Package Explorer
 - : Internet Explorer
 - : Navigator
 - +: Project Explorer
- 14. S: Какой компонент управляет распределенным множеством данных приложения?
 - : сервис (Service)
 - : активность (Activity)
 - : приемник широковещательных сообщений (Broadcast Receiver)

- +: контент-провайдер (Content Provider)
15. S: Какой язык разметки используется для описания иерархии компонентов графического пользовательского интерфейса Android-приложения?
 -: html
 +: xml
 -: gml
 -: xhtml
16. S: Выберите верную последовательность действий, необходимых для создания в приложении контент-провайдера.
 -: Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными
 -: Проектирование способа хранения данных; Определение способа организации данных;
 -: Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными
 +: Проектирование способа хранения данных: Создание класса-наследника от класса ContentProvider: Определение строки авторизации провайдера. URI для его строк и имен столбцов
17. S: Выберите верные утверждения относительно объекта-намерения (Intent).
 -: представляет собой структуру данных, о обычно используется для запуска активности или сервиса
 +: используется для передачи сообщений пользователю
 -: используется для получения инструкций от пользователя
 +: используются для передачи сообщений между основными компонентами приложений?
18. S: Расположение элементов мобильного приложения:
 -: полезно для передачи иерархии
 -: влияет на удобство использования
 -: полезно для создания пространственных отношений между объектами на экране и объектами реального мира
 +: все варианты ответа верны
19. S: Какие элементы управления применяются для действий по настройке?
 -: командные элементы управления
 +: элементы выбора
 -: элементы ввода
 -: элементы отображения
20. S: Примерами комбо-элементов не являются:
 -: комбо-список
 -: все вышеперечисленное
 -: комбо-кнопка
 +: комбо-поле
21. S: Дизайн или проектирование интерфейса для графических дизайнеров
 +: все варианты ответа верны
 -: прозрачность и понятность информации
 -: тон, стиль, композиция, которые являются атрибутами бренда
 -: передача информации о поведении посредством ожидаемого назначения
22. S: Более крупные элементы:
 +: привлекают больше внимания
 -: все варианты ответа верны
 -: размер не влияет на уровень внимания
 -: привлекают меньше внимания
23. S: К традиционным типографическим инструментам не относят

- : масштаб
 - +: цвет
 - : разреженность
 - : выравнивание по сетке
24. S: К элементам ввода относят:
- : ограничивающие элементы ввода Б) ползунки
 - : счетчики
 - +: все вышеперечисленное
25. S: Выделяют следующие категории плотности экрана для Android-устройств
- : LDPI, MDPI, HDPI, XHDPI, и XXXHDPI
 - : правильный вариант ответа отсутствует
 - +: LDPI, MDPI, HDPI, XHDPI, и XXXHDPI
 - : LDPI, MDPI, HDPI
26. S: Следующие утверждения не верны:
- +: не используйте интерфейсные элементы
 - : картинки работают быстрее, чем слова
 - : на любом шаге должна быть возможность вернуться назад
 - : если объекты похожи, они должны выполнять сходные действия
27. S: Следующие утверждения верны:
- : текстура бесполезна для передачи различий или привлечения внимания
 - : восприятие направления затруднено при больших размерах объектов
 - +: все варианты ответа верны
 - : люди легко воспринимают контрастность
28. S: Основные вкладки (FixedTabs) удобны при отображении
- : от четырех вкладок
 - : двух вкладок
 - : трех и более вкладок
 - +: трех и менее вкладок
29. S: Диалоговое окно, содержащее линейку процесса выполнения какого-то действия это
- : DatePickerDialog
 - : AlertDialog
 - +: ProgressDialog
 - : DialogFragment
30. S: Уведомления стоит использовать, когда
- : сообщение не требует ответа пользователя, но важно для продолжения его работы
 - : сообщение является важным и требует немедленного прочтения и ответа
 - : сообщение является важным, однако требует немедленного прочтения, но не ответа
 - +: сообщение является важным, однако не требует немедленного прочтения и ответа
31. S: Какой метод запускает новую активность?
- +: startActivity()
 - : beginActivity()
 - : intentActivity()
 - : newActivity()
32. S: ProgressDialog это:
- : контейнер для создания собственных диалоговых окон
 - : диалоговое окно с predefined интерфейсом, позволяющее выбрать дату или время

- + : диалоговое окно, содержащее линейку процесса выполнения какого-то действия
 - : диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое
33. S: AlertDialog это:
- : контейнер для создания собственных диалоговых окон
 - : диалоговое окно, содержащее линейку процесса выполнения какого-то действия
 - + : диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое
 - : диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время
34. S: Что необходимо сделать при добавлении в проект новой активности?
- : скачать и установить специальный инструмент MultiActivity SDK
 - + : прописать в манифесте информацию о новой активности
 - : создать новый проект
 - : запустить эмулятор
35. S: Системы позиционирования смартфона могут включать
- + : все перечисленное
 - : систему GPS
 - : систему ГЛОНАСС
 - : сигналы WiFi и Bluetooth
36. S: Какая константа не определена в классе MotionEvent, для обозначения сенсорных событий
- : ACTIONDOWN
 - : ACTIONUP
 - + : ACTION_CLICK
 - : ACTION_MOVE
37. S: С какой целью используется метод releaseQ в классах MediaPlayer и MediaRecorder?
- + : конец жизненного цикла объекта и освобождение ресурсов
 - : перевод объекта в ожидающее состояние
 - : обновление объекта и запуск его работы
 - : создание объекта и запуск его работы
38. S: К датчикам окружающей среды, встроенным в мобильное устройство относят
- : датчики вектора вращения
 - + : датчики освещенности
 - : акселерометры
 - : гироскопы
39. S: Библиотека Universal Image Loader for Android позволяет:
- : парсить HTML-страницы
 - : строить графики и диаграммы
 - + : загружать, кешировать и отображать изображения
 - : использовать анимацию, доступную только с версии 3.x, на более ранних вариантах платформы Android
40. S: Facebook SDK for Android — это библиотека, позволяющая:
- : получать доступ к информации любого пользователя
 - : отправлять рекламные сообщения от имени пользователя
 - + : писать сообщения на стену, читать и менять статусы, смотреть ленту друзей
 - : парсить страницы пользователей
41. S: Что из перечисленного не относится к правилам безопасности при подключении библиотек?

- : с осторожностью использовать библиотеки из сомнительных источников
 - : ознакомиться с форумами и сайтами, где могут обсуждаться библиотеки
 - +: лично познакомиться с разработчиками библиотеки
 - : использовать скомпрометированные библиотеки
42. S: Библиотека MapNavigator предназначена для:
- : работы с любыми картами
 - : работы с Яндекс.Картами
 - : морской навигации
 - +: работы с картами Google Maps
43. S: Библиотека jsoup не позволяет:
- : находить и извлекать данные, используя DOM и селекторы CSS
 - : манипулировать HTML-элементами, атрибутами и текстом
 - +: писать сообщения на стену, читать и менять статусы, смотреть ленту друзей
 - : принимать в качестве параметра URL. файл или строк.
44. S: При настройке обратной совместимости необходимо добавить в файл манифеста следующую информацию:
- : только минимальную версию Android SDK
 - +: минимальную и основную ('целевую') версии Android SDK
 - : информацию о подключенной библиотеке
 - : только основную (целевую) версии Android SDK
45. S: Какая библиотека предназначена для упрощения загрузки изображений?
- : Yandex.Metrica for Apps;
 - +: Universal Image Loader for Android
 - : ActionBarSherlock
 - : NineOldAndroids
46. S: Библиотеки совместимости предназначены для
- : сбора статистики
 - : рисования графиков
 - +: использования возможностей, появившиеся в какой-то версии ОС Android, на более ранних версиях платформы
 - : подключения нестандартных элементов управления
47. S: Какая библиотека предназначена для использования анимации?
- : Universal Image Loader for Android
 - +: NineOldAndroids
 - : Yandex.Metrica for Apps
 - : ActionBarSherlock
48. S: Для чего служит папка res/anim/ проекта?
- : в этой папке находятся файлы, содержащие набор картинок, предназначенных для кадровой анимации
 - : в этой папке находятся файлы, содержащие анимированные ролики для воспроизведения в приложении
 - : в этой папке находятся XML файлы, задающие реализацию анимации свойств
 - +: в этой папке находятся XML файлы, задающие последовательность инструкций анимации преобразований
49. S: В какой файл обязательно добавляется информация при создании нового Activity в приложении?
- +: AndroidManifest.xml
 - : main.java
 - : lay out.xml
 - : activity.xml
50. S: Какой метод жизненного цикла активности вызывается системой непосредственно перед появлением активности на экране?

- : onVisible()
 - : onOpen()
 - : onResume()
 - +: onCreate()
51. S: С какой целью используется метод SurfaceHolder.lockCanvas()?
- +: блокировка Canvas для перерисовки
 - : игнорирование дальнейшего взаимодействия с Canvas
 - : сокрытие Canvas
 - : блокировка Canvas от сворачивания
52. S: Может ли мобильное приложение получить доступ к базе данных, созданной в другом приложении?
- : не может ни при каких обстоятельствах
 - +: может, но только с помощью контент-провайдеров
 - : право на доступ открывает приложение-хозяин базы данных
 - : может обращаться напрямую
53. S: С помощью какого метода можно запретить смену ориентации устройства, при запущенном приложении?
- +: setRequestedOrientation
 - : setChangeOrientation
 - : disableChangeOrientation
 - : setOrientation
54. S: Какой из датчиков не используется для определения положения смартфона в пространстве?
- : акселерометр
 - +: gps
 - : гироскоп
 - : магнитометр
55. S: К новым возможностям HTML5 относят (выберите все верные варианты ответа):
- +: возможность добавления аудио и видео без использования вспомогательных средств
 - +: возможность рисования на холсте
 - : возможность прямого доступа к оперативной памяти
 - : форматирование данных в режиме таблицы
56. S: Возможен ли перенос приложений iOS* в среду HTML5:
- : нет, прямой перенос приложений невозможен
 - +: да, используя средства Intel XDK
 - : да, используя только сторонние средства
 - : да, только для iPhone, используя средства Intel XDK
57. S: Следующие утверждения верны:
- : JavaScript не позволяет подключать другие внешние библиотеки, написанные на других языках
 - : приложения html5 исполняются быстрее и требуют меньше ресурсов, чем «нативные»
 - : среда Intel XDK не работает с мультисенсорностью
 - +: приложения html5 исполняются медленнее и требуют больших ресурсов, чем «нативные»
58. S: Разрабатывать приложения в среде Intel XDK можно:
- : пользоваться заготовленными примерами
 - +: все варианты ответа верны
 - : «с нуля», прописывая все элементы
 - : использовать встроенный «дизайнер элементов» для отрисовки элементов

59. S: JavaScript не позволяет:
- +: получать прямой доступ к памяти
 - : работать с реестром
 - : работать с картами
 - : одновременно использовать несколько подключаемых библиотек
60. S: В среде Intel XDK можно разрабатывать приложения для следующих платформ:
- : Android
 - +: все варианты ответа верны
 - : Apple iOS
 - : Tizen
61. S: В заготовке любого приложения, разрабатываемого в среде Intel XDK прописана:
- +: все варианты ответа верны
 - : скрытие окна заставки Intel XDK
 - : настройка размеров приложения под размеры устройства
 - : фиксация размеров приложения (запрет «скроллинга»)
62. S: Создавать и редактировать пользовательский интерфейс приложений в среде Intel XDK можно:
- : используя встроенное приложение App Designer
 - : только изменяя готовые шаблоны с интерфейсом
 - +: все варианты ответа неверны
 - : только прописывая теги вручную
63. S: Удобное средство обмена между двумя NFC-устройствами
- : Wi-Fi Direct
 - +: AndroidBeam
 - : Dalvik
 - : Bluetooth.
64. S: Переключения между активностями осуществляются
- : только при помощи кнопок
 - : только с использованием сенсорного экрана смартфона
 - : только при помощи кнопок и других элементов управления
 - +: все три варианта возможны

ВОПРОСЫ, ВЫНОСИМЫЕ НА ЗАЧЕТ (ЭКЗАМЕН), ПО ДИСЦИПЛИНЕ

1. Понятие защищенной информационной системы.
2. Свойства защищенной ОС.
3. Безопасность информационных систем в нормативных документах.
4. Классификация защищенности ОС по международным стандартам.
5. Политика безопасности, формальное представление политик.
6. Классификация изъянов защиты.
7. Категории изъянов защиты в ОС.
8. Мобильное программирование, платформы для разработки.
9. Основы работы с сенсорным вводом. Обработка нескольких касаний. Использование изображений.
10. Акселерометр и служба определения местоположения, вторичные потоки выполнения, обработка асинхронных операций и доступ к Веб-сервисам.
11. Особенности использования pivot и panorama.
12. Краткая история ОС Android.
13. Intel для Android: партнерство и инструментарий разработчика.
14. Архитектура приложений для Android. Ресурсы приложения.

15. Пользовательский интерфейс. Инструментарий разработки приложений для Android.
16. Обзор шагов разработки типичного приложения под Android. Особенности разработки с использованием эмулятора. Отладка кода в эмуляторе и на реальных приложениях.
17. Пример простейших программ Android-приложения. Запуск приложения на эмуляторе.
18. Тестирование приложения с помощью Dalvik Debug Monitor Server (DDMS).
19. Планирование покадровой анимации, анимирование, анимация шаблонов, видов, использование класса Camera.
20. Проверка безопасности, работа со службами, основанными на местоположении, использование HTTP-служб, службы AIDL.
21. Подготовка AndroidManifest.xml для загрузки, локализация приложения, подготовка ярлыка приложения, подготовка APK-файла для загрузки, работа пользователя с Android Market.
22. Работа с инструментами Intel для оптимизации отладки Android приложений.
23. Краткая история iOS.
24. Пример простейших программ iOS-приложения. Запуск приложения на эмуляторе.
25. Пользовательский интерфейс. Инструментарий разработки приложений для iOS.
26. Обзор шагов разработки типичного приложения под iOS. Особенности разработки с использованием эмулятора. Отладка кода в эмуляторе и на реальных приложениях.
27. Основные платформы мобильных приложений, сравнительная характеристика.
28. Нативные приложения, веб-приложения, гибридные и кроссплатформенные приложения, их области применения.
29. Основные языки для разработки мобильных приложений (Java, ObjectiveC и др.)
30. Инструменты разработки мобильных приложений (JDK/AndroidStudio/WebView/Phonegap и др.)
31. Инструментарий среды разработки мобильных приложений. Структура типичного мобильного приложения.
32. Архитектура платформы Android. Dalvik Virtual Machine. Среда разработки для Android. Eclipse IDE. II
33. Плагин ADT. Android Virtual Device.
34. Android SDK. Версии SDK и Android API Level.
35. Инструментарий среды разработки мобильных приложений
36. Структура типичного мобильного приложения
37. Элементы управления и контейнеры
38. Работа со списками
39. Способы хранения данных
40. Структура проекта Android-приложения в Eclipse. Каталоги ресурсов. Файл R.java
41. Графический интерфейс пользователя в Android- приложениях. XML разметка интерфейса.
42. XML-разметка интерфейса пользователя
43. Ресурсы в Android-приложениях
44. Ресурсы в Windows Phone-приложениях.
45. Многопоточные приложения в Android и Windows Phone. Использование системных таймеров и системного времени.
46. Использование объектов Intent. Intent-фильтры
47. Использование ресурсов. Ссылки на ресурсы. Загрузка простых типов из ресурсов. Загрузка файлов произвольного типа.
48. Пользовательские настройки. Использование SharedPreferences. Виды настроек. Службы в Android. Компонент Service.
49. Датчики мобильных устройств. Управление датчиками в приложении. Виды датчиков и особенности их использования

50. Программный доступ к дисплею устройства. Менеджер окон. Параметры дисплея.

Критерии оценки контрольного задания

Критерии	Баллы
Высокий уровень подготовки, безупречное владение материалом, студент демонстрирует творческий подход к решению нестандартных ситуаций. Студент дал полный и развернутый ответ на все теоретические вопросы, подтверждая теоретический материал практическими примерами. Студент активно работал на практических занятиях.	23-25
В целом хорошая подготовка с заметными ошибками или недочетами. Студент дает полный ответ на все вопросы, но имеются неточности в определениях понятий, процессов и т.п. Допускаются ошибки при ответах на вопросы контрольного задания. Студент работал на практических занятиях.	18-22
Минимально достаточный уровень подготовки. Студент показывает минимальный уровень знаний, делает существенные ошибки, но при ответах на вопросы контрольного задания, может правильно сориентироваться и в общих чертах дать правильный ответ. Студент посещал практические занятия.	11-17
Подготовка недостаточная и требует дополнительного изучения материала. Студент дает ошибочные ответы, в том числе на вопросы контрольного задания. Студент пропустил большую часть практических занятий.	10 и менее

Зачтено	11-17 баллов – удовлетворительно, 18-22 баллов – хорошо, 23-25 баллов – отлично
Не зачтено	10 баллов – неудовлетворительно

ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАДАЧ-ПРИМЕРОВ

ПРИМЕР 1. Разработайте с помощью MS Visual Studio .NET серверное веб-приложение, использующее управляющий ASP.NET Mobile элемент: <MOBILE:CALENDAR>. Добавьте для этого элемента обработчик события OnSelectionChanged, который должен отображать выбранный пользователем мобильного устройства день календаря с помощью элемента:<MOBILE:LABEL>.

РЕШЕНИЕ:Код обработчика события может выглядеть следующим образом:

```
<script runat="server">
```

```
protected void CalChanged (Object sender, EventArgs e)
```

```
Label1.Text="You selected:"+calendar1/SelectedDate;
```

```
</script>
```

Создайте необходимый виртуальный каталог на веб-сервере и проверьте работу приложения с помощью эмулятора мобильного устройства.

ПРИМЕР 2. Создайте на простейшей веб-странице, отображающей на экране мобильного устройства сообщение "Hello world!".

ПРИМЕР 3. Создать игру, реализованную как midp приложение. Создать меню игры, содержащее пункты: Запуск игры Уровень сложности и другие настройки. В игре должны подсчитываться набранные очки и лучший результат сохраняется в течение игры.

ПРИМЕР 4. Создать сервер обмена текстовыми сообщениями. Создать сервер, который будет получать данные от клиента в виде строк и выводить на экран мидлета.

Сервер должен быть многопоточным, т.е. Обслуживать одновременно с несколько клиентов. Для завершения работы клиент должен послать строку «exit». Создать клиента для сервера обмена сообщениями. Клиентское приложение должно иметь поле для ввода данных для отправки.

По выбору студента приложение может выполнять функции и сервера, и клиента одновременно, либо могут быть реализованы, как два отдельных приложения.

ПРИМЕР 5. Разработать приложение calculator с одним activity. Графическое представление приложения calculator реализовать с кнопками цифр, математических операций (сложения, вычитания, умножения, деления и др.), получения результата.

ПРИМЕР 6. Разработать приложение Taxi, состоящее из трех Activity (рисунок). В первом Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку Registration для запуска второго Activity. При нажатии на кнопку Registration выполнить явный вызов второго Activity с передачей данных о пользователе (телефон, имя и фамилия).

Во втором Activity создать два текстовых поля (TextView) для вывода переданной информации о пользователе (имя+фамилия, телефон), пустое по умолчанию текстовое поле (TextView) для вывода маршрута движения, кнопку Set path для ввода этого маршрута, кнопку вызова такси Call Taxi (недоступна, пока не введен маршрут движения). При нажатии на кнопку Set path выполнить неявный вызов третьего Activity с помощью метода startActivityForResult.

В третьем Activity создать шесть редактируемых текстовых полей (EditText) для ввода параметров маршрута движения, кнопку ОК для возврата во второе Activity. При нажатии на кнопку ОК реализовать возврат во второе Activity с передачей в качестве результата параметров маршрута движения.

После возврата во второе Activity в текстовое поле вывести информация о маршруте движения и предложение вызвать такси, кнопку вызова такси Call taxi сделать доступной. При нажатии на кнопку Call Taxi вывести всплывающее сообщение об успешной отправке такси. Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки Registration должно программно меняться на Log in.

ПРИМЕР 7. Разработать приложение MiniShop, состоящее из двух Activity (рисунок).

В первом Activity создать список ListView с Header и Footer. В Footer разместить текстовое поле (TextView) для ввода количества активированных пользователем товаров, кнопку Show Checked Items для перехода в корзину товаров. Реализовать кастомизированный список ListView с помощью собственного адаптера, наследующего класс BaseAdapter. В каждом пункте списка отобразить следующую информацию о товаре: идентификационный номер, название, стоимость, чек-бокс для возможности выбора товара пользователем.

В текстовом поле (TextView) Footer списка динамически отображать общее текущее количество активированных товаров. При нажатии на кнопку Show Checked Items реализовать переход во второе Activity с корзиной товаров. Корзину товаров реализовать в виде нового кастомизированного списка с выбранными товарами. Продемонстрировать работу приложения MiniShop на эмуляторе или реальном устройстве.

ПРИМЕР 8. Разработать приложение MyNotes представляющее собой View Pager. Поместить в View Pager четыре фрагмента: FragmentShow, FragmentAdd, FragmentDel, FragmentUpdate. В View Pager добавить верхнее меню вкладок (PagerTabStrip) с заголовками Show, Add, Del, Update. Во фрагменте FragmentShow реализовать кастомизированный список заметок ListView с помощью собственного адаптера. В каждом пункте списка отобразить следующую информацию о заметке пользователя:

номер, описание заметки. Хранение, а также предоставление информации о заметках адаптеру реализовать с помощью базы данных SQLite.

Во фрагменте FragmentAdd реализовать функционал добавления новой заметки посредством ввода описания заметки в поле EditText и добавления информации в базу данных SQLite по нажатию на кнопку Add.

Во фрагменте FragmentDel реализовать функционал удаления новой заметки посредством ввода ее номера в поле EditText и удаления информации из базы данных SQLite по нажатию на кнопку Del.

Во фрагменте FragmentUpdate реализовать функционал обновления существующей заметки посредством ввода ее номера в поле EditText, ввода нового описания в поле EditText и обновления информации в базе данных SQLite по нажатию на кнопку Update.

Предусмотреть обработку исключительной ситуации отсутствия заметки по указанному номеру посредством вывода пользователю всплывающего сообщения соответствующего содержания. Продемонстрировать работу приложения MyNotes на эмуляторе или реальном устройстве.

ПРИМЕР 9. Требуется разработать приложение с графическим пользовательским интерфейсом, поддерживающее создание/редактирование/удаление/поиск заметок. Два варианта хранения заметок: А) в базе SQLite. Б) С использованием файловой системы.

ПРИМЕР 10. Создать приложение с графическим пользовательским интерфейсом с функциями: Определение местоположения пользователя на карте Google Map; Определение скорости и направления движения пользователя; Масштабирование карты. Программа должна быть конфигурируемой. Настройки: Режим определения местоположения (через GPS либо по сотам); Включение/отключение режима поиска.

ПРИМЕР 11. Разработать приложение-таймер с использованием датчика ориентации в виде песочных часов. Каждый раз для того чтобы активировать таймер, необходимо перевернуть экран мобильного устройства вверх ногами. Используйте анимацию для показа «перетекающего песка» и переворота песочных часов. Для задания времени перетекания песка требуется разработать push-notification сервер. Через форму ввода на сервере можно отправлять клиенту (приложение-таймер) в указанное время (числовой ввод).

ПРИМЕР 12. Разработка программы для обмена мгновенными сообщениями. Требуется разработать приложение для обмена мгновенными сообщениями через Wi-Fi/Bluetooth. Поддерживаемые режимы: 1. Активный режим. Приложение занимает весь экран, содержит поля для отправки сообщений и список принятых сообщений. 2. Режим уведомлений. Приложение через уведомления показывает принятые сообщения.

ПРИМЕР 13. Разработка мобильного сайта с адаптивным дизайном. Требуется разработать сайт, пригодный для просмотра на смартфонах и планшетах, с использованием принципов адаптивного дизайна: 1) Гибкая сетка. 2) Пропорциональные шрифты. 3) Масштабируемые изображения. 4) Медиа-запросы.

Ответ обучающегося на экзамене оценивается по следующей шкале:

Критерии	Баллы
На теоретические вопросы даны практически полные ответы. Задание выполнено полностью без ошибок, могут присутствовать отдельные недочеты.	от 16 до 60 баллов - зачтено
Получены неполные ответы. Задание не выполнено полностью.	15 и менее баллов - не зачтено