

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ



УТВЕРЖДАЮ
Декан ФИиВТ

УТВЕРЖДАЮ /А.А. Кречетов/
(Ф.И.О. декана (директора института))

15.01.2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б.1.2.11 Программирование мобильных устройств

(код и наименование дисциплины по учебному плану)

Направление подготовки
(специальность)

09.03.01 Информатика и вычислительная техника

Квалификация выпускника

Бакалавр

(бакалавр/магистр/специалист)

Направленность

Вычислительные машины, комплексы, системы и сети

Курс 4
Семестр 7

Распределение учебного времени

Трудоемкость по учебному плану	108 / 3	часов/зачетных единиц
Лекции	16	часов
Лабораторные работы	32	часов
Практические занятия	-	часов
Иная контактная работа	-	часов
Всего контактной работы (без учета экз.)	48	часов
Контактная работа по экзамену	-	часов
Курсовой проект (работа)	-	семестр
Самостоятельная работа обучающихся (без учета экз.)	60	часов
Самостоятельная работа по подготовке к экзамену	-	часов
Экзамен	-	семестр
Зачет	-	семестр
БРК, ДЗ	7	семестр

(год)

Программа составлена в соответствии с требованиями ФГОС ВО направления подготовки (специальности) 09.03.01 Информатика и вычислительная техника

Программу составили:

доцент с ученой степенью кандидата наук	ИВС	СОГЛАСОВАНО	В.Б. Малашкевич
(должность)	(кафедра)		(И.О. Фамилия)

РАССМОТРЕНА и ОДОБРЕНА на заседании кафедры, за которой закреплена дисциплина
Кафедра информационно-вычислительных систем

		(наименование кафедры)	
14.01.2025	протокол №	19	
(дата)			
Заведующий кафедрой	СОГЛАСОВАНО	Д.В. Морохин	
		(И.О. Фамилия)	

Рабочая программа СОГЛАСОВАНА с факультетом (институтом), выпускающей(ими)
кафедрой(ами).
СООТВЕТСТВУЕТ действующей ОП.

Заведующий кафедрой	СОГЛАСОВАНО	Д.В. Морохин
		(И.О. Фамилия)

Председатель методической комиссии факультета (института), в который входит
выпускающая кафедра

СОГЛАСОВАНО	А.А. Кречетов
	(И.О. Фамилия)

Эксперт(ы): Чернов Михаил Павлович, Заместитель генерального директора по
производству ЗАО СКБ "Хроматэк"

Рабочая программа проверена и зарегистрирована в УМЦ 04.02.2025 г.

Специалист учебно-методического центра СОГЛАСОВАНО /Т.А. Смирнова/

Раздел 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является достижение планируемых результатов обучения, соответствующих установленным в ОПОП индикаторам достижения компетенций:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения
1. ПК-1 Способен выполнять работы и управлять работами по проектированию, созданию, модификации и сопровождению ИС	ПК-1.1. Знать: принципы функционирования информационно-коммуникационных систем, методы рас-чета количественных оценок информации для информационной системы	знания: Знать принципы функционирования информационно-коммуникационных систем, методы рас-чета количественных оценок информации для информационной системы умения: навыки:
	ПК-1.2. Уметь: осуществлять поиск информации об условиях использования и возможностях предлагаемых информационно-коммуникационных систем, их составляющих и комплектующих	знания: умения: Уметь осуществлять поиск информации об условиях использования и возможностях предлагаемых информационно-коммуникационных систем, их составляющих и комплектующих навыки:
	ПК-1.3. Владеть: навыками определения количества информации в информационной системе, полосы пропускания для передачи сигналов с заданной погрешностью	знания: умения: навыки: Владеть навыками определения количества информации в информационной системе, полосы пропускания для передачи сигналов с заданной погрешностью

Раздел 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к части, формируемой участниками образовательных отношений ОПОП.

Дисциплина является обязательной

Для продолжения формирования заявленных компетенций необходимы знания предшествующих дисциплин: Моделирование (ПК-1), Интернет программирование (ПК-1)
Изучаемая дисциплина является основой для продолжения формирования указанных компетенций в следующих государственной итоговой аттестации в форме: Подготовка к процедуре защиты и защита выпускной квалификационной работы (ПК-1)

Раздел 3. ОПИСАНИЕ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

Для формирования заявленных компетенций используются методологические технологии, реализующие деятельностный, личностно-ориентированный, практико-ориентированный подходы.

Основными стратегическими технологиями являются: лекционные занятия, практические и лабораторные занятия, процедуры самообучения

На достижение конкретных целей обучения направлены применяемые тактические технологии: классическая лекция, задания

Раздел 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

7 семестр

Виды и темы занятий	Количество часов	Формируемые компетенции
Платформы и операционные системы мобильных систем	28	ПК-1
Лекция. Введение в дисциплину. Обзор платформ для мобильных систем. Системное и прикладное программное обеспечение мобильных систем. Роль операционной системы. Характеристика операционных систем Android, iOS.	2	
Лабораторная работа. ПО для мобильной разработки. Установка и настройка Cordova Cli, Android Studio.	2	
Лекция. Интерфейсы ввода/вывода данных мобильных систем и принципы их работы. Основы работы с сенсорным вводом. Акселерометр, датчики определения местоположения. API устройств ввода/вывода. Доступ к Web-сервисам.	2	
Лабораторная работа. Разработка простых кроссплатформенных приложений HTML/CSS/JS в среде Cordova Cli.	6	
Задания для самостоятельной работы, в том числе выполнение Проработка лекций Подготовка к практическим работам Встроенное ПО мобильных устройств под управлением ОС Android, iOS.	16	
Технологии разработки программного обеспечения для мобильных устройств	38	ПК-1
Лекция. Языки программирования для разработки приложений для мобильных систем: Java, Kotlin, Swift, C++, C#. Виртуальная машина Java. Среда MS Visual Studio и кроссплатформенная среда разработки Xamarin.	2	
Лекция. Нативные, Web- и гибридные приложения. Способы разработки мобильных приложений. Технология разработки NoCode. Распространение приложений. Типы мобильных приложений. Событийно-ориентированное программирование.	4	
Лабораторная работа. Разработка кроссплатформенных приложений HTML/CSS/JS в среде Cordova Cli для работы с датчиками.	4	
Лабораторная работа. Локальные хранилища данных для мобильных приложений в кроссплатформенных приложениях HTML/CSS/JS	4	
Лабораторная работа. NoCode-разработка в среде Adalo/Glide	4	

Задания для самостоятельной работы, в том числе выполнение Проработка лекций Подготовка к практическим работам Языки Web-программирования HTML5/CSS3/JS	20	ПК-1
Разработка нативных приложений	42	
Лекция. Архитектура платформы Android. Состав компонентов и их API ОС Android.	1	
Лабораторная работа. Разработка простых нативных приложений в среде Android Studio.	2	
Лекция. Особенности исполнения приложений Android. Многозадачность Android - процессы и потоки. Безопасность в Android - принцип "песочницы".	2	
Лекция. Архитектура приложения Android, основные компоненты. Компонент Activity. Жизненный цикл Activity.	2	
Лабораторная работа. Жизненный цикл Activity. Разработка приложения с двумя Activity.	4	
Лекция. Компоненты Service и Broadcast Receivers, Виды фоновых сервисов и принципы их работы. Компонент Content provider.	1	
Лабораторная работа. Разработка приложений с фоновыми сервисами.	3	
Лабораторная работа. Хранилища данных для мобильных приложений.	3	
Задания для самостоятельной работы, в том числе выполнение Проработка лекций Подготовка к практическим работам Алгоритмический язык Java	24	
Иная контактная работа:	0	

Раздел 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Изучение дисциплины "Программирование мобильных устройств" рекомендуется начать с ознакомления с рабочей программой, ее структурой и содержанием разделов. Учебный материал структурирован, изучение дисциплины осуществляется в тематической последовательности.

Занятия лекционного типа дают систематизированные знания по дисциплине "Программирование мобильных устройств", концентрируют внимание на наиболее сложных и важных вопросах. Во время лекционных занятий рекомендуется вести конспектирование учебного материала; обращать внимание на формулировки и категории, раскрывающие суть проблемы, явления или процесса; зафиксировать выводы и практические рекомендации. Подготовка к **занятиям лабораторного типа** включает ознакомление с планом лабораторного занятия; работу с конспектом лекций, выполнение домашнего задания, работу с учебной и учебно-методической литературой, научными изданиями и электронными образовательными ресурсами, рекомендованными рабочей программой дисциплины "Программирование мобильных устройств".

Содержание **самостоятельной работы** определяется рабочей программой дисциплины "Программирование мобильных устройств", оценочными и методическими материалами, заданиями и указаниями преподавателя. Самостоятельная работа может осуществляться в аудиторной и внеаудиторной формах. Эффективным средством осуществления самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к образовательной программе, рабочей программе дисциплины "Программирование мобильных устройств", к электронным библиотечным системам,

профессиональным базам данных и информационным справочным системам.

Периодичность проведения, формы текущего контроля успеваемости, система оценивания хода освоения дисциплин представлены в рабочей программе. Формой промежуточной аттестации по дисциплине "Программирование мобильных устройств" является БРК.

Раздел 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Учебно-методическое обеспечение

№№ п/п	Список используемой литературы	Количество экземпляров печатных изданий, имеющихся в библиотеке, или электронный адрес издания (ресурса) в сети Интернет
УЧЕБНЫЕ, УЧЕБНО-МЕТОДИЧЕСКИЕ И НАУЧНЫЕ ИЗДАНИЯ		
1.	Нехорошкова, Л. Г. Управление проектами разработки программного обеспечения [Текст] : практикум / Л. Г. Нехорошкова. Йошкар-Ола: МарГТУ, 2009. - 151 с. Экземпляры: всего 86.	86 / https://portal.volgatech.net/books/nexoroshkova_upravlenie_proektami.pdf
2.	Горнаков, Станислав Геннадьевич. Программирование мобильных телефонов на Java 2 Micro Edition [Текст] / С. Г. Горнаков. Москва: ДМК Пресс, 2005. - 335 с. ISBN 5-94074-279-3. Экземпляры: всего 14.	14
3.	Евсеев, Дмитрий Андреевич. Web-дизайн в примерах и задачах [Текст] : [учебное пособие для студентов вузов по специальности 080801 "Прикладная информатика" и другим экономическим специальностям] / Д. А. Евсеев, В. В. Трофимов ; под ред. В. В. Трофимова; С.-Петерб. гос. ун-т экономики и финансов. Москва: КноРус, 2010. - 263 с. ISBN 978-5-406-00017-5. Экземпляры: всего 10.	10
4.	Заяц, А. М. Проектирование и разработка WEB-приложений. Введение в frontend и backend разработку на JavaScript и node.js [Электронный ресурс] / Заяц А. М., Васильев Н. П. 3-е изд., стер. Санкт-Петербург: Лань, 2021. - 120 с. ISBN 978-5-8114-7042-6.	https://e.lanbook.com/book/154380
5.	Васильев, Н. П. Введение в гибридные технологии разработки мобильных приложений [Электронный ресурс] / Васильев Н. П., Заяц А. М. 3-е изд., стер. Санкт-Петербург: Лань, 2022. - 160 с. ISBN 978-5-507-44502-8.	https://e.lanbook.com/book/230387

6.2. Материально-техническая база и программное обеспечение

№№ п/п	Аудитории для проведения учебных занятий, самостоятельной работы и проведения государственной итоговой аттестации	Перечень основного оборудования	Программное обеспечение
1.	518 (III)	ПК 5 - ICL RAY P222.3 ,клавиат.,мышь.,монитор LG E2251T-BN (14), Сист. блок CE 331/256*2/PC 3200/80 Gb/FDD/DVD	Microsoft Windows Enterprise, Microsoft Office Standard, Агент Dr.Web, Microsoft

		-ROM/КЛАВ+МЫШЬ+коврик (1), Комплект учебной мебели (1)	Access, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач
--	--	---	---

Раздел 7. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ/ ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Критерии оценивания индикаторов достижения компетенций направлены на:

- усвоение теоретического материала (объем знаний, глубина усвоения), предусмотренного рабочей программой;
- умение излагать материал (четкость, грамотность изложения материала, точность и полнота воспроизведения учебного материала);
- умение применять теоретические знания при решении практических заданий.

Шкала оценивания представлена ниже.

Уровень сформированности элементов компетенции	Критерии оценивания	Шкала оценивания
Пороговый уровень	Обучающийся имеет знания основного материала, проявляет умение логично его излагать, но может допускать неточности в изложении материала, недостаточно правильные формулировки, испытывает затруднения в выполнении практических заданий.	удовлетворительно
Продвинутый уровень	Обучающийся твердо знает программный материал, излагает его грамотно и по существу, не допускает существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения	хорошо
Высокий уровень	Обучающийся глубоко и прочно усвоил программный материал, грамотно и логически стройно его излагает, дает исчерпывающие ответы на поставленные вопросы. В ответе тесно увязывается теория с практикой, при этом обучающийся не затрудняется с ответом при видоизменении задания, свободно справляется с задачами, вопросами и другими видами применения знаний, показывает знакомство с монографической литературой, периодическими изданиями, правильно обосновывает принятые решения, свободно владеет разносторонними навыками, приемами выполнения практических работ	отлично

7.1. Текущий контроль успеваемости

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины (модуля) и производится с применением технологии рейтингового контроля в соответствии с технологической картой дисциплины. Порядок составления технологической карты и алгоритм проведения процедуры оценивания видов деятельности обучающихся, направленных на освоение знаний, умений, навыков и/или опыта деятельности, по накопительной системе в баллах устанавливается положением о системе РИТМ в ФГБОУ ВО «ПГТУ»

7.2. Промежуточная аттестация обучающихся

Промежуточная аттестация обучающихся направлена на оценивание результатов обучения по дисциплине (модулю) и проводится с использованием фондов оценочных средств.

Примеры типовых контрольных заданий из базы фонда оценочных средств по образовательной программе.

Примеры типовых контрольных заданий

Какое хранилище данных нельзя использовать в Android?

1. **WebSQL**
2. SQLite
3. LocalStorage
4. IndexedDB

Что такое JSON?

1. **Текстовый формат для обмена данными между приложениями**
2. Класс, используемый для преобразования форматов данных.
3. Бинарный формат долговременного хранения больших объемов данных.
4. Класс, предоставляющий методы передачи и приема данных по сети.

Что такое поток?

1. **Это одна из параллельно исполняемых ветвей программного кода**
2. Это последовательность команд параллельно исполняемых приложений.
3. Это последовательность команд одного из параллельно исполняемых приложений.
4. Это последовательность команд всех параллельно исполняемых ветвей программного кода.

Что такое ContentProvider и для чего его используют?

1. **Это компонент приложения, который управляет доступом к структурированному набору данных.**
2. Это компонент приложения, который управляет доступом к данным Activity.
3. Это средство доступа в сеть Интернет.
4. Это средство поиска данных в сети Интернет.

Укажите функцию объектов класса Sensor на платформе Android

1. **Предоставляет методы для определения возможностей, доступных конкретному датчику мобильного устройства.**
2. Предоставляет необработанные данные конкретного датчика.
3. Предоставляет данные о точности измерений датчика.
4. Предоставляет методы калибровки и регистрации датчиков.

Для чего не используют объект Intent в приложениях на платформе Android?

1. **Для доступа к данным приложения.**
2. Для запуска нового экземпляра Activity.
3. Для запуска службы.
4. Для передачи сообщения другим приложениям

Какие базовые Android-компоненты мобильного приложения можете назвать?

1. **Activity, ContentProvider, Service, BroadcastReceiver**
2. Intent, Activity, Service, Sensor
3. SensorManager, BroadcastReceiver, Thread
4. Thread, SensorEvent, Layout, Service

Приложения какого типа создаются с помощью Cordova Cli?

1. **Кроссплатформенные приложения.**
2. «Нативные» приложения
3. Клиент-серверные приложения
4. Многопоточные приложения

Какие языки программирования используются в разработке кроссплатформенных приложений?

1. **HTML, CSS, JavaScript, Dart**
2. Java, Dart, XML, Kotlin
3. C++, Swift, Java, HTML
4. JSON, HTML, Kotlin, ObjectC

Android SDK – это ...

1. **универсальный набор библиотек и средств разработки мобильных приложений для операционной системы Android.**
2. набор библиотек и инструментов разработки графического интерфейса мобильного приложения
3. набор библиотек и инструментов разработки клиент-серверных приложений
4. набор библиотек и инструментов, обеспечивающих эмуляцию мобильного устройства на настольном компьютере.

Укажите основные принципы событийно-ориентированного программирования.

Событийно-ориентированное программирование - парадигма программирования, в которой выполнение программы определяется событиями.

Событие— это сигнал, возникающий в программной среде (среде выполнения кода) и на который можно отреагировать (обработать событие). Программист может явно задать конкретное поведение программы при наступлении конкретного события. Источники событий – это действия пользователя (нажатие кнопки), поступление данных или сообщения операционной системы (разряд батареи,

дефицит памяти)

Обработчик события – это блок кода, написанный программистом для определения реакции программы на конкретное событие. Обычно это функция или метод класса. Обработчик вызывается при генерации события.

Подписка на событие – это связь, указывающая, что конкретное событие является значимым для конкретного обработчика.

Момент времени и порядок исполнения обработчиков в программе определяется событиями и в общем случае случайны.

Назначение объекта Intent (Намерение) в Android-приложении.

Intent - это механизм для описания одной операции, а также способ обмена сообщениями между компонентами приложения.

Приложение Android – это набор компонентов. Все компоненты (приложения и экраны) изолированы. Единственный способ, которым они общаются друг с другом, — это намерения.

В частности, интенды можно использовать для запуска различных компонентов: активностей, служб и широковеб-приемников. Например, чтобы запустить активность, можно использовать метод.

```
Context.startActivity(intent)
```

Операционная система запустит компонент, который указан в объекте **intent**.

Android поддерживает два типа намерений: явные и неявные. Когда приложение определяет свой целевой компонент в намерении, это явное намерение. Если приложение не называет целевой компонент, это является неявным намерением.

Например, для запуска нового **Activity** следует использовать явное намерение. Необходимо создать объект типа **Intent** и указать в его параметрах полное имя класса **Activity**, другие параметры (если требуется) и передать объект методу **Context.startActivity()**. Далее операционная система запустит **Activity**, указанную в объекте **Intent**.

Неявные намерения позволяют операционной системе найти среди всех установленных приложений подходящие компоненты приложений, а пользователю – выбрать конкретный компонент.

Разработка мобильных приложений в Cordova Cli

Apache Cordova Cli - это среда разработки для создания кроссплатформенных мобильных приложений с помощью Web-технологий на языках HTML, CSS, JavaScript и его фреймворков.

После установки фреймворка Cordova Cli разработка приложений ведется в следующем порядке. Первоначально создается типовый проект приложения – дерево каталогов заданной структуры. Одним из каталогов этого дерева является WWW. В этом каталоге размещаются файлы Web-приложения, в остальных каталогах - служебные утилиты и модули.

Программист ведет разработку в обычном стиле SPA приложения. *SPA (англ. single page application)* – “одностраничное приложение”. Технически SPA - это Web-сайт. Интерфейс определяется с помощью HTML- и CSS-кода. Функционал приложения программируется средствами JavaScript-кода.

Для отладки программного кода Cordova предоставляет платформу браузера. Это позволяет выполнять пошаговую отладку кода приложения в режиме разработчика штатного Web-браузера

Cordova имеет мощный инструмент, позволяющий расширять функционал нативным кодом – это делается с помощью установки плагинов. Каждый отдельный плагин обеспечивает JavaScript

интерфейс к специфическим функциям, например, доступ к измерительным датчикам или к СУБД.

Для установки разработанного приложения на реальном устройстве Cordova Cli позволяет добавить в сборку приложения платформы Android или iOS.

Собранное приложение Cordova способна загрузить на реальное мобильное устройство Android или iOS обычно без корректировки HTML/CSS/JS-кода. В этом состоит кроссплатформенность приложений, созданных в Cordova.

В мобильном устройстве разработанное приложение выполняется специальным компонентом операционной системы Webview.

Перечень вопросов для проведения промежуточной аттестации

5. Какие устройства называются мобильными?
6. Укажите типы мобильных устройств общего применения и специализированных мобильных устройств.
7. Какими датчиками оснащены современные мобильные устройства?
8. Перечислите наиболее популярные операционные системы для мобильных устройств.
9. Каковы особенности операционных систем для мобильных устройств?
10. Какие сервисы предоставляют операционные системы современных мобильных устройств?
11. Какие коммуникационные технологии поддерживают ОС мобильных устройств?
12. Какие хранилища данных используются в ОС Android?
13. Где используются форматы данных JSON, XML?
14. Назовите классы основных компонентов приложения Android.
15. Укажите основные принципы событийно-ориентированного программирования.
16. Ядро какой ОС использует Google Android?
17. Каковы основные возможности ОС Google Android?
18. Какой язык программирования применяется для разработки приложений для ОС Google Android?
19. Какую реализацию машины Java использует Google Android?
20. Какие инструменты разработки применяются для приложений для Google Android?
21. Ядро какой ОС использует iOS?
22. Укажите особенности платформы iOS.
23. Объясните жизненный цикл приложения iOS.
24. Какие основные встроенные и сервисные приложения предоставляет iOS?
25. Какие инструменты разработки применяются для приложений для iOS?
26. Каковы перспективы и направления развития ОС для мобильных устройств?
27. В чем заключаются особенности мобильного программирования?

28. Перечислите типы мобильных приложений.
29. Какие приложения называются «нативными» (native)?
30. Какие языки программирования используются для разработки «нативных» приложений в различных мобильных ОС?
31. Объясните особенности, механизм запуска и работы кроссплатформенных Web-приложений.
32. Какие языки программирования используются для разработки Web-приложений?
33. Какие приложения называются гибридными?
34. В чем заключаются особенности гибридных приложений для мобильных устройств?
35. Какие средства программирования используются для разработки приложений на языке Java?
36. Какие языки программирования являются основными для разработки приложений для ОС iOS?
37. Какие языки программирования являются основными для разработки приложений для ОС Windows Phone?

Какие приложения называются кроссплатформенными?

38. Какие технологии и языки программирования применяются для разработки кроссплатформенных приложений?
39. Какова последовательность разработки, тестирования и распространения приложений для мобильных устройств?
40. Дайте характеристику технологии разработки мобильных приложений HTML5/JS.
41. Почему приложение, созданное по технологии HTML5/JS, является кроссплатформенным?
42. Как запускается приложение, созданное по технологии HTML5/JS?
43. Какие средства обеспечивают графический интерфейс пользователя в приложениях HTML5/JS?
44. Как обеспечивается реакция приложения HTML5/JS на воздействия пользователя и сигналы датчиков?
45. Какие средства программирования используются для разработки приложений по технологии HTML5/JS?
46. Какова структура проекта приложения HTML5/JS в среде Cordova Cli?
47. Как выполняется отладка и тестирование приложения HTML5/JS?
48. Как установить созданное приложение HTML5/JS на мобильное устройство?
49. Какой язык программирования используется для разработки «нативных» приложений для ОС Android?
50. Объясните особенности исполнения программ, написанных на языке Java.
51. Объясните смысл терминов «Байт-код» и «Виртуальная машина».
52. Каким образом достигается кроссплатформенность мобильных приложений?
53. Объясните методы отладки мобильных приложений, разработанных на настольном

компьютере.

54. Что такое «эмулятор мобильного устройства» и как он используется при разработке мобильного приложения?
55. Укажите методы монетизации мобильных приложений.
56. Как влияют методы монетизации на процесс работы и привлекательность приложений для пользователей?
57. Какие уровни входят в состав архитектуры ОС Android?
58. Каковы функции ядра ОС Android?
59. Каковы функции уровня библиотек и среды исполнения в архитектуре ОС Android?
60. Какие библиотеки включены в состав ОС Android?
61. Что входит в состав среды исполнения ОС Android?
62. Какие компоненты входят в состав каркаса приложений (Application Framework) ОС Android?
63. Объясните функции представлений (Views) ОС Android.
64. Объясните функции контент-провайдеров (Content Providers) ОС Android.
65. Объясните функции менеджера ресурсов (Resource Manager) ОС Android.
66. Объясните функции менеджера оповещений (Notification Manager) ОС Android.
67. Объясните функции менеджера местоположения (Location Manager) ОС Android.
68. Укажите основные виды Android-приложений.
69. Что такое «приложения переднего плана»?
70. Какие приложения называются фоновыми?
71. Что такое смешанные приложения?
72. Определите особенности приложений-виджетов (App Widgets).
73. Какие виды многозадачности реализуются в мобильных ОС?
74. Как реализуется многозадачность на основе потоков?
75. Как реализуется многозадачность в ОС iOS?
76. Как реализуется многозадачность на основе процессов?
77. Как реализуется безопасное исполнение приложений в ОС Android?
78. Что называется «песочницей» в операционной системе?
79. Какие средства используются в ОС Android для контроля привилегий?
80. Могут ли разные приложения ОС Android использовать компоненты других приложений?
81. Имеют ли приложения ОС Android единую точку входа (старта приложения)?
82. На основе каких типовых компонентов строятся приложения ОС Android?
83. Каковы функции компонентов Активности (Activities)?

84. Каковы функции компонентов Сервисы (Services)?
85. Каковы функции компонентов Контент-провайдеры (Content providers)?
86. Каковы функции компонентов Приемники широковещательных сообщений (Broadcast Receivers)?
87. Каковы функции компонентов Активности (Activities)?
88. Какие задачи решают экземпляры класса Намерение (Intent)?
89. Объясните жизненный цикл Активности (Activities).
90. Какие события предусмотрены для Активности (Activities)?
91. Укажите два способа исполнения сервисов. В чем их отличие?
92. Объясните жизненный цикл Сервисов (Services).
93. Какие события предусмотрены для Сервисов (Services)?
94. Какие события предусмотрены для Контент-провайдеров (Content providers)?
95. Какие виды сообщений обрабатывают приемники широковещательных сообщений (Broadcast Receivers)?
96. Что такое «манифест приложения» и каковы его функции?
97. Как организуются ресурсы приложения?

An error has occurred while processing HtmlTextBox 'htmlTextBox3': List item () must be in a list (or).