

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Галунин Сергей Александрович
Должность: проректор по учебной работе
Дата подписания: 23.12.2025 13:42:26
Уникальный программный ключ:
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП
«Разработка программно-
информационных систем»



СПбГЭТУ «ЛЭТИ»
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение высшего образования
**«Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ» им. В.И.Ульянова (Ленина)»
(СПбГЭТУ «ЛЭТИ»)**

РАБОЧАЯ ПРОГРАММА
дисциплины
«МОБИЛЬНАЯ РАЗРАБОТКА НА ANDROID Ч.1»
для подготовки бакалавров
по направлению
09.03.04 «Программная инженерия»
по профилю
«Разработка программно-информационных систем»

Санкт-Петербург

2025

ЛИСТ СОГЛАСОВАНИЯ

Разработчики:

доцент, к.т.н., доцент Заславский М.М.

Рабочая программа рассмотрена и одобрена на заседании кафедры МОЭВМ
20.01.2025, протокол № 1

Рабочая программа рассмотрена и одобрена учебно-методической комиссией
ФКТИ, 28.01.2025, протокол № 1

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

1 СТРУКТУРА ДИСЦИПЛИНЫ

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	МОЭВМ
Общая трудоемкость (ЗЕТ)	3
Курс	3
Семестр	5

Виды занятий

Электронные лекции (акад. часов)	34
Электронные практические (академ. часов) (академ. часов)	34
Иная контактная работа (академ. часов)	1
Все контактные часы (академ. часов)	1
Самостоятельная работа, включая часы на контроль (академ. часов)	39
Всего (академ. часов)	108

Вид промежуточной аттестации

Дифф. зачет (курс)	3
--------------------	---

2 АННОТАЦИЯ ДИСЦИПЛИНЫ

«МОБИЛЬНАЯ РАЗРАБОТКА НА ANDROID Ч.1»

Дисциплина является первой частью курса, который знакомит с принципами разработки для мобильных устройств. Курс позволяет освоить создание мобильных приложений для платформы Android, охватывает основные концепции и инструменты, необходимые для разработки современных и функциональных Android-приложений. В процессе освоения курса предлагается изучить основы языка программирования Kotlin, среду разработки Android Studio, получить навыки использования ее для создания и отладки приложений, получить практический опыт разработки собственных приложений

SUBJECT SUMMARY

«MOBILE DEVELOPMENT ON ANDROID PART 1»

The discipline is the first part of the course, which introduces the principles of development for mobile devices. The course allows you to master the creation of mobile applications for the Android platform, covers the basic concepts and tools necessary for the development of modern and functional Android applications. In the process of mastering the course, it is proposed to study the basics of the Kotlin programming language, the Android Studio development environment, gain skills in using it to create and debug applications, and gain practical experience in developing your own applications.

3 ОБЩИЕ ПОЛОЖЕНИЯ

3.1 Цели и задачи дисциплины

1. Целью дисциплины является получение теоретических знаний в области разработки приложений для мобильных платформ, а также практических умений и навыков по применению полученных знаний для решения задач профессиональной деятельности.
2. Задачами дисциплины являются приобретение знаний, умений и навыков в области разработки приложений для мобильных платформ.
3. Обучающиеся получают знания о принципах проектирования, разработки, отладки мобильных приложений и их пользовательских интерфейсов.
4. В результате освоения дисциплины приобретаются умения проектировать и разрабатывать мобильные приложения.
5. Обучающиеся приобретают навыки работы с инструментами и технологиями разработки инструментальных средств оценки качества мобильных приложений.

3.2 Место дисциплины в структуре ОПОП

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Информатика»
2. «Программирование»
3. «Информационные технологии»

и обеспечивает изучение последующих дисциплин:

1. «Мобильная разработка на Android ч.2»
2. «Производственная практика (преддипломная практика)»

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

В результате освоения образовательной программы обучающийся должен достичь следующие результаты обучения по дисциплине:

Код компетенции/ индикатора компетенции	Наименование компетенции/индикатора компетенции
ПК-0	Способен разрабатывать информационные модели и применять их для решения задач профессиональной деятельности
ПК-0.1	<i>Знает современные виды информационных моделей, применяемых при решении задач профессиональной деятельности</i>
ПК-0.2	<i>Создает и модифицирует информационные модели для решения задач профессиональной деятельности</i>
ПК-0.3	<i>Применяет информационные модели для решения задач профессиональной деятельности</i>

4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1 Содержание разделов дисциплины

4.1.1 Наименование тем и часы на все виды нагрузки

№ п/п	Наименование темы дисциплины	ЭЛек, ач	ЭПр, ач	ИКР, ач	СР, ач
1	Введение	1			1
2	Язык Kotlin	6	4		6
3	Проектная работа на языке Kotlin	6	6		6
4	Введение в UI и хранение данных	6	6		6
5	Продвинутый UI -списки и библиотеки	7	6		6
6	Сеть и хранение данных	7	6		6
7	Заключение	1	6	1	8
	Итого, ач	34	34	1	39
	Из них ач на контроль	0	0	0	0
	Общая трудоемкость освоения, ач/зе				108/3

4.1.2 Содержание

№ п/п	Наименование темы дисциплины	Содержание
1	Введение	Отличие разработки приложений для мобильных устройств от разработки для настольных ПК. Влияние мобильности устройств на сценарии использования приложений. Рабочая программа, коммуникации между заинтересованными лицами
2	Язык Kotlin	От Java к Kotlin. Для чего создавали Kotlin и чем интересен его код. Типы данных и переменные, приведение типов, операции с простыми типами. Операторы в Kotlin, условные операторы в Java и Kotlin (if и when). Циклы for и while в Kotlin. Чем они отличаются от циклов Java. Классы и методы в Kotlin, их свойства, блок init, дефолтные и именованные параметры. Как использовать nullable-типы, паттерн Singleton и объявление объекта, анонимные и enum-классы.
3	Проектная работа на языке Kotlin	Наследование и система типов в Kotlin. Изолированные классы и интерфейсы. Массивы и коллекции в Kotlin. Функции kotlin.collections. Функции высшего порядка, расширения и мультидекларации. Android Debug Bridge (ADB), Android Debugger. Логирование. Командная строка. Устройство хранилища Git, создание коммита в Git, манипулирование коммитами и роль .gitignore-файла, удалённый репозиторий Git.

№ п/п	Наименование темы дисциплины	Содержание
4	Введение в UI и хранение данных	Инструмент вёрстки Figma и язык разметки XML. Структура и синтаксис ресурсных файлов. Изображения в приложениях, величина DPI: от чего она зависит и как влияет на качество картинок на экране. Типы масштабирования. Навигация приложения, компонент Activity для взаимодействия пользователя с приложением, жизненный цикл Activity lifecycle, роль и типы класса Intent. Что такое стили и для чего нужны. Синтаксис стилей, эффективное использование стилей, сторонние шрифты. Чем тема отличается от стиля, как она устроена и как её настроить. Место Context в Android-разработке, типы Context при создании приложения. API Context, Intent Filters, передача данных с помощью Intent. EditText в тестовом приложении Prime Number, методы и атрибуты EditText, инструменты Android SDK для работы с вводимым текстом, TextWatcher. Данные в жизни приложения и их хранение на практике.
5	Продвинутый UI -списки и библиотеки	Что такое класс RecyclerView, как его создать, без каких вспомогательных классов не работает RecyclerView. LayoutManager и его базовые типы. ViewHolder. Что такое класс ViewHolder, почему он появился, когда и зачем использовать его основные методы. Adapter как целое и часть RecyclerView. Собираем RecyclerView. Особенности работы с Adapter. Что такое библиотеки и для чего они нужны, подключение библиотек к проекту. Библиотека Glide, загрузка изображений из интернета в ImageView.
6	Сеть и хранение данных	Выход в сеть. Кто разрабатывает сервер и как он работает. REST API. HTTP. Путь данных от клиента на сервер, их трансформация. Типы запросов, формат данных и сетевые ошибки. TCP/IP, HTTP, URI, JSON. Популярные аннотации в библиотеках Android и их возможности. Библиотеки Gson и Retrofit. API и ошибки. Авторизация запросов, практика с API крупного сервиса IMDb.
7	Заключение	Общие выводы по изложенному материалу. Содержание второй части курса. Связь с последующими дисциплинами траектории.

4.2 Перечень лабораторных работ

Лабораторные работы не предусмотрены.

4.3 Перечень практических занятий

Наименование практических занятий	Количество ауд. часов
1. UI и хранение данных	11
2. Списки и библиотеки	11
3. Сеть и хранение данных	12
Итого	34

4.4 Курсовое проектирование

Курсовая работа (проект) не предусмотрены.

4.5 Реферат

Реферат не предусмотрен.

4.6 Индивидуальное домашнее задание

Индивидуальное домашнее задание не предусмотрено.

4.7 Доклад

Доклад не предусмотрен.

4.8 Кейс

Кейс не предусмотрен.

4.9 Организация и учебно-методическое обеспечение самостоятельной работы

Самостоятельной записи на курс нет. Студент заходит на курс, используя логин/пароль от единой учетной записи университета (единий логин и пароль). Каждую неделю будет доступна новая тема курса: кратко раскрывающие содержание каждой темы, презентации и конспекты, с которыми обучающиеся смогут ознакомиться в любое удобное время. Все темы включают практические занятия, которые предусматривают самостоятельное выполнение заданий,

а также задания с автоматической проверкой, результаты которых учитываются при общей аттестации полученных знаний. В конце каждой лекции необходимо пройти небольшой контрольный тест, который покажет насколько усвоен предложенный материал. Рекомендуем изучать материал последовательно, что существенно облегчит работу. У каждого контрольного задания имеется своя форма (тест или практическое задание) есть срок выполнения (окончательный срок), по истечении которого даже правильные ответы система принимать не будет! В расписании курса указан окончательный срок каждого задания, который варьируется от двух до четырех недель в зависимости от его сложности. Весь учебный курс рассчитан на 16 недель. Его итоги будут подведены в течение нескольких недель после его окончания.

Текущая СРС	Примерная трудоемкость, ач
Работа с лекционным материалом, с учебной литературой	15
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	6
Самостоятельное изучение разделов дисциплины	6
Выполнение домашних заданий, домашних контрольных работ	0
Подготовка к лабораторным работам, к практическим и семинарским занятиям	6
Подготовка к контрольным работам, коллоквиумам	0
Выполнение расчетно-графических работ	0
Выполнение курсового проекта или курсовой работы	0
Поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
Работа над междисциплинарным проектом	0
Анализ данных по заданной теме, выполнение расчетов, составление схем и моделей, на основе собранных данных	0
Подготовка к зачету, дифференциированному зачету, экзамену	6
ИТОГО СРС	39

5 Учебно-методическое обеспечение дисциплины

5.1 Перечень основной и дополнительной литературы, необходимой для освоения дисциплины

№ п/п	Название, библиографическое описание	К-во экз. в библ.
Основная литература		
1	Шалин Павел Программирование на Kotlin для Android : Пер. с англ. / Павел Шалин, проекта руковод., 2024. -336 с. -Текст : непосредственный.	неогр.
Дополнительная литература		
1	Нейл Т. Мобильная разработка : Галерея шаблонов / Т. Нейл, 2013. -208 с. -Текст : непосредственный.	неогр.

5.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых при освоении дисциплины

№ п/п	Электронный адрес
1	Руководство по языку Kotlin https://metanit.com/kotlin/tutorial/

5.3 Адрес сайта курса

Адрес сайта курса: <https://vec.etu.ru/moodle/course/view.php?id=23616>

6 Критерии оценивания и оценочные материалы

6.1 Критерии оценивания

Для дисциплины «Мобильная разработка на Android ч.1» формой промежуточной аттестации является дифф. зачет. Оценивание качества освоения дисциплины производится с использованием рейтинговой системы.

Дифференцированный зачет

Оценка	Количество баллов	Описание
Неудовлетворительно	0 – 7	теоретическое содержание курса не освоено, необходимые практически навыки и умения не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над курсом не приведет к существенному повышению качества выполнения учебных заданий
Удовлетворительно	8 – 11	теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки и умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки
Хорошо	12 – 13	теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки и умения сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками
Отлично	14 – 15	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки и умения сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено количеством баллов, близким к максимальному

Особенности допуска

К дифференцированному зачету допускаются студенты, показавшие прогресс в ЭОС Moodle (не менее 80%) и выполнившие 3 практические работы на оценку не менее "удовлетворительно".

Дифференцированный зачет проводится в виде тестирования в электронной образовательной среде Moodle. Тест содержит 15 вопросов с развернутыми ответами. Каждый вопрос теста оценивается в 1 балл в случае правильного ответа. Оценка за дифф. зачет выставляется в зависимости от количества баллов, полученных за итоговый тест следующим образом:

"Неудовлетворительно" -менее 8 баллов

"Удовлетворительно" -от 8 до 11 баллов

"Хорошо" -от 12 до 13 баллов

"Отлично" -от 14 до 15 баллов.

6.2 Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине

Вопросы к дифф.зачету

№ п/п	Описание
1	Отличие разработки приложений для мобильных устройств от разработки для настольных ПК.
2	Для чего создавали Kotlin и чем интересен его код. Типы данных и переменные, приведение типов, операции с простыми типами.
3	Операторы в Kotlin, условные операторы в Java и Kotlin (if и when).
4	Циклы for и while в Kotlin. Чем они отличаются от циклов Java.
5	Классы и методы в Kotlin, их свойства, блок init, дефолтные и именованные параметры.
6	Как использовать nullable типы, паттерн Singleton и объявление объекта, анонимные и enum классы
7	Массивы и коллекции в Kotlin.
8	Функции kotlin.collections.
9	Функции высшего порядка, расширения и мультидекларации.
10	Изображения в приложениях, величина DPI: от чего она зависит и как влияет на качество картинок на экране.
11	Типы масштабирования.

12	Навигация приложения, компонент Activity для взаимодействия пользователя с приложением, жизненный цикл Activity lifecycle, роль и типы класса Intent.
13	Что такое класс ViewHolder, почему он появился, когда и зачем использовать его основные методы.
14	Adapter как целое и часть RecyclerView.
15	Особенности работы с Adapter.
16	Библиотека Glide, загрузка изображений из интернета в ImageView.
17	REST API. HTTP
18	Путь данных от клиента на сервер, их трансформация.
19	Типы запросов, формат данных и сетевые ошибки. TCP/IP, HTTP, URI, JSON.
20	API и ошибки

Вариант теста

1. Выберите верное утверждение
 - (a) **Array имеет постоянный размер**
 - (b) List это реализация неизменяемого массива
 - (c) Все списки в kotlin - связанные.
2. Отметьте типы данных Kotlin, значения которых НЕ **могут быть null**.
 - (a) String?
 - (b) **String**
 - (c) Int
 - (d) Int?
3. Что означает ключевое слово val при объявлении переменной
 - (a) Значение переменной нельзя менять
 - (b) **Ссылку переменной нельзя менять**
 - (c) Значение переменной можно менять
 - (d) Ссылку переменной можно менять
4. Что означает ключевое слово var при объявлении переменной
 - (a) Значение переменной нельзя менять
 - (b) Ссылку переменной нельзя менять
 - (c) Значение переменной можно менять
 - (d) **Ссылку переменной можно менять**

5. Укажите свойство, которое определяет внешний отступ элемента.
- (a) **margin**
 - (b) height
 - (c) padding
 - (d) width
 - (e) weight
 - (f) layout_width
 - (g) layout_height
6. Какой язык программирования является основным для разработки Android-приложений?
- (a) C#
 - (b) **Java**
 - (c) Python
 - (d) Kotlin
7. Какое расширение имеет файл, который используется для определения пользовательского интерфейса в Android?
- (a) **.xml**
 - (b) .json
 - (c) .html
 - (d) .java
8. Что такое Android Manifest?
- (a) Содержит описание активности
 - (b) Описывает настройки приложения
 - (c) Определяет зависимости приложения
 - (d) **Все вышеперечисленное**
9. Какой метод вызывается, когда Activity становится видимой пользователю?
- (a) onStart()
 - (b) **onResume()**

- (c) onCreate()
 - (d) onPause()
10. Что делает метод startActivity()?
- (a) Создает новое приложение
 - (b) **Запускает новую активность**
 - (c) Закрывает текущее приложение
 - (d) Обновляет текущую активность

Образцы задач (заданий) для контрольных (проверочных) работ

Пример практической работы

Разработка приложения для ведения списка покупок, которое должно удовлетворять требованиям:

- реализована возможность экспорта (Отправить) отдельных элементов данных через соцсети
- реализовано хранение пользовательских предпочтений (Порядок сортировки, фильтры)
- реализованы пользовательские закладки (отметки “Мне нравятся”) с возможностью просмотра отмеченных страниц на отдельном экране

Весь комплект контрольно-измерительных материалов для проверки сформированности компетенции (индикатора компетенции) размещен в закрытой части по адресу, указанному в п. 5.3

6.3 График текущего контроля успеваемости

Неделя	Темы занятий	Вид контроля
1	Язык Kotlin Проектная работа на языке Kotlin	
2		
3		
4		
5		Практическая работа
6	Введение в UI и хранение данных Продвинутый UI -списки и библиотеки	
7		
8		
9		
10		Практическая работа
11	Сеть и хранение данных	
12		
13		
14		
15		
16		Практическая работа

6.4 Методика текущего контроля

на лекционных занятиях

Текущий контроль включает в себя:

- контроль прогресса в ЭОС Moodle (не менее 80%).

на практических занятиях

- контроль прогресса в ЭОС Moodle (не менее 80%)

В процессе обучения по дисциплине студент обязан выполнить 3 практические работы в ЭОС Moodle.

Студент должен показать: понимание постановки задачи, подхода к ее решению, ход решения, выбор тех или иных методик решения задачи. Преподаватель определяет глубину понимания теоретического материала, который лежит в основе решения задачи практической работы, а также самостоятельность ее выполнения.

Критерии оценивания практических работ:

«отлично» - вопрос раскрыт полностью, задача решена правильно

«хорошо» - вопрос раскрыт не полностью, задача решена частично

«удовлетворительно» - в ответе на вопрос имеются существенные ошибки; задача не решена или решена неправильно, ход решения правильный

«неудовлетворительно» - отсутствует ответ на вопрос или содержание ответа не совпадает с поставленным вопросом, задача не решена, ход решения неправильный

самостоятельной работы студентов

Контроль самостоятельной работы студентов осуществляется на осуществляется на лекционных, практических занятиях по методикам, описанным выше.

7 Описание информационных технологий и материально-технической базы

Тип занятий	Тип помещения	Требования к помещению	Требования к программному обеспечению
Лекция	Лекционная аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, ПК	1) Linux Альт Образование 10 и выше; 2) Р7-Офис 7 и выше либо LibreOffice 7 и выше
Практические занятия	Помещение для самостоятельной работы	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, ПК	1) Linux Альт Образование 10 и выше; 2) Р7-Офис 7 и выше либо LibreOffice 7 и выше
Самостоятельная работа	Помещение для самостоятельной работы	Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.	1) Linux Альт Образование 10 и выше; 2) Р7-Офис 7 и выше либо LibreOffice 7 и выше

8 Адаптация рабочей программы для лиц с ОВЗ

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Дата	Изменение	Дата и номер протокола заседания УМК	Автор	Начальник ОМОЛА