Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 23.05.2024 10:24:59
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет информационных технологий

УТВЕРЖДАЮ
Декан факультета
«Информационные технологии»
_____ / Д.Г.Демидов /
«15» февраля 2024г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Основы разработки мобильных приложений»

Направление подготовки/специальность
09.03.01 Информатика и вычислительная техника

Профиль/специализация
«Веб-технологии»

Квалификация
Бакалавр

Формы обучения
Очная, заочная

Москва, 2024 г.

Разработчик(и):

ст.преподаватель / М.В.Даньшина /

Согласовано:
Заведующий кафедрой «Инфокогнитивные технологии»,
к.т.н., доцент / Е.А. Пухова /



Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 6
3 Структура и содержание дисциплины 6
3.1 Виды учебной работы и трудоемкость 6
3.2 Тематический план изучения дисциплины 7
3.3 Содержание дисциплины 8
4 Учебно-методическое и информационное обеспечение 13
4.1 Основная литература 13
4.2 Дополнительная литература 15
5 Материально-техническое обеспечение 15
5.1 Требования к оборудованию и помещению для занятий 15
5.2 Требования к программному обеспечению 14
6 Методические рекомендации 16
6.1 Методические рекомендации для преподавателя по организации обучения 16
6.2 Методические указания для обучающихся по освоению дисциплины 16
7 Фонд оценочных средств 17
7.1 Методы контроля и оценивания результатов обучения 17
7.2 Шкала и критерии оценивания результатов обучения 17
7.3 Оценочные средства 21
7.3.1 Экзаменационное задание. 21
7.3.2 Типовой экзаменационный билет. 22
7.3.3 Типовые практические задания. 22

1 Цели, задачи и планируемые результаты обучения по дисциплине

К **основным целям** освоения дисциплины «Основы разработки мобильных приложений» относится:

- изучение технологии разработки программного обеспечения для мобильных устройств с операционными системами на различных платформах, основ управления качеством и стандартизации разработки программных средств;
- формирование навыков использования современных технологий программирования;
- закрепление получаемых в семестре знаний и навыков на практике;
- формирование взаимосвязей, получаемых в семестре знаний и навыков с изученными ранее и изучаемых параллельно с данной дисциплиной;
- подготовка студентов к деятельности в соответствии с квалификационной характеристикой бакалавра.

К **основным задачам** дисциплины «Основы разработки мобильных приложений» относятся:

- программирование приложений, создание прототипа информационной системы, документирование проектов информационной системы на стадиях жизненного цикла, использование функциональных и технологических стандартов;
- изучение и освоение теоретического материала, как в процессе контактной, так и в ходе самостоятельной работы;
- выполнение предоставленных практических заданий различных форм, как в процессе контактной, так и в ходе самостоятельной работы;
- самостоятельная работа над тематикой дисциплины для формирования компетенций основной профессиональной образовательной программы (далее, ОПОП).

Обучение по дисциплине «Основы разработки мобильных приложений» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	ИУК-1.1. Анализирует задачу, выделяя ее базовые составляющие. ИУК-1.2. Осуществляет поиск, критически оценивает, обобщает, систематизирует и ранжирует информацию, требуемую для решения поставленной задачи. ИУК-1.3. Рассматривает и предлагает рациональные варианты решения поставленной задачи, используя

	системный подход, критически оценивает их достоинства и недостатки.
ПК-1. Способен разрабатывать требования и проектировать программное обеспечение	ИПК-1.1. Знает возможности существующей программно-технической архитектуры; возможности современных и перспективных средств разработки программных продуктов, технических средств; методологии разработки программного обеспечения и технологии программирования; методологии и технологии проектирования и использования баз данных; языки формализации функциональных спецификаций; методы и приемы формализации задач; методы и средства проектирования программного обеспечения; методы и средства проектирования программных интерфейсов; методы и средства проектирования баз данных; принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методы и средства проектирования программного обеспечения и баз данных; методы и средства проектирования программных интерфейсов. ИПК-1.2. Умеет проводить анализ исполнения требований; вырабатывать варианты реализации требований; проводить оценку и обоснование рекомендуемых решений; осуществлять коммуникации с заинтересованными сторонами; выбирать средства реализации требований к программному обеспечению; вырабатывать варианты реализации программного обеспечения; проводить оценку и обоснование рекомендуемых решений; осуществлять коммуникации с заинтересованными сторонами; использовать существующие типовые решения и шаблоны проектирования

	программного обеспечения; применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов; осуществлять коммуникации с заинтересованными сторонами. ИПК-1.3. Владеет современным инструментарием и средами разработки программного кода; современным инструментарием и средами проектирования программного кода, методами тестирования ПО.

В процессе освоения образовательной программы данные компетенции, в том числе их отдельные компоненты, формируются поэтапно в ходе освоения обучающимися дисциплин (модулей), практик в соответствии с учебным планом и календарным графиком учебного процесса.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к числу учебных дисциплин обязательной части, части формируемой участниками образовательных отношений блока Б1.2.

Дисциплина взаимосвязана логически и содержательно-методически со следующими дисциплинами и практиками ОПОП:

- Базы данных;
- Проектирование пользовательских интерфейсов в веб;
- Проектирование веб-сервисов;
- Разработка мобильных приложений;
- Математическая логика и теория алгоритмов в практике программирования.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 5 зачетных единицы, т.е. 180 академических часов (из них 108 часов - самостоятельная работа студентов и 72 часа – аудиторные занятия).

Разделы дисциплины изучаются на третьем курсе в пятом семестре, форма промежуточной аттестации – экзамен, курсовой проект.

3.1 Виды учебной работы и трудоемкость (по очной форме обучения)

№ п/п	Вид учебной работы	Количество часов	Семестры	
			5	
1	**Аудиторные занятия**	**72**	72	

	В том числе:			
1.1	Лекции	36	18	
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	36	36	
2	**Самостоятельная работа**	**108**	108	
	Курсовой проект			
3	**Промежуточная аттестация**			
	Экзамен		экзамен	
	Итого:	**180**	180	

3.2 Тематический план изучения дисциплины (по очной форме обучения)

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	**Аудиторная работа**				**Самостоятельная работа**
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Тема 1. Типы данных, операции и условные операторы языка программирования Kotlin	19	9				10
2	Тема 2. Циклические алгоритмы. Условия в алгоритмах	19	9				10
3	Тема 3. Работа со строками в языке программирования Kotlin	19	9				10
4	Тема 4. Объектно-ориентированное программирование. Инкапсуляция, полиморфизм и наследование	19	9				10
5	Тема 5. Создание Activity. Жизненный цикл Activity	19			9		10
6	Тема 6. Основы верстки Android приложений	19			9		10
7	Тема 7. Создание мобильного приложения с использованием ListView и SharedPreferences	19			9		10
8	Тема 8. Современные методы организации сетевого взаимодействия Архитектура REST	19			9		10
9	Тема 9. Создание мобильного приложения с использованием фрагментов	10					10
10	Тема 10. Создание мобильного приложения с использованием SQLite (КП)	18					18
Итого		**180**	**36**		**36**		**108**

3.3 Виды учебной работы и трудоемкость (по заочной форме обучения)

№ п/п	Вид учебной работы	Количество часов	Семестры	
			5	
1	**Аудиторные занятия**	**16**	16	
	В том числе:			
1.1	Лекции	8	8	
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	8	8	
2	**Самостоятельная работа**	**164**	164	
	Курсовой проект			
3	**Промежуточная аттестация**			
	Экзамен		экзамен	
	Итого:	**180**	180	

3.4 Тематический план изучения дисциплины (по заочной форме обучения)

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Тема 1. Типы данных, операции и условные операторы языка программирования Kotlin	18	2				16
2	Тема 2. Циклические алгоритмы. Условия в алгоритмах	18	2				16
3	Тема 3. Работа со строками в языке программирования Kotlin	18	2				16
4	Тема 4. Объектно-ориентированное программирование. Инкапсуляция, полиморфизм и наследование	18	2				16
5	Тема 5. Создание Activity. Жизненный цикл Activity	18			2		16
6	Тема 6. Основы верстки Android приложений	18			2		16
7	Тема 7. Создание мобильного приложения с использованием ListView и SharedPreferences	18			2		16
8	Тема 8. Современные методы организации сетевого взаимодействия Архитектура REST	18			2		16

9	Тема 9. Создание мобильного приложения с использованием фрагментов	16					16
10	Тема 10. Создание мобильного приложения с использованием SQLite (КП)	20					20
Итого		**180**	**8**		**8**		**164**

3.5 Содержание дисциплины

Содержание и темы лабораторных работ

ЛР-1	**Типы данных, операции и условные операторы языка программирования Kotlin**

Цель выполнения лабораторной работы: знакомство со структурой языка программирования Kotlin.

Результат: Работающая программа согласно полученному заданию

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:
- приобретение навыков: применения арифметических и логических операций; работы с простыми типами данных; использования условных операторов; использования класса Math для различных математических операций.
- Выполнение задания по индивидуальному заданию.
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое Kotlin?
2. Какой основное назначение Kotlin?
3. В чем отличие статического и динамического контента?
4. Где выполняется Kotlin-код?
5. Какие математические операторы доступны в Kotlin?

ЛР-2	**Циклические алгоритмы. Условия в алгоритмах.**

Цель выполнения лабораторной работы: Изучение структуры и синтаксиса языка Kotlin, получение базовых навыков построения программ, использования основных операторов. Закрепление знаний использования циклов и условных операторов, как основы построения алгоритмов.

Результат: Работающая программа согласно полученному заданию

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:
    - изучение математических операторов Kotlin;
    - изучение условных операторов Kotlin;
    - изучение операторов цикла в Kotlin.
- Получение варианта задания.
- Разработка программы, согласно требованиям.
- Тестирование программы, в том числе на устойчивость работы при некорректных данных.
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое переменная?
2. Что такое цикл?
3. Что такое условный оператор?
4. Какие виды циклов бывают?
5. Чем цикл с предусловием отличается от цикла с постусловием?
6. Чем отличаются операторы break и continue?
7. Что такое итерация?
8. Какой тип данных определяет условие?
9. Что такое булева алгебра?
10. Какие операции булевой алгебры (логические операции) Вы знаете?
11. Как в Kotlin округлить значение до заданной точности?
12. Как в Kotlin преобразовать вещественный тип к целому?
13. Какие виды округления используются в Kotlin?
14. Что такое конструкция выбора?
15. Какие правила преобразования типов существуют в Kotlin?

ЛР-3	**Работа со строками в языке программирования Kotlin.**

Цель выполнения лабораторной работы: Знакомство с типами данных String и StringBuffer, получение навыков создания и обработки строк

Результат: Работающая программа согласно полученному заданию

Порядок выполнения лабораторной работы:

- Подготовка к выполнению лабораторной работы – приобретение навыков работы со строками в языке программирования Kotlin, применения следующих операций: создание и инициализация объекта класса String; создание и инициализация объекта класса StringBuffer; определение размера строки; модификация строк; сравнение строк; поиск в строках; извлечение символов и подстрок из строки.
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое String в Koltin? Какой это тип данных?
2. Какие есть способы создания объекта String?
3. Напишите метод проверки, является ли строка палиндромом.
4. Как мы можем перевести строку в верхний регистр или нижний регистр?
5. В чем отличие String от StringBuffer?

ЛР-4	**Объектно-ориентированное программирование. Инкапсуляция, полиморфизм и наследование.**

Цель выполнения лабораторной работы: Знакомство с объектно-ориентированным программированием, получение навыков представления объектов реального мира из какой-либо предметной области в форме классов с набором полей и методов на языке программированияKotlin.

Результат: Работающая программа согласно полученному заданию

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:
    - приобретение навыков создания собственных классов и объектов на языке программирования Java, а также изучить основные конструкции, используемые при построении классов и объектов; разобраться с понятием инкапсуляция; разобраться с понятием полиморфизм; разобраться с

понятием наследование; изучить основные конструкции, используемые при создании интерфейсов и их реализация;
- создание базового класса по варианту.
- создание подкласса созданного базового класса.
- создание иерархии классов.
- создание интерфейса по варианту.
- реализация созданного интерфейса в базовом классе

- Защита лабораторной работы.

Контрольные вопросы:

1. Назовите принципы ООП и расскажите о каждом?
2. Дайте определение понятию “класс”?
3. Что такое поле/атрибут класса?
4. Как правильно организовать доступ к полям класса?
5. Дайте определение понятию “конструктор”?
6. Сколько раз инструкция return может быть использована в теле функции?
7. Сколько аргументов может быть передано функции?
8. Что-такое аргументы "по умолчанию"?
9. Может ли функция вообще не иметь аргументов?
10. Должны ли совпадать имена переменных-аргументов при объявлении и при вызове функции?

ЛР-5	**Создание Activity. Жизненный цикл Activity**

Цель выполнения лабораторной работы: Знакомство с пользовательским интерфейсом Android приложения

Результат: Apk-файл Android приложения

Порядок выполнения лабораторной работы:

- Подготовка к выполнению лабораторной работы: изучить структуру проекта; разобраться с понятием Activity; разобраться в каких состояниях может быть Activity.
- Создание проекта в Android Studio.
- Создание двух Activity.
- Организация переходов между Activity.
- Организация передачи параметров между Activity.
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое Activity?
2. Что делает метод onCreate?
3. Расскажите про жизненный цикл Activity?
4. Как создать Activity?
5. Как передать параметр между Activity?
6. Как сделать Activity стартовой?
7. Что такое Intent?

ЛР-6	**Основы верстки Android приложений**

Цель выполнения лабораторной работы: Изучить основы создания пользовательского интерфейса Android приложений

Результат: Apk-файл Android приложения

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:

- изучить основы верстки; научиться управлять пользовательским интерфейсом мобильного приложения;
- Создание проекта в Android Studio.
- Создание трех Activity с пользовательским интерфейсом по варианту.
- Организация переходов между Activity.
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое XML?
2. Какие типы ViewGroup бывают?
3. В чем отличие LinearLayout- и RelativeLayout?
4. Какие элементы формы существуют?
5. Как передает свое значение каждый элемент формы в Мобильное приложение?
6. Как перейти на другое Activity?

ЛР-7	**Создание мобильного приложения с использованием ListView и SharedPreferences**

Цель выполнения лабораторной работы: Приобретение навыков построения многооконного пользовательского интерфейса, хранение данных во внутренней памяти устройства

Результат: Мобильное приложение «Каталог фильмов». Apk-файл Android приложения.

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:
  - изучить создание пользовательского интерфейса со списком элементов;
  - приобретение навыков хранения данных во внутренней памяти устройства.
- Создание проекта в Android Studio.
- Создание экрана для авторизации в приложении.
- Сохранение идентификатора входа в приложении.
- Создание экрана со списком фильмов.
- Организация перехода при нажатии на конкретный фильм из списка на новый экран, с передачей данных о фильме.
- Создание меню в приложении для перехода между экранами.
- Создание функции выхода из приложения.
- Создание экрана «О приложении»
- Защита лабораторной работы.

Контрольные вопросы:

1. Как использовать SharedPreferences в Android для хранения, извлечения и редактирования значений?
2. Как удалить общие настройки?
3. Какая разница между commit() и apply() в общих предпочтениях?
4. Как получить SharedPreferences из PreferenceActivity в Android?
5. В чем разница между getDefaultSharedPreferences и getSharedPreferences в Android?
6. Как Android SharedPreferences сохраняет/сохраняет объект?
7. Где хранятся общие настройки?
8. Должен ли доступ к SharedPreferences делать из потока пользовательского интерфейса?

ЛР-8	**Современные методы организации сетевого взаимодействия Архитектура REST**

Цель выполнения лабораторной работы: Получение представлений об организации сетевого взаимодействия в Koltin, принципов работы

Результат: Apk-файл Android приложения

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:
  - изучение способов построение сетевого взаимодействие;
  - изучение приемов работы REST;
- Разработка программы для получения информации от сервера.
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое REST?
2. Какие способы организации сетевого взаимодействия вы знаете?
3. Можно ли сделать запрос в main потоке?
4. Что такое сериализации?
5. Как обработать ответ от сервера?

ЛР-9	**Создание мобильного приложения с использованием фрагментов.**

Цель выполнения лабораторной работы: Приобретение навыков построения пользовательского интерфейса с использованием фрагментов.

Результат: Мобильной приложение «Каталог фильмов и сериалов».Apk-файл Android приложения.

Порядок выполнения лабораторной работы:

- Подготовка к выполнению к работе, в том числе:
  - приобретение навыков создания пользовательского интерфейса с использованием фрагментов;
  - приобретение навыков создания навигации в приложении при помощи BottomNavigationView.
- Создание проекта в Android Studio.
- Создание экрана для авторизации в приложении.
- Сохранение идентификатора входа в приложении.
- Создание экрана для хранения фрагментов
- Создание фрагмента со списком фильмов.
- Организация перехода при нажатии на конкретный фильм из списка на новый экран, с передачей данных о фильме.
- Создание фрагмента со списком сериалов.
- Создание меню в приложении для сортировки фильмов/сериалов.
- Создание функции выхода из приложения.
- Создание экрана «О приложении»
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое BottomNavigationView?
2. Что такое DDMS? Опишите некоторые из его возможностей.?
3. В чем разница между фрагментом и активностью? Объясните взаимосвязь между ними.?
4. В чем разница между Serializable и Parcelable? Что предпочтительнее использовать в Android?
5. В чем разница между Service и IntentService? Как они используются?

ЛР-10	**Создание мобильного приложения с использованием SQLite**

Цель выполнения лабораторной работы: Приобретение навыков работы с базой данных в мобильном приложении

Результат: Мобильной приложение «Каталог фильмов и сериалов».Apk-файл Android приложения.

Порядок выполнения лабораторной работы:

- Создание проекта в Android Studio.
- Создание экрана для авторизации в приложении.
- Создание экрана для регистрации в приложении.
- Сохранение идентификатора входа в приложении.
- Создание экрана для хранения фрагментов
- Создание фрагмента со списком фильмов.
- Организация перехода при нажатии на конкретный фильм из списка на новый экран, с передачей данных о фильме.
- Создание фрагмента со списком сериалов.
- Создание фрагмента для отображения профиля пользователя.
- Создание меню в приложении для сортировки фильмов/сериалов.
- Создание функции выхода из приложения.
- Создание экрана «О приложении»
- Защита лабораторной работы.

Контрольные вопросы:

1. Что такое SQL? Основные операторы языка SQL?
2. Оператор INSERT. Для чего он предназначен, какие формы может иметь?
3. Источники данных для оператора INSETRT: как добавить записи в таблицу из другой таблицы?
4. Оператор UPDATE: назначение, синтаксис, форма записи?
5. Формы оператора SELECT?
6. Условие отбора в операторе SELECT?
7. Функции в операторе SELECT?
8. Сортировка записей и ее типы?
9. Группировка записей: назначение и отличие от сортировки?
10. РаботаSQL- функций с группировкой и без группировки?
11. Ограничение результата SQL-запроса: способы и параметры?
12. Псевдонимы полей в операторе SELECT?
13. SQLiteOpenHelper – назначение и способ применения?
14. Отличие процедурного и объектно-ориентированного стиля в SQLiteOpenHelper?
15. Буферизированные и не буферизированные запросы в SQLiteOpenHelper?
16. Функции и методы расширения SQLiteOpenHelper?
17. Какова цель и преимущества использования псевдопеременных в SQL?
18. В каких случаях имеет смысл делать отдельный SQL-запрос для определения текущей записи из списка выводимых записей таблицы базы данных?

4 Учебно-методическое и информационное обеспечение

4.1 Основная литература

1. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2023. — 175 с. — (Профессиональное образование). — ISBN 978-5-534-10680-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/518008

2. Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2023. — 320 с. — (Высшее образование). — ISBN 978-5-534-02444-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/511712

4.2 Дополнительная литература

1. Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебное пособие для вузов / В. В. Соколова. — Москва : Издательство Юрайт, 2022. — 175 с. — (Высшее образование). — ISBN 978-5-9916-6525-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/490305

2. Черпаков, И. В. Основы программирования : учебник и практикум для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2023. — 219 с. — (Высшее образование). — ISBN 978-5-9916-9983-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/511703

4.4. Электронные образовательные ресурсы

1. Разработка мобильных приложений для IOS

https://lms.mospolytech.ru/course/view.php?id=6375

5 Материально-техническое обеспечение

5.1 Требования к оборудованию и помещению для занятий

Лабораторные работы и самостоятельная работа студентов должны проводиться в специализированной аудитории, оснащенной современной оргтехникой и персональными компьютерами с программным обеспечением в соответствии с тематикой изучаемого материала. Число рабочих мест в аудитории должно быть достаточным для обеспечения индивидуальной работы студентов. Рабочее место преподавателя должно быть оснащены

современным компьютером с подключенным к нему проектором на настенный экран, или иным аналогичным по функциональному назначению оборудованием.

5.2 Требования к программному обеспечению

Для выполнения лабораторных работ и самостоятельной работы необходимо следующее программное обеспечение:

1. Microsoft Windows.
2. Notepad++.
3. Android Studio.
4. Веб-браузер, Chrome.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

1. При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы – теоретические сведения, задачи и др. При проведении занятия следует контролировать подачу материала и решение заданий с учетом учебного времени, отведенного для занятия.

2. При проверке работ и отчетов следует учитывать не только правильность выполнения заданий, но и оптимальность выбранных методов решения, правильность выполнения всех его шагов.

6.2 Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи учебным планом. Основой теоретической подготовки студентов являются аудиторные занятия, семинары и практики.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, дорабатывают конспекты и записи, готовятся к промежуточной аттестации, а также самостоятельно изучают отдельные темы учебной программы.

На занятиях студентов, в том числе предполагающих практическую деятельность, осуществляется закрепление полученных, в том числе и в процессе самостоятельной работы, знаний. Особое внимание обращается на развитие умений и навыков установления связи положений теории с профессиональной деятельностью будущего специалиста.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).

Текущий контроль осуществляется на аудиторных занятиях, промежуточный контроль осуществляется на зачете в письменной (устной) форме.

Критериями оценки результатов самостоятельной работы студента являются:

- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность компетенций;
- оформление материала в соответствии с требованиями.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- выполнение лабораторных работ,
- экзамен,
- курсовой проект.

7.2 Шкала и критерии оценивания результатов обучения

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине.

Показатель	**Критерии оценивания**			
	2	**3**	**4**	**5**
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач				
ИУК-1.1. Анализирует задачу, выделяя ее базовые составляющие. ИУК-1.2. Осуществляет поиск, критически оценивает, обобщает, систематизирует и ранжирует информацию, требуемую для решения поставленной задачи.	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие материалу дисциплины знаний, указанных в индикаторах компетенций дисциплины.	Обучающийся демонстрирует неполное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины. Допускаются значительные ошибки, проявляется недостаточность знаний, по ряду	Обучающийся демонстрирует частичное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины. Но допускаются незначительные ошибки, неточности, затруднения при	Обучающийся демонстрирует полное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины. Свободно оперирует приобретенными знаниями.

ИУК-1.3. Рассматривает и предлагает рациональные варианты решения поставленной задачи, используя системный подход, критически оценивает их достоинства и недостатки.		показателей, обучающийся испытывает значительные затруднения при оперировании знаниями при их переносе на новые ситуации.	аналитических операциях.	
ПК-1. Способен разрабатывать требования и проектировать программное обеспечение				
ИПК-1.1. Знает возможности существующей программно-технической архитектуры; возможности современных и перспективных средств разработки программных продуктов, технических средств; методологии разработки программного обеспечения и технологии программирования ; методологии и технологии проектирования и использования баз данных; языки формализации функциональных спецификаций; методы и приемы формализации задач; методы и	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие материалу дисциплины знаний, указанных в индикаторах компетенций дисциплины.	Обучающийся демонстрирует неполное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины. Допускаются значительные ошибки, проявляется недостаточность знаний, по ряду показателей, обучающийся испытывает значительные затруднения при оперировании знаниями при их переносе на новые ситуации.	Обучающийся демонстрирует частичное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины. Но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях.	Обучающийся демонстрирует полное соответствие следующих знаний, указанных в индикаторах компетенций дисциплины. Свободно оперирует приобретенными знаниями.

средства проектирования программного обеспечения; методы и средства проектирования программных интерфейсов; методы и средства проектирования баз данных; принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методы и средства проектирования программного обеспечения и баз данных; методы и средства проектирования программных интерфейсов. ИПК-1.2. Умеет проводить анализ исполнения требований; вырабатывать варианты реализации требований; проводить оценку				

и обоснование рекомендуемых решений; осуществлять коммуникации с заинтересованным и сторонами; выбирать средства реализации требований к программному обеспечению; вырабатывать варианты реализации программного обеспечения; проводить оценку и обоснование рекомендуемых решений; осуществлять коммуникации с заинтересованным и сторонами; использовать существующие типовые решения и шаблоны проектирования программного обеспечения; применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов; осуществлять коммуникации с заинтересованным и сторонами.				

ИПК-1.3. Владеет современным инструментарием и средами разработки программного кода; современным инструментарием и средами проектирования программного кода, методами тестирования ПО.				

Шкала оценивания результатов промежуточной аттестации и её описание:

Форма промежуточной аттестации: экзамен.

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю), при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю) методом экспертной оценки. По итогам промежуточной аттестации по дисциплине (модулю) выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине.

Шкала оценивания	**Описание**
Отлично	Среднее значение для всех формируемых на момент проведения аттестации уровней компетенций – 5. Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Среднее значение для всех формируемых на момент проведения аттестации уровней компетенций – 4. Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений,

	навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Среднее значение для всех формируемых на момент проведения аттестации уровней компетенций – 3. Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не достигнуто пороговое значение хотя бы для одного уровня формируемых на момент проведения аттестации компетенций. Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

7.3 Оценочные средства

7.3.1 Экзаменационное задание.

Экзаменационное задание выполняется студентом индивидуально, по итогам изучения дисциплины или ее части. При этом достижение порогового результата работы над экзаменационным заданием соответствует описанному в п. 3 данного документа этапу освоения соответствующих компетенций на базовом или продвинутом уровне.

Базовый уровень: способность выполнять полученное задание, применяя полученные знание и умения на практике, владеть соответствующими индикаторами компетенции при выполнении задания.

Продвинутый уровень: способность выполнять полученное задание и решать самостоятельно сформированные задачи, применяя полученные знание и умения на практике. Уверенно владеть соответствующими индикаторами компетенции при выполнении задания, комбинировать их между собой и с индикаторами других компетенций для достижения проектных результатов.

Форма экзаменационного задания выбирается преподавателем и утверждается на заседании кафедры. Экзамен может проходить в следующих формах и с использованием следующих оценочных средств.

Форма	**Представление оценочного средства в ФОС**
Устная.	Банк контрольных вопросов, соответствующих отдельным темам дисциплины (см. п. 4 настоящего документа). Вопросы формируют экзаменационный билет (см. ниже), состоящий из теоретических вопросов и практических заданий (типовые практические задания представлены ниже). Билеты, включая вопросы и практические задания, формируются преподавателем и утверждаются на заседании кафедры. В

	них могут быть включены дополнительные контрольные вопросы и задания, не требующие у студентов наличия не формируемых данной дисциплиной компетенций или более высоких этапов сформированности формируемых. Для ответа на каждый вопрос и для решения любого практического задания студент должен находится на требуемом для данной дисциплине уровне сформированности всех соответствующих ей компетенций: каждый вопрос и задание проверяет уровень сформированности всех соответствующих данной дисциплине компетенций.
Письменная.	Оценочное средство полностью соответствует оценочным средствам устной формы задания.
Практико-ориентированная (формат WorldSkills).	Типовое задание практико-ориентированного экзамена. Задание практико-ориентированного формируется преподавателем на основе типового и Методических рекомендаций по разработке задания ПОЭ, утверждаются на заседании кафедры. Задание ПОЭ проверяет уровень сформированности всех соответствующих дисциплине компетенций.

7.3.2 Типовой экзаменационный билет.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

по дисциплине

«Основы разработки мобильных приложений»

направление подготовки 09.03.01 «Информатика и вычислительная техника»

ВОПРОСЫ

1. В чем отличие статического и динамического контента?
2. Сколько аргументов может быть передано функции?
3. Какие параметры присутствуют у функции microtime() и для чего они используются?
4. Практическое задание. Разработать программу – обработчик полученных оценок за экзамен. На входе через GET-параметры последовательно передается информация об оценке и ФИО студента. После каждой обработки должна выводиться столбчатая диаграмма с распределением оценок. Под каждым столбцом выводятся соответствующие ФИО студента.

Утверждено: _____/_____/ «___»_____202__г.

7.3.3 Типовые практические задания.

1. Разработать программу-обработчик полученных оценок за экзамен. На входе через GET-параметры последовательно передается информация об оценке и ФИО студента. После каждой обработки должна выводится столбчатая диаграмма с распределением оценок. Под каждым столбцом выводятся соответствующие ФИО.

2. Разработать программу, формирующую для указанного URL список располагающихся на соответствующей веб-странице уникальных ссылок.

3. Разработать программу, строящую гистограмму количества слов в тексте без учета регистра символов.

4. Разработать программу, реализующую алгоритм чет-нечет.

5. Разработать программу, загружающую файл со списком формата "e-mail:сообщение" и рассылающую адресатам соответствующие сообщения.

Типовое задание в практико-ориентированном формате

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

ЭКЗАМЕНАЦИОННОЕ ЗАДАНИЕ
по дисциплине
«Основы разработки мобильных приложений»
направление подготовки 09.03.01 Информатика и вычислительная техника

ВРЕМЯ ВЫПОЛНЕНИЯ ЗАДАНИЯ.

6 астрономических часов без учета перерыва на обед.

ЗАДАНИЕ ЭКЗАМЕНА.

Компания Marvel, впечатленная достигнутыми результатами в разработки мобильных приложений, поручает Вам создать мобильное приложение, состоящие из следующих экранов:

- Экран ввода пароля для входа в приложение;
- Экран со списком комиксов;
- Экран с детальной информацией о комиксе;
- Экран со списком персонажей комиксов;
- Экран с детальной информацией о персонаже комикса;
- Экран со списком избранных персонажей и комиксов;
- Экран с информацией о приложении.

1. Содержание экрана ввода пароля в приложение:

1.1. Логотип
1.2. Поле ввода пароля (пароль - admin);
1.3. Кнопка входа.
2. Содержание элемента списка персонажей / комиксов:
2.1. Изображение персонажа / комикса;
2.2. Наименование персонажа / комикса;
2.3. Количество комиксов с персонажем / цена комикса;
2.4. Кнопка для добавления в избранное.
3. Содержание экрана с детальной информацией персонажа / комикса:
3.1. Изображение персонажа / комикса;
3.2. Наименование персонажа / комикса;
3.3. Полное описание персонажа / комикса;
3.4. Кнопка для добавления в избранное;
3.5. Количество комиксов с персонажем / цена комикса.
4. Содержание экрана со списком избранных персонажей и комиксов:
4.1. Список избранных персонажей и комиксов.
5. Содержание экрана с информацией о приложении:
5.1. Логотип;
5.2. Информация о приложении;
5.3. Дата создания приложения;
5.4. Информация о Вас.

Переходы между экранами должны быть реализованы через нижнюю панель навигации (BottomNavigationView), позволяющая переключаться между экранами приложения в одно касание.

В рамках поставленной задачи необходимо реализовать на некоторых экранах следующие функции.
1. На экране ввода пароля для входа в приложение:
1.1. сохранение входа в приложение;
2. На экране с детальной информацией персонажа / комикса:
2.1. Добавление в избранное;
2.2. Удаление из избранного.
3. На экране со списком персонажей / комиксов:
3.1. открытие экрана с детальной информацией об элементе;
3.2. сортировка списка персонажей по алфавиту (по возрастанию и убыванию);
3.3. сортировка списка комиксов по цене (по возрастанию и убыванию).

Приложение должно уметь кешировать данные, то есть работать в offline режиме.

ВХОДНЫЕ ДАННЫЕ.

В качестве исходного материала для работы необходимо использовать:

- Предоставленные медиа файлы (изображения и т д).;
- Документация Marvel API;
- Ключи доступа к Marvel API.

ВЫХОДНЫЕ ДАННЫЕ.

В результате работы в папке "*Рабочий стол/ФИО - Номер группы* " должны быть предоставлены следующие материалы:

- Архив проекта Android Studio;
- Apk-файл созданного приложения;

УСЛОВИЯ ВЫПОЛНЕНИЯ РАБОТЫ.

Для выполнения задания может использоваться следующее программное обеспечение:

- Android Studio;
- Notepad++;
- Adobe Photoshop;
- Браузер.

Не разрешается запускать и использовать другие программы. Не допускается использование *flash*-накопителей, материалов на серверах. Разрешается использование лекций учебного курса и печатных учебников по Java, Android Studio, Разработки мобильных приложений под ОС Android.

КРИТЕРИИ ОЦЕНКИ ЗАДАНИЯ.

При проверке работы будет последовательно выполнены описанные в таблице действия. При соответствии выполняемого действия ожидаемому результату подтверждается указанная в правой колонке оценка. В случае ошибочного результата или отсутствия реализации проверяемой функции, вследствие чего выполнить требуемые действия невозможно, итоговая оценка экзамена будет взята как предыдущая подтвержденная оценка. В этом случае дальнейшая проверка работы осуществляться не будет.

№	Выполняемые действия	Проверяемая функция	Ожидаемый результат	Оценка
1.	Просмотр строковых ресурсов проекта(/res/values/strings.xml)	Все строки вынесены в ресурсы	Все строки, используемые в приложение находятся в ресурсах(/res/values/strings.xml)	неудовлетворительно
2.	На экране ввода пароля для входа в приложение нажатие на кнопку входа(с соответствующей надписью или графическим обозначением)	Вход в приложение	При вводе верного пароля происходит вход в приложение, при вводе неверного пароля отображается уведомление с соответствующим сообщением.	
3.	Повторный вход в приложение	Повторный вход в приложение	При повторном входе в приложение пароль не запрашивается	

4.	Нажатие на нижнюю панель навигации	Открытие экрана	Наличие всех пунктов меню для перехода на другой экран	
5.	Открытие экрана о приложении	Открытие экрана	Наличие экрана о приложении	удовлетворительно
6.	Отображение списка персонажей комиксов	Корректное отображение элемента списка	Список реализован при помощи виджета ListView либо RecyclerView. Внешний вид элемента списка не меняется при разных размерах экрана устройства.	
7.	Нажатие на элемент списка персонажей комиксов	Переход на экран детального отображения	Реализован интерфейс OnItemClickListener для перехода на экран детального отображения	
8.	Отображение списка комиксов	Корректное отображение элемента списка	Список реализован при помощи виджета ListView либо RecyclerView. Внешний вид элемента списка не меняется при разных размерах экрана устройства.	хорошо
9.	Нажатие на элемент списка комиксов	Переход на экран детального отображения	Реализован интерфейс OnItemClickListener для перехода на экран детального отображения	
10.	Нажатие на кнопку избранное на элементе списка	Добавление / удаление элемента	Реализована логика добавления в избранное и удаления из избранного	
11.	Открытие экрана избранное	Открытие экрана	Отображение списка с избранными элементами	

12.	Активаций элемента управления (кнопка или элемент списка) "Сортировка по алфавиту по возрастанию" (с соответствующей надписью или графическим обозначением)	Сортировка списка персонажей по алфавиту (по возрастанию)	Строки списка сортируются по алфавиту в порядке возрастания. Содержимое каждой строки не изменяется – меняется лишь порядок строк	
13.	Активаций элемента управления (кнопка или элемент списка) "Сортировка по алфавиту по убыванию " (с соответствующей надписью или графическим обозначением)	Сортировка списка персонажей по алфавиту (по убыванию)	Строки списка сортируются по алфавиту в порядке возрастания. Содержимое каждой строки не изменяется – меняется лишь порядок строк	отлично
14.	Активаций элемента управления (кнопка или элемент списка) "Сортировка по цене по возрастанию" (с соответствующей надписью или графическим обозначением)	Сортировка списка комиксов по цене (по возрастанию)	Строки списка сортируются по алфавиту в порядке возрастания. Содержимое каждой строки не изменяется – меняется лишь порядок строк	
15.	Активаций элемента управления (кнопка или элемент списка) "Сортировка по цене по убыванию " (с соответствующей надписью или графическим обозначением)	Сортировка списка комиксов по цене (по убыванию)	Строки списка сортируются по алфавиту в порядке возрастания. Содержимое каждой строки не изменяется – меняется лишь порядок строк	
16.	Открытие приложения без доступа к Интернету	Кеширование информации	Отображение всех персонажей, комиксов без доступа в интернет, кроме первого запуска приложения	

17.	Просмотр архитектуры приложения	Реализация одного из паттернов проектирования MVP/ MVVM	Реализован один из паттернов проектирования MVP/ MVVM	

Набранный уровень достижения, и соответствующая им оценка имеет рекомендательный характер – экзаменатор имеет право скорректировать оценку в ту или иную сторону.