

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 14.05.2024 09:46:12  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

**МИНОБРНАУКИ РОССИИ**  
**Федеральное государственное бюджетное образовательное**  
**учреждение высшего образования**  
**«Юго-Западный государственный университет»**  
**(ЮЗГУ)**

Кафедра вычислительной техники

УТВЕРЖДАЮ

Проректор по учебной работе  
О.Г. Локтионова

« 25 » 02



**Разработка мобильных приложений**

Методические указания к выполнению самостоятельных работ  
по дисциплине «Разработка мобильных приложений» для  
студентов направления подготовки 09.03.01

Курск 2022

УДК 004

Составитель: А.В. Киселев, О.О. Яночкина

Рецензент

Кандидат технических наук, доцент *Ю.А. Халин*

**Разработка мобильных приложений:** методические указания к выполнению самостоятельных работ по дисциплине «Разработка мобильных приложений» / Юго-Зап. гос. ун-т; сост.: А.В. Киселев, О.О. Яночкина. – Курск, 2022. 10 с.: Библиогр.: с. 10.

Методические указания соответствуют требованиям рабочей программы по дисциплине «Разработка мобильных приложений» и разработанным оценочным средствам.

Предназначены для студентов направления подготовки 09.03.01 очной формы обучения.

Содержат основные сведения об организации самостоятельной работы студентов. Описаны основные виды самостоятельной работы. Приведены вопросы для самостоятельного изучения при подготовке к собеседованию и зачету.

Текст печатается в авторской редакции

Подписано в печать . Форма 60x84 1/16.

Усл. печ. л. . Уч.-изд.л. 0,5 . Тираж \_\_\_ экз. Заказ. 899

Бесплатно

Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## Содержание самостоятельной работы

	Тема СРС	Задание
1	Основы разработки приложений для ОС Android	<p>Цель задания: Исследовать и описать ключевые этапы развития методов и инструментов разработки приложений для операционной системы Android, начиная с ранних версий и до современных трендов.</p> <p>Задачи:</p> <ol style="list-style-type: none"><li>1. Изучить историю операционной системы Android, начиная с первой версии.</li><li>2. Проанализировать ключевые инструменты и платформы разработки приложений для Android, которые использовались на различных этапах ее развития.</li><li>3. Исследовать изменения в подходах к проектированию пользовательского интерфейса в Android-приложениях с течением времени.</li><li>4. Рассмотреть влияние новых технологий, таких как Kotlin, Jetpack, и AR/VR, на разработку приложений для Android.</li><li>5. Проанализировать основные принципы и практики тестирования мобильных приложений под Android и их эволюцию.</li></ol> <p>Требования к выполнению задания:</p> <ol style="list-style-type: none"><li>1. Реферат должен содержать введение, описание основных этапов эволюции разработки приложений для Android, заключение и список использованных источников.</li><li>2. Объем реферата - не менее 1500 слов.</li><li>3. Использование актуальных источников информации, в том числе научных статей, книг, официальной документации Android и других авторитетных ресурсов.</li><li>4. Наличие иллюстративного материала (скриншоты, диаграммы, графики) для визуализации ключевых моментов развития разработки под Android.</li><li>5. Аккуратное оформление работы с соблюдением требований к оформлению научных текстов (шрифт, отступы, списки использованных</li></ol>

		<p>источников и т.д.).</p> <p>6. В тексте реферата должны быть использованы термины и понятия, характерные для разработки приложений для Android.</p>
2	Активности и интенты.	<p>Цель задания: Систематизировать и сравнить основные характеристики и применения активностей и интентов в разработке приложений для операционной системы Android.</p> <p>Задачи:</p> <ol style="list-style-type: none"> <li>1. Создать таблицу сравнения, включающую различные аспекты активностей и интентов в Android.</li> <li>2. Проанализировать основные характеристики активностей и интентов.</li> <li>3. Выделить особенности использования активностей и интентов в разработке Android-приложений.</li> <li>4. Описать примеры сценариев, где активности и интенты используются взаимосвязанно.</li> <li>5. Подготовить краткое объяснение каждой характеристики в таблице.</li> </ol> <p>Требования к выполнению задания:</p> <p>Таблица должна включать следующие столбцы:</p> <ol style="list-style-type: none"> <li>1. Название характеристики,</li> <li>2. Описание активностей,</li> <li>3. Описание интентов.</li> <li>4. В таблице нужно учитывать такие характеристики, как назначение, способы инициализации, передача данных и т.д.</li> <li>5. Наличие примеров использования активностей и интентов в различных сценариях.</li> <li>6. Краткое и информативное описание каждой характеристики, чтобы представить основное предназначение и особенности использования.</li> <li>7. Четкость и логичность структуры таблицы для удобства восприятия информации.</li> <li>8. При оформлении таблицы следует учитывать единый стиль и соответствие стандартам оформления данных.</li> <li>9.</li> </ol>

3	<p>Пример простого приложения в архитектуре MVC</p>	<p>Напишите доклад, представляющий собой описание процессов архитектурно-ориентированного проектирования. Ваша цель - подробно и наглядно представить основные этапы, методы и подходы, используемые в данном процессе.</p> <p>Примерное содержание доклада:</p> <ol style="list-style-type: none"> <li>1. Введение: <ul style="list-style-type: none"> <li>• Вступление должно содержать обзор архитектурно-ориентированного проектирования и его важности в разработке программного обеспечения.</li> <li>• Необходимо четко сформулировать цель и задачи данного задания.</li> </ul> </li> <li>2. Основная часть: <ul style="list-style-type: none"> <li>• Обоснованно опишите каждый этап процесса архитектурно-ориентированного проектирования:</li> <li>• Анализ требований и создание архитектурных целей.</li> <li>• Выбор архитектурного стиля и паттернов проектирования.</li> <li>• Создание и моделирование архитектурных представлений.</li> <li>• Проектирование системной структуры и компонентов.</li> <li>• Разработка и документирование архитектурных решений.</li> <li>• Верификация и валидация архитектуры.</li> <li>• Для каждого этапа приведите подробное описание, включая его цели, основные задачи и инструменты, используемые на этапе.</li> <li>• Приведите примеры применения каждого этапа процесса в реальных проектах.</li> </ul> </li> <li>3. Заключение: <ul style="list-style-type: none"> <li>• Подведите итоги представленных вами информации.</li> <li>• Подчеркните важность архитектурно-ориентированного проектирования и его преимущества.</li> </ul> </li> </ol> <p>Требования к докладу:  Материал должен содержать графические схемы,</p>
---	---	--

		<p>диаграммы и примеры, чтобы визуально иллюстрировать каждый этап процесса.</p> <p>Ссылайтесь на качественные источники информации, такие как научные статьи, книги или официальные документы для обоснования ваших утверждений.</p> <p>Формат представления материала:</p> <p>Возможными форматами может быть презентация или текстовый документ, описывающий каждый этап процесса.</p>
4	Класс View и его возможности	<p>Проведите анализ и сравнительное исследование процессов управления требованиями.</p> <p>Требования к заданию:</p> <ol style="list-style-type: none"> <li>1. Описание цели и задачи исследования: <ul style="list-style-type: none"> <li>• Целью данного исследования является анализ и сравнение процессов управления требованиями в инженерии требований.</li> <li>• Задачи исследования включают: <ul style="list-style-type: none"> <li>• Выявление основных процессов управления требованиями в инженерии требований.</li> <li>• Сравнительный анализ процессов управления требованиями с целью выявления их преимуществ и недостатков.</li> <li>• Исследование лучших практик и методов в управлении требованиями.</li> </ul> </li> </ul> </li> <li>2. Описание методологии исследования: <ul style="list-style-type: none"> <li>• Исследование будет проводиться путем анализа академической литературы, стандартов и публикаций, связанных с инженерией требований.</li> <li>• Также будет проведено сравнительное исследование практического применения процессов управления требованиями в различных проектах.</li> <li>• Для анализа и сравнения процессов будут использованы расчетные показатели, такие как эффективность, эффективность и надежность процессов.</li> </ul> </li> <li>3. Описание этапов исследования: <ul style="list-style-type: none"> <li>• Сбор и анализ литературных источников, стандартов и публикаций, связанных с инженерией требований.</li> <li>• Изучение примеров практического применения процессов управления требованиями в различных</li> </ul> </li> </ol>

		<p>проектах.</p> <ul style="list-style-type: none"> <li>• Сравнительный анализ процессов управления требованиями, включая выявление их преимуществ и недостатков.</li> <li>• Исследование лучших практик и методов в управлении требованиями, основанное на полученных данных.</li> <li>• Подготовка и написание исследовательского отчета с анализом и выводами.</li> </ul> <p>4. Ожидаемые результаты:</p> <ul style="list-style-type: none"> <li>• Обзор и анализ основных процессов управления требованиями в инженерии требований.</li> <li>• Сравнительный анализ преимуществ и недостатков различных процессов управления требованиями.</li> <li>• Идентификация лучших практик и методов в управлении требованиями.</li> <li>• Исследовательский отчет с подробным описанием анализа и выводами.</li> </ul>
5	Работа с ресурсами	<p>Цель задания: Создать иерархию классов для простого приложения с использованием архитектуры Model-View-Controller (MVC).</p> <p>Задачи:</p> <ol style="list-style-type: none"> <li>1. Создать три основных компонента архитектуры MVC: модель (Model), представление (View) и контроллер (Controller).</li> <li>2. Реализовать логику каждого компонента в соответствии с их назначением в архитектуре MVC.</li> <li>3. Установить взаимодействие между компонентами медиатором (например, событиями или наблюдателями).</li> <li>4. Создать простое приложение с использованием разработанных компонентов, где пользователь может взаимодействовать с интерфейсом.</li> <li>5. Разработать функциональность для изменения состояния модели при взаимодействии пользователя с приложением через контроллер.</li> <li>6. Обеспечить обновление представления на основе изменений в модели и контроллере с</li> </ol>

		<p>соблюдением принципов MVC.</p> <p><u>Пример простого приложения:</u>  Тема приложения: Простой список дел.</p> <p>1. Модель (Model):  Создать класс <code>TodoItem</code> для представления элемента списка дел (<code>id</code>, название, статус выполнения и т.д.).  Реализовать класс <code>TodoListModel</code>, который будет содержать список <code>TodoItem</code>'ов и методы для добавления, удаления и изменения элементов.</p> <p>2. Представление (View):  Создать интерфейс, отображающий список дел и элементы <code>todo</code>.  Реализовать класс <code>TodoListView</code>, который будет отображать элементы списка и обновлять интерфейс в соответствии с изменениями модели.</p> <p>3. Контроллер (Controller):  Создать контроллер <code>TodoListController</code>, который будет обрабатывать действия пользователя (добавление нового элемента, отметка выполненного, удаление элемента).  Организовать взаимодействие между контроллером, моделью и представлением.</p> <p>Требования к заданию:</p> <ol style="list-style-type: none"> <li>1. Программа должна быть написана на выбранном вами языке программирования (например, Java, Python, JavaScript и т.д.).</li> <li>2. Обеспечить ясное разделение обязанностей между классами <code>Model</code>, <code>View</code> и <code>Controller</code> в соответствии с паттерном MVC.</li> <li>3. Продемонстрировать работу приложения, где изменения в модели приводят к автоматическому обновлению представления через контроллер.</li> <li>4. Документировать код и внедрить комментарии для объяснения основного функционала каждого класса.</li> <li>5. Обеспечить простой и интуитивно понятный пользовательский интерфейс для управления списком дел.</li> </ol>
6	Хранение данных	Цель задания: Создать простое приложение, использующее хранение данных для сохранения и



загрузки информации.

Задачи:

1. Создать структуру базы данных или файловой системы для хранения данных приложения.
2. Реализовать интерфейс для взаимодействия с данными (например, формы для ввода данных и отображения результата).
3. Написать логику для сохранения данных в хранилище при создании новой записи или редактировании существующей.
4. Обеспечить функциональность по загрузке данных из хранилища для отображения на интерфейсе приложения.
5. Учитывать вопросы безопасности и целостности данных при работе с хранилищем.

Тема приложения: Личный дневник событий.

Задание:

1. Хранилище данных: Создать базу данных SQLite или файловую систему для хранения записей дневника (заголовок, текст, дата создания).
2. Интерфейс и логика: Разработать интерфейс приложения, где пользователь может создавать, просматривать и редактировать записи дневника.
3. Функционал: Добавить возможность сохранения новых записей в базу данных и их обновление.
4. Реализовать функцию загрузки записей из базы данных для отображения на главном экране.

Требования к заданию:

1. Программа должна быть написана на выбранном вами языке программирования (например, Python, Java, JavaScript).
2. Обеспечить правильное размещение и организацию данных в хранилище.
3. Реализовать интерфейс, позволяющий пользователю взаимодействовать с данными (добавление, редактирование, удаление записей).
4. Обеспечить сохранность данных при перезапуске приложения.

		<ol style="list-style-type: none"> <li>5. Добавить проверки на валидность данных при сохранении и редактировании записей.</li> <li>6. Документировать код и внедрить комментарии для объяснения основного функционала каждого модуля.</li> </ol>
7	<p>Пример приложения, использующего БД для хранения данных</p>	<p>Цель задания: Создать простое приложение, которое использует базу данных для хранения и управления данными.</p> <p>Задачи:</p> <ol style="list-style-type: none"> <li>1. Создать базу данных с необходимой структурой для хранения данных приложения.</li> <li>2. Разработать интерфейс приложения для взаимодействия с данными, включая отображение, добавление, редактирование и удаление информации.</li> <li>3. Написать код для подключения к базе данных и выполнения запросов (например, SQL запросов).</li> <li>4. Обеспечить сохранение данных в базе данных при создании новых записей или редактировании существующих.</li> <li>5. Реализовать функциональность для извлечения данных из базы данных и отображения их на интерфейсе приложения.</li> </ol> <p>Тема приложения: Управление контактами.</p> <p>Задание:</p> <ol style="list-style-type: none"> <li>1. База данных: Создать базу данных SQLite или MySQL с таблицей contacts, содержащую поля: id, name, email, phone.</li> <li>2. Интерфейс и логика: Разработать интерфейс для отображения списка контактов, добавления новых контактов, редактирования существующих и удаления контактов.</li> <li>3. Функционал: Реализовать функции CRUD (create, read, update, delete) для управления контактами в базе данных.</li> <li>4. Обеспечить сохранность данных при перезапуске приложения.</li> </ol> <p>Требования к заданию:</p>

		<ol style="list-style-type: none"><li>1. Программа должна быть написана на выбранном языке программирования (например, Python, Java, JavaScript).</li><li>2. Организовать подключение к базе данных и выполнение SQL запросов для работы с данными.</li><li>3. Реализовать интерфейс с использованием простого графического интерфейса или через командную строку.</li><li>4. Обеспечить целостность данных в базе (например, проверки на уникальность и целостность данных).</li><li>5. Добавить возможность поиска контактов по определенным критериям.</li><li>6. Документировать код и внедрить комментарии для объяснения работы каждого компонента приложения.</li></ol>
--	--	--

## Перечень литературы

1. Соколова, В. В. Разработка мобильных приложений: учебное пособие / В. В. Соколова; Национальный исследовательский Томский государственный университет. - Томск : Издательство Томского политехнического университета, 2015. - 176 с. : ил., табл., схем. - URL: <http://biblioclub.ru/index.php?page=book&id=442808> (дата обращения: 12.08.2020) . - Режим доступа: по подписке. - Библиогр. в кн. - ISBN 978-5-4387-0369-3 : Б. ц. - Текст : электронный.

2. Введение в разработку приложений для ОС Android : учебное пособие / Ю. В. Березовская, О. А. Юфрякова, В. Г. Вологодина и др. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 434 с. : ил. - URL: <https://biblioclub.ru/index.php?page=book&id=428937> (дата обращения: 12.07.2021). – Режим доступа: по подписке. – Библиогр. в кн. – Текст : электронный.

3. Пирская, Л. В. Разработка мобильных приложений в среде AndroidStudio : учебное пособие : [16+] / Л. В. Пирская ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2019. – 125 с. : ил., табл. – URL: <https://biblioclub.ru/index.php?page=book&id=598634> (дата обращения: 12.07.2021). – Режим доступа: по подписке. – Библиогр. в кн. – ISBN 978-5-9275-3346-6. – Текст : электронный.

4. Коматинени, С., Маклин, Дэйв. Android 4 для профессионалов / С. Коматинени, Д. Маклин. Создание приложений для планшетных компьютеров и смартфонов.: пер. с англ. - М. : ООО “И.Д. Вильямс”, 2012. - 880 с.

5. Котляров, В. П. Основы разработки приложений беспроводных устройств : учебное пособие / В. П. Котляров, Н. В. Воинов. - 2-е изд., исправ. - Москва: Открытый Университет «ИНТУИТ», 2016. - 328 с. : ил. - URL: <https://biblioclub.ru/index.php?page=book&id=429085> (дата обращения: 12.07.2021). - Режим доступа: по подписке. - Библиогр. в кн. - Текст: электронный.

6. Долженко, А. И. Разработка приложений на базе WPF и Silverlight : учебное пособие/ А. И. Долженко. - 2-е изд., исправ. - Москва: Открытый Университет «ИНТУИТ», 2016. - 453 с. : ил. - URL: <https://biblioclub.ru/index.php?page=book&id=428797> (дата обращения: 12.07.2021). - Режим доступа: по подписке. - Библиогр. в кн. - Текст: электронный.