МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»



**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

по выполнению лабораторных работ

по дисциплине «Программирование мобильных устройств»

для студентов направлений 09.03.03 «Прикладная информатика» и 09.03.02 «Информационные системы и технологии».

Ставрополь

2021

# **Лабораторная работа 12. Виды представлений.**

**Цель работы:** Научиться добавлять различные виды активностей.

**Формируемые компетенции:** ПК-7

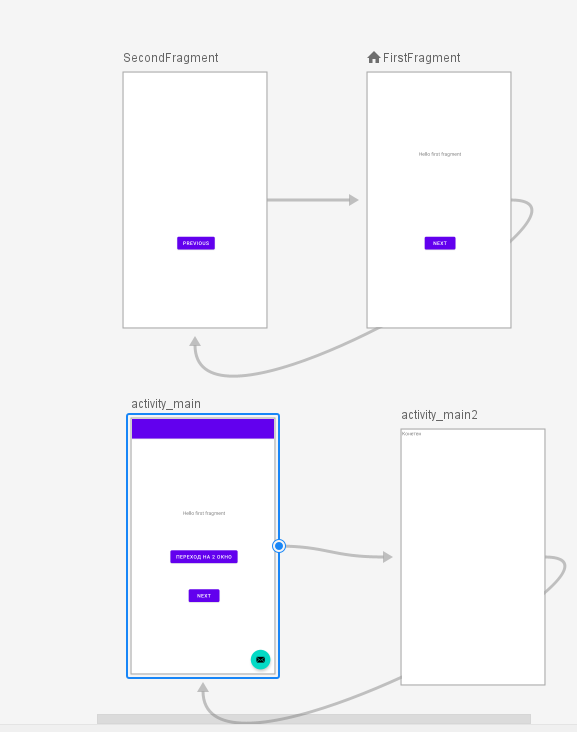
**Теоретическая часть:**

При создании новых активностей в мобильном приложении есть несколько вариантов выбора. До этого мы создавали пустые активности, на которых был только один элемент TextView. Если нам требовалось что-то ещё, то мы помещали эти компоненты в разметку вручную. Тем не менее, Android Studio поддерживает возможность использовать несколько шаблонов активностей, которые ускоряют процесс разработки. Рассмотрим каждый из них.

1. Empty Activity – простая пустая активность, которую мы создавали всегда.
2. Basic Activity. Базовая активность, которая содержит несколько компонентов:
   1. Частичное представление content\_main. Этот компонент добавлен с помощью тэга <include>. Про него мы говорили в 5 лабораторной работе.
   2. Компонент AppBarLayout. Верхняя панель приложения в большинстве случаев отвечает за навигацию. В ней находится название окна, где сейчас находится пользователь, кнопка меню и кнопка «назад», если с него можно вернуться на предыдущее окно. Про само меню и его пункты мы уже говорили в лабораторной работе 9.
   3. Кнопка перехода на второе окно.
   4. Плавающая кнопка, нажатие на которую происходит некоторое действие, которое мы захотим.

В данной активности присутствует ещё один компонент, который называется «Фрагмент», Фрагмент можно тоже считать частичным представлением. Однако, у него есть свои особенности. Безусловно, у него тоже есть возможность взаимодействовать с другими объектами и активностями. Но, главная его задача заключается в том, чтобы обеспечить более рациональное распределение компонентов в активности на разных экранах. Например, в портретной ориентации список и TextView из лабораторной работы 7 смотрелись удобно один под другим. Однако, если перевернуть телефон, то справа от списка будет огромная пустота. Чтобы это компенсировать, фрагменты можно располагать как угодно на странице в зависимости от положения телефона. Неплохой пример Вы можете рассмотреть по [ссылке](http://developer.alexanderklimov.ru/android/theory/fragments.php).

Также, в данной активности добавили такую вещь, как Navigation Architecture Component. Он позволяет по-другому осуществить переходы между активностями. Более того, этот компонент позволяет визуализировать переходы с помощью специального графа:



Вы можете использовать его, а можете и нет. На процесс работы приложения это никак не влияет. Однако, по мере разрастания Вашего проекта и увеличения числа активностей, можно запутаться. Данный компонент позволяет визуально рассмотреть, какие зависимости есть между компонентами: из какой активности/фрагмента можно перейти в другое место.

1. Bottom Navigation Activity. Очень удобная активность. В неё навигационная панель уже располагается внизу. Каждый из элементов меню – это отдельный фрагмент, который можно увидеть, выбрав определённый пункт из панели. Безусловно, вместо фрагментов можно создать свои активности (в таком случае, правильным подходом будет использование такого же шаблона для каждой из них).
2. FullscreenActivity – активность, которая позволяет отобразить содержимое в полном окне. При таком подходе контент будет растянут на весь экран, однако, при любом прикосновении можно увидеть снова верхнюю панель приложения и строку состояния. Все настройки, связанные с анимацией полного экрана можно указать в файле .java. На самом деле любую активность можно сделать полноэкранной. Но здесь все настройки уже указаны и нам остаётся только подстроить их под себя.
3. Login Activity. Данная активность при создании совмещает в себе сразу несколько основных компонентов, отвечающих за реализацию страницы с логином/регистрацией. Он включает в себя:
   1. ViewModel – модель, имеющая основные свойства, с которыми мы будем работать. Например, LoginRepository (хранит список всех логинов в приложении), LoginResult (возвращает информацию, успешно или нет пользователь ввёл данные), LoginFormState (хранит информацию об ошибках пользователя, например, неправильная пара логин/пароль).
   2. Слушатели на ввод данных через EditText (чтобы подсвечивать ошибки пользователю).
   3. Отображение Toast об успешной/безуспешной авторизации.
   4. Кнопка логин, запускающая процесс поиска/регистрации пользователя.
4. Navigation Drawer Activity. Одна из самых распространённых активностей. Данная активность включает в себя компонент навигации, то есть боковое меню. NavigationView – тот самый компонент. Скопировав его и поместив его на любую свою страницу Вы сможете открыть его смахнув слева. Обработку выбора пунктов меню Вы уже делали на лабораторной 9.
5. ScrollingActivity – активность, которая поддерживает отображение большого содержимого. Любой компонент в xml можно заставить прокручиваться. Но сделать это можно только поместив его в контейнер ScrollView. Scrolling Activity же позволяет сразу всё содержимое страницы заставлять прокручиваться. Причём шапка активности будет анимированно прятаться.
6. Settings Activity. Когда для Вашего приложения предполагаются некоторые настройки, который пользователь выбирает для себя, их выносят в отдельную активность. Самым простым вариантом создания такой активности это Setting Activity. Изменив содержимое на те компоненты, которые будут использоваться у Вас, можно за несколько минут оформить вполне приличную страницу с настройками.

В связи с существованием такой страницы рассмотрим ещё кое-что. Все настройки, сохранённые пользователем должны быть спрятаны где-то в телефоне так, чтобы при следующем открытии можно было их получить. По умолчанию, при открытии приложения заново всё, что не записалось в БД или файлы (Кэш) автоматически очищается телефоном. Здесь мы не можем допустить подобное. Для решения данной задачи используется встроенный компонент, называемый SharedPreferences.

Данный класс позволяет хранить данные следующих типов:

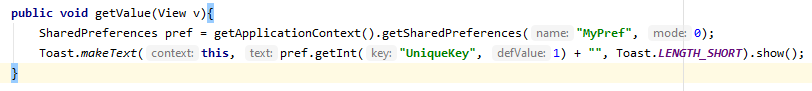
* Boolean (true/false);
* Float (Число с плавающей точкой);
* Int (Целое число);
* Long(Целое число, но в два раза больше Int)
* String (Строка).

Для записи данных используются методы, начинающиеся с put (как в SQLite), например, putBoolean(“ключ”, false). В качестве ключа должно использоваться уникальное слово/фраза. Ключи не должны повторяться. Чтобы получить значение, можно воспользоваться методом, начинающимся с get (например, getBoolean(“ключ”, false), где вторым параметром передаётся значение по умолчанию. Значение по умолчанию позволяет гарантировать, что, если такого ключа нет в настройках, то мы хоть что-то, но получим).

Например, на странице настроек добавим следующий код:



А в MainActivity:



Добавив кнопку на главной странице и навесив на неё это событие, мы сможем увидеть, как значение сохраняется:



Если закрыть и открыть приложение заново, то данные сохранятся.

1. Tabbed Activity. По своему внешнему виду напоминает Bottom Navigation Activity. Отличается тем, что содержимое вкладок может быть только разметка или фрагмент, а в BTN – это отдельные активности.
2. Google AdMob Activity – активность для настройки рекламного отображения. Не будет использовать в данном курсе.
3. Google Maps Activity – активность для отображения содержимого из Google Maps.

В лабораторной работе 5 мы узнали, какими способами можно добавлять новые активности. Естественно, с каждой из них работает такой же подход. Шаблон лишь позволяет ускорить этот процесс, создавая все необходимые файлы.

**Задания:**

1. Создать приложение с минимум 3 видами окон.
2. Все последующие лабораторные работы будут нацелены на создание одного полноценного игрового приложения.
3. В соответствии с выбранным вариантом создайте шаблоны активностей, и настройте переходы между ними.

**Варианты для игры:**

1. Крестики нолики:
   1. Главный экран;
   2. Экран настроек;
   3. Экран игры.
2. Реверси
   1. Главный экран;
   2. Экран игры;
   3. Экран добавления игроков;
   4. Экран с победителями.
3. Шашки
   1. Главный экран;
   2. Экран игры;
   3. Экран добавления игроков;
   4. Экран с победителями.
4. Таблички
5. Морской бой
   1. Главный экран;
   2. Экран добавления кораблей;
   3. Экран игры;
   4. Экран добавления игроков;
   5. Экран с победителями.
6. Гонки
   1. Главный экран;
   2. Экран игры;
   3. Экран с рекордами.
7. Поймай круг
   1. Главный экран;
   2. Экран игры;
   3. Экран с рекордами.
8. Угадай карту
   1. Главный экран;
   2. Экран игры;
   3. Экран с рекордами.
9. Больше-меньше
   1. Главный экран;
   2. Экран игры;
   3. Экран с рекордами.
10. Сапёр
    1. Главный экран;
    2. Экран игры;
    3. Экран с настройками;
    4. Экран с рекордами.
11. Волк ловит яйца
    1. Главный экран;
    2. Экран игры;
    3. Экран с настройками;
    4. Экран с рекордами.
12. Плитки фортепиано
    1. Главный экран;
    2. Экран игры;
    3. Экран с настройками;
    4. Экран с рекордами.
13. Судоку
    1. Главный экран;
    2. Экран игры;
    3. Экран с настройками;
    4. Экран с рекордами.
14. 10 отличий.
    1. Главный экран;
    2. Экран игры;
    3. Экран с рекордами.
15. Пятнашки
    1. Главный экран;
    2. Экран игры;
    3. Экран с настройками;
    4. Экран с рекордами.
16. Змейка
    1. Главный экран;
    2. Экран игры;
    3. Экран с настройками;
    4. Экран с рекордами.

**Оборудование и материалы:** для выполнения данной лабораторной работы необходим компьютер с установленной операционной системой Windows 8 (8.1, 10) и программным продуктом: AndroidStudio.

**Указания по технике безопасности:** к выполнению лабораторных работ допускаются студенты, ознакомившиеся с правилами работы в лаборатории, прошедшие инструктаж безопасности.

**Содержание отчета:** отчет по лабораторной работе должен быть выполнен в редакторе MSWord и оформлен согласно требованиям. Требования по форматированию: Шрифт TimesNewRoman, интервал – полуторный, поля левое – 3 см., правое – 1,5 см., верхнее и нижнее – 2 см. Абзацный отступ – 1,25. Текст должен быть выровнен по ширине.

Отчет должен содержать титульный лист с темой лабораторной работы, цель работы и описанный процесс выполнения вашей работы. В конце отчета приводятся выводы о проделанной работе.

В отчет необходимо вставлять скриншоты выполненной работы и добавлять описание к ним. Каждый рисунок должен располагаться по центру страницы, иметь подпись (Рисунок 1 – Создание подсистемы) и ссылку на него в тексте.

**Контрольные вопросы:**

1. Окна

**Список литературы, рекомендуемый к использованию по данной теме:**

1. Пархимович М. Н. , Липницкий А. А. , Некрасова В. А. Основы интернет-технологий: учебное пособие / Пархимович М. Н. , Липницкий А. А. , Некрасова В. А. – М.: Архангельск: ИПЦ САФУ, 2013 – 366 с.; То же [Электронный ресурс]. - URL:http://biblioclub.ru/index.php?page=book\_red&id=436379&sr=1
2. Соколова В. В. Разработка мобильных приложений: учебное пособие/ Соколова В. В. Томск: Издательство Томского политехнического университета, 2015 – 176с. То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book\_red&id=442808&sr=1