

## ОСНОВЫ СОЗДАНИЯ КЛИЕНТ–СЕРВЕРНОГО ПРИЛОЖЕНИЯ ДЛЯ ПЛАТФОРМЫ ANDROID

*А.С. Кулешов, магистрант*

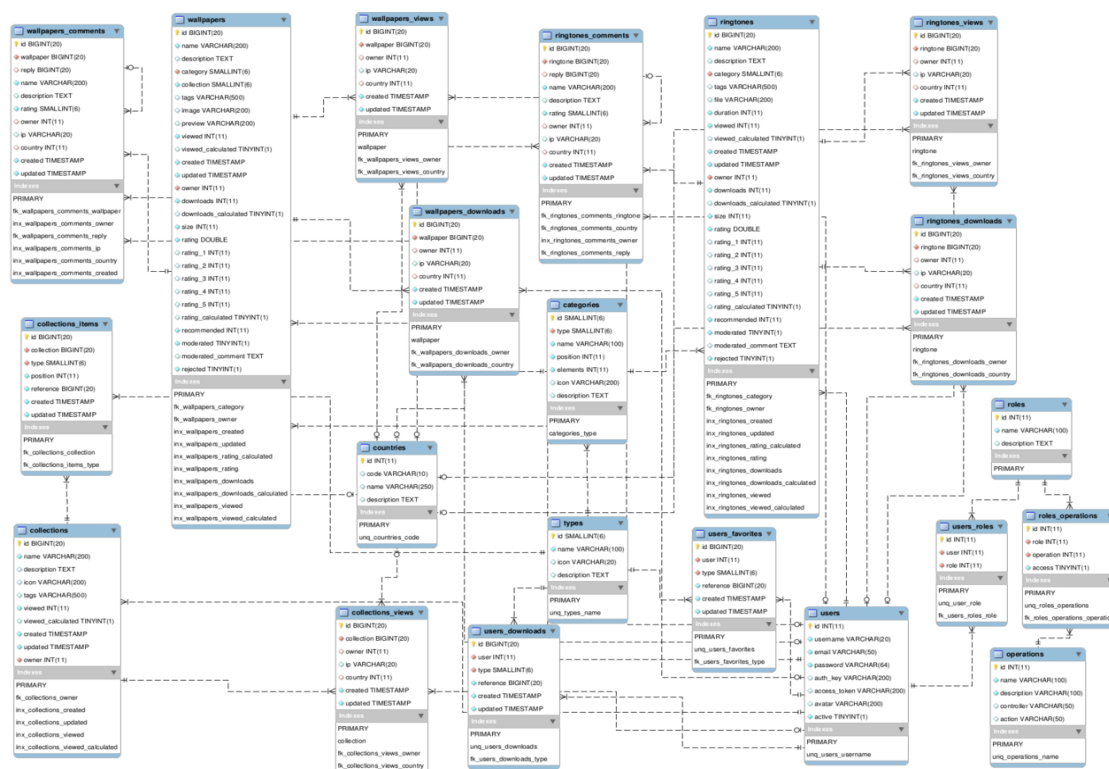
*Научный руководитель – М.И. Жадан, к.ф.–м.н., доцент*

*Гомельский государственный университет им. Ф. Скорины*

В настоящей работе речь пойдёт о связи сервера с клиентским приложением. Для этих целей было создано приложение для телефонов под операционную систему Android. Приложение представляет собой программу–контейнер с контентом различного вида: обои для рабочего стола и телефонных контактов, рингтоны, игры и другое.

Со стороны сервера приложение имеет веб-сайт, на который пользователи могут загружать соответствующий контент в свои аккаунты. Обмен данными между клиентом и сервером основан на работе с json-файлами. В клиентском приложении за работу с json отвечает компонент AQuery. Со стороны сервера придёт соответствующий запросу ответ.

При разработке серверной части также необходимо было создать базу данных, где хранились бы нужные нам данные о контенте и пользователях. Структура базы представлена на рисунке 1.

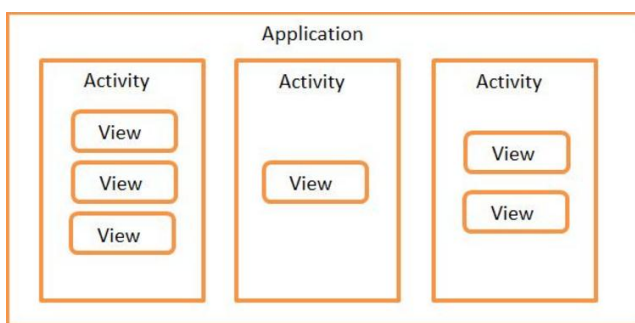


**Рисунок 1 – Структура бд серверной части приложения**

При разработке дизайна для клиентского приложения было решено остановиться на современном дизайне от компании Google – Material Design. Исходя из этого и графические компоненты, используемые в программе, необходимо использовать соответствующие.

Рассмотрим внутреннюю структуру большинства приложений под Android. Если проводить аналогию с Windows, то приложение состоит из окон, называемых Activity. В конкретный момент времени обычно отображается одно Activity и занимает весь экран, а приложение переключается между ними. В качестве примера можно рассмотреть почтовое приложение. В нем одно Activity – список писем, другое – просмотр письма, третье – настройки ящика. При работе вы перемещаетесь по ним.

Содержимое Activity формируется из различных компонентов, называемых View. Самые распространенные View – это кнопка, поле ввода, чекбокс и т.д. Примерно это можно изобразить так, как на рисунке 2.



**Рисунок 2 – Структура любого android-приложения**

Необходимо заметить, что View обычно размещаются в ViewGroup. Самый распространенный пример ViewGroup – это Layout. Layout бывает различных типов и отвечает за то, как будут расположены его дочерние View на экране (таблицей, строкой, столбцом и т.д.).

Представление Activity перед пользователем осуществляется с помощью xml-файла, называемого layout. В нем мы определяем набор и расположение элементов View, которые хотим видеть на экране. При запуске приложения, Activity читает этот файл и отображает нам то, что мы настроили. Пример того, как выглядит графическая часть layout-файла в Android Studio можно увидеть на рисунке 3. В дальнейшем на этом layout-файле и размещаются все графические компоненты.

Разработанное приложение состоит из нескольких Activity, которые включают в себя такие компоненты как Fragments.

Начинается приложение с запуска Activity, на котором идёт инициализация и загрузка всех необходимых для работы данных. Обмен данными с сервером осуществляется через компонент AQuery. Также на этом Activity инициализируются sdk рекламных блоков, блоков сбора статистики и осуществляется проверка на наличие интернет-соединения.

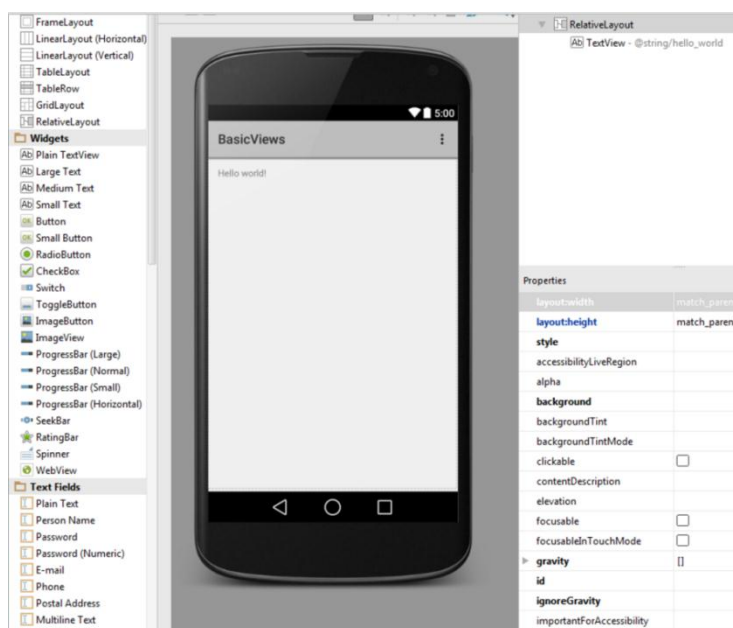


Рисунок 3 – Отображения layout-файла

После этого идёт запуск следующего Activity. Само Activity представляет собой галерею изображений, представленных различным образом.

Структура данного Activity представляет собой Activity, которое наследуется от AppCompatActivity и включает в себя 5 фрагментов. Работа с фрагментами осуществляется с помощью адаптера TabFragmentPagerAdapter. Данные во фрагменты передаются с помощью стандартного конструктора фрагментов и позже обрабатываются непосредственно в выбранном фрагменте. Данные во фрагментах обрабатываются и заносятся в компонент RecyclerView.

Все картинки выводятся на экран не сразу, чтобы не выгрузить всю память телефона, а с помощью подгрузки. Подгрузка данных в RecyclerView осуществляется с помощью переопределения метода onScroll() во фрагменте. При этом посылается запрос на сервер и получается ответ в виде json с информацией о выводимых изображениях.

После нажатия на выбранное изображение считывается id изображения и открывается следующее Activity. При этом посылается запрос на сервер и получаются данные о конкретном изображении. Навигация по приложению осуществляется с помощью NavigationView и TabLayout. Для удобства пользователя было выбранное боковое всплывающее меню, которое позволяет значительно сэкономить место и удачно вписывается в дизайн приложения.

Пока в рабочую версию приложения добавлены только часть Wallpapers.

Приложение было успешно залито на play.google.com и имеет уже более 100000 скачиваний.