**Инструменты использующиеся в разработке Mobile модуля приложения**

Android <https://ru.wikipedia.org/wiki/Android>

Android Подготовка к разработке <http://developer.alexanderklimov.ru/android/android1.php>

Используем Retrofit 2 в Android-приложении <https://habr.com/ru/post/429058/>

Работа с базами данных SQLite <https://metanit.com/java/android/14.5.php>

SQLite <http://developer.alexanderklimov.ru/android/sqlite/>

Mobile часть приложения была сделана под операционную систему Android на языке Java. Для работы с сетью использовалась, довольно популярная среди Android-разработчиков, библиотека Retrofit. Mobile приложение не будет хранить все данные на сервере. В режиме offline данные будут хранится в компактной встраиваемой СУБД SQLite.

**Android**

Android — операционная система для смартфонов, планшетов, электронных книг, цифровых проигрывателей, наручных часов, фитнес-браслетов, игровых приставок, ноутбуков, нетбуков, смартбуков, телевизоров и других устройств (в 2015 году появилась поддержка автомобильных развлекательных систем и бытовых роботов).

Основана на ядре Linux и собственной реализации виртуальной машины Java от Google. Изначально разрабатывалась компанией Android Inc., которую затем купила Google. Впоследствии Google инициировала создание альянса Open Handset Alliance (OHA), который сейчас занимается поддержкой и дальнейшим развитием платформы. Android позволяет создавать Java-приложения, управляющие устройством через разработанные Google библиотеки. Android Native Development Kit позволяет портировать библиотеки и компоненты приложений, написанные на Си и других языках.

Android стала самой революционной операционной системой благодаря разработке по революционной методологии SCRUM. В 86 % смартфонов, проданных во втором квартале 2014 года, была установлена операционная система Android. На конференции для разработчиков в мае 2017 года Google объявила, что за всю историю Android было активировано более 2 млрд Android-устройств.

Важно понимать, что само приложение для Android пишется на Java (а теперь ещё и на Kotlin), а среда разработки выбирается по вкусу. То, что можно сделать на Android Studio (или в его старшем брате IntelliJ IDEA), можно сделать и на Eclipse и в других редакторах. Хотя с 2016 официальная поддержка Eclipse прекратилась.

**Retrofit**

Retrofit — это известная среди Android-разработчиков библиотека для сетевого взаимодействия, некоторые даже считают её в каком-то роде стандартом. Причин для такой популярности масса: библиотека отлично поддерживает REST API, легко тестируется и настраивается, а запросы по сети с её помощью выполняются совсем просто.

Для работы с Retrofit необходимы следующие три класса:

* Model класс, который используется как модель JSON
* Интерфейсы, которые определяют возможные HTTP операции
* Класс Retrofit.Builder — экземпляр, который использует интерфейс и API Builder, чтобы задать определение конечной точки URL для операций HTTP

Каждый метод интерфейса представляет собой один из возможных вызовов API. Он должен иметь HTTP аннотацию (GET, POST и т. д.), чтобы указать тип запроса и относительный URL. Возвращаемое значение завершает ответ в Call-объекте с типом ожидаемого результата.

Retrofit позволяет сделать полноценный REST-клиент, который может выполнять POST, GET, PUT, DELETE. Для обозначения типа и других аспектов запроса используются аннотации. Например, для того, чтобы обозначить, что требуется @GET запрос, нам нужно написать перед методом GET, для @POST запроса POST, и так далее.

**SQLite**

Платформа Android имеет встроенный инструментарий для управления базой данных sqlite3. SQLite - это замечательный продукт, который успел завоевать признание во всем мире и получить множество наград. Интересно отметить, что iOS (iPhone, iPod, iPad) также используют базы данных SQLite. А недавно и Microsoft присоединилась к данному решению и телефоны Windows Phone 8 также работают с SQLite.

SQLite - это проект с открытыми исходными кодами, поддерживающий стандартные возможности обычной SQL: синтаксис, транзакции и др. Занимает очень мало места - около 250 кб. Домашняя страница SQLite: http://www.sqlite.org.

Основную функциональность по работе с базами данных предоставляет пакет android.database. Функциональность непосредственно для работы с SQLite находится в пакете android.database.sqlite.

База данных в SQLite представлена классом android.database.sqlite.SQLiteDatabase. Он позволяет выполнять запросы к бд, выполнять с ней различные манипуляции.

Класс android.database.sqlite.SQLiteCursor предоставляет запрос и позволяет возвращать набор строк, которые соответствуют этому запросу.

Класс android.database.sqlite.SQLiteQueryBuilder позволяет создавать SQL-запросы.

Сами sql-выражения представлены классом android.database.sqlite.SQLiteStatement, которые позволяют с помощью плейсхолдеров вставлять в выражения динамические данные.

Класс android.database.sqlite.SQLiteOpenHelper позволяет создать базу данных со всеми таблицами, если их еще не существует.

В SQLite применяется следующая система типов данных:

* INTEGER: представляет целое число, аналог типу int в java
* REAL: представляет число с плавающей точкой, аналог float и double в java
* TEXT: представляет набор символов, аналог String и char в java
* BLOB: представляет массив бинарных данных, например, изображение, аналог типу int в java

Сохраняемые данные должны представлять соответствующие типы в java.