

АНАЛИЗ ЯЗЫКОВЫХ ИНСТРУМЕНТОВ ОС ANDROID ДЛЯ РЕАЛИЗАЦИИ ВЗАИМОДЕЙСТВИЯ ПЕРСОНАЛА И АДМИНИСТРАЦИИ С ПОМОЩЬЮ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ.

Зимцев П.П., Чередникова О.Ю.

Донецкий Национальный Технический Университет
кафедра компьютерных наук и технологий
E-mail:zimtsev.p@yandex.ua

Аннотация:

Зимцев П.П., Чередникова О.Ю. Анализ языковых инструментов ОС Android для реализации взаимодействия персонала и администрации с помощью современных технологий. Исследованы возможности организации взаимодействия персонала и администрации. Проанализированы разные языковые инструменты для программирования под Android. Рассмотрены преимущества, недостатки и актуальность языковых инструментов.

Annotation:

Zimtsev PP, Cherednikova O.Yu. Analysis of the language tools of the Android OS for the implementation of staff and administration interaction using modern technologies. The possibilities of organization of interaction between staff and administration are analyzed. Various language tools for programming under Android are analyzed. The advantages, disadvantages and urgency of language tools are considered.

Введение

В современном мире ценность своевременной коммуникации в рабочих условиях очень важна. Есть разные способы коммуникации, как пример можно привести телефонную связь, однако с увеличением доступности качественного и быстрого интернет соединения популярность стремительно набрали мессенджеры. Именно они дают возможность, используя интернет соединение, не только общаться, а и передавать необходимые данные и сразу предоставлять их для персональной обработки человеком.

Постановка задачи

Современные мобильные телефоны имеют огромный функционал и большие возможности, с помощью которых можно получать и обрабатывать информацию своевременно и устраняет необходимость использования компьютера. Так как современные мобильные устройства используют две операционные системы – это ОС Android и iOS, но ОС Android является более распространенной системой, то одним из требований является использование языковых инструментов именно для ОС Android. Следствием этого стала идея создания современного способа взаимодействия рабочего персонала на основе современных технологий. А именно взаимодействие рабочего персонала и администрации путем коммуникации через мессенджер. Однако для распространения и простоты использования необходимо чтобы данный способ информационной системы отвечал следующим главным требованиям:

1. Информационная система должна быть доступна для популярной ОС, которая используется мобильными устройствами.

2. Платформа для реализации информационной системы должна быть надежной и ее выбор должен быть оправдан.
3. Интерфейс доступен и прост для любого пользователя.

Реализация взаимодействия персонала и администрации

Часто случаются ситуации, когда начальству необходимо передать информацию сотрудникам не в рабочее время. Такую задачу можно решить или используя мобильную связь, или же связь по интернету. Поэтому для реализации взаимодействия между персоналом и администрацией выбран способ взаимодействия путем коммуникации через мессенджер. Прежде чем начать разработку необходимо проанализировать языковые инструменты для разработки, их сетевые возможности и особенность реализации. Для реализации непосредственного взаимодействия персонала и администрации путем коммуникации через мессенджер необходимо разработать соответствующие базы данных для успешного ведения общения через текстовые сообщения. Необходимо не менее двух баз данных, которые будут впоследствии расположены на сервере и на мобильном устройстве пользователя, а так же технологию их взаимодействия, которой может являться WebSocket или Long Polling.

Анализ языковых инструментов для реализации поставленной задачи

Для реализации поставленной задачи был проанализирован ряд языковых инструментов, которые позволят успешно реализовать поставленную задачу, а так же смогли бы выполнить все требования данной задачи. В данном случае представлены 3 языковых инструмента-это Java, Kotlin и Scala.

Java

Одной из популярных платформ на данный момент остается одна из основных платформ – это Java. Данная платформа является основной платформой для разработки Android приложений. Является слишком типизированным языком программирования, принадлежит компании Oracle. Имеет собственный байт-код Java, в который транслируются программы и в последствии выполняются виртуальной машиной Java.

Виртуальная машина предполагает собой программу, которая обрабатывает байтовый код и передает инструкции оборудованию напоминая интерпретатор. Основным достоинством такого варианта исполнения программ выражается в независимости байт-кода от ОС и оборудования, а это дает возможность независимости от оборудования, которое располагает собственной виртуальной машиной. Гибкая система безопасности является также важным достоинством, на которое необходимо обращать особое внимание. В рамках этой системы безопасности контроль над исполнением программы находится у виртуальной машины. Все операции, которые превышают установленные полномочия вызывают немедленное прерывание.

Часто к недостаткам концепции виртуальной машины относят снижение производительности. Ряд усовершенствований несколько увеличил скорость выполнения программ на Java:

- применение технологии трансляции байт-кода в машинный код непосредственно во время работы программы (JIT-технология) с возможностью сохранения версий класса в машинном коде,
- широкое использование платформенно-ориентированного кода (native-код) в стандартных библиотеках,
- аппаратные средства, обеспечивающие ускоренную обработку байт-кода.

К недостаткам Java можно отнести нестабильность и большое потребление памяти. Этот вывод можно сделать опираясь на исследования компании Google, по результатам которого был сделан вывод что тестовый пример выявил существенную низкую производительность Java и большие затраты памяти в отличии от аналогичного примера, который был реализован на платформе C++. В свою очередь данные портала shootout.alioth.debian.org, пользователями которого был также проведен эксперимент, так же могут подтвердить данный недостаток.

Основные возможности:

- автоматическое управление памятью;
- расширенные возможности обработки исключительных ситуаций;
- богатый набор средств фильтрации ввода-вывода;
- набор стандартных коллекций: массив, список, стек и т.п.;
- наличие простых средств создания сетевых приложений (в том числе с использованием протокола [RMI](#));
- наличие классов, позволяющих выполнять HTTP-запросы и обрабатывать ответы;
- встроенные в язык средства создания многопоточных приложений, которые потом были портированы на многие языки (например, python);
- унифицированный доступ к базам данных:
 - на уровне отдельных SQL-запросов — на основе JDBC, SQLJ;
 - на уровне концепции объектов, обладающих способностью к хранению в базе данных — на основе [Java Data Objects](#) и Java Persistence API;
- поддержка обобщений (начиная с версии 1.5);
- поддержка лямбд, замыканий, встроенные возможности функционального программирования (с 1.8);
- множество вариантов реализации многопоточных программ.

Для сетевой работы в ОС Android присутствуют разные пакеты, которые реализуют разные классы, ниже представлены некоторые из них:

- `android.app.admin`. Предоставляет возможность управления устройством со стороны персонала, такого как администраторы предприятия.
- `android.accounts`. Предоставляет классы для управления учетными записями, такими как Google, Facebook и т.п. Основными классами являются `AccountManager` и `Account`. Этот API-интерфейс кратко рассматривается в главе 30 при обсуждении `Contacts API`.
- `android.net`. Реализует базовые сетевые API-интерфейсы уровня сокетов. Основные классы — `Uri`, `ConnectivityManager`, `LocalSocket` и `LocalServerSocket`. Здесь стоит упомянуть, что Android поддерживает работу с HTTPS не только на уровне браузера, но и на сетевом уровне. ОС Android также поддерживает JavaScript в своем браузере.

Доступ к базам данных реализуется с помощью JDBC. Платформенно-независимый промышленный стандарт взаимодействия Java-приложений с различными СУБД, реализованный в виде пакета `java.sql`, входящего в состав Java SE. JDBC основан на концепции так называемых драйверов, позволяющих получать соединение с базой данных по специально описанному URL. Драйверы могут загружаться динамически (во время работы

программы). Загрузившись, драйвер сам регистрирует себя и вызывается автоматически, когда программа требует URL, содержащий протокол, за который драйвер отвечает.

Kotlin

Одной из современных платформ по разработке продукции для Android является язык программирования Kotlin (Kotlin). Данный язык был разработан группой разработчиков из российского отделения компании JetBrains. Язык позиционируется разработчиками как платформа, которая сможет заменить Java. Полностью совместим и дает возможность разработчикам плавного перехода с платформы Java. С мая 2017 года является официальным инструментом разработки для ОС Android и включен Android Studio 3.0. Ниже рассмотрим основные преимущества и недостатки данной платформы.

Преимущества:

1. Объем кода. Неиспользование ненужных архаичных кусков кода ускоряет процесс разработки и повышает читаемость;
2. Безопасность. Заложенные в синтаксис языка правила создания кода позволяют избежать мелких трудно отслеживаемых ошибок, которые возникают лишь в момент исполнения программы;
3. Совместимость в Java. Это удобно и с точки зрения обучения новому языку, и с точки зрения постепенного перевода ваших программных продуктов с Java на Kotlin. К примеру, именно так произошло с приложением Basecamp.

Недостатки:

Как у языка, который был разработан на основе Java, у Kotlin по сравнению с ним есть два очевидных недостатка: скорость и проработанность. В первом случае все понятно: работая на JVM трудно обойти по быстродействию Java. Вторая проблема немного надуманная для столь молодого языка, ведь в официальном обороте он существует чуть больше года. Да, здесь существенно меньше инструментов и библиотек, чем на Java. Но, во-первых, пока это не критично, а во-вторых, в мире Java количество далеко не везде перешло в качество.

Kotlin же пока со своими главными задачами справляется, а полная совместимость с Java помогает закрыть глаза на указанные пробелы. Для сетевого взаимодействия приложения можно использовать библиотеки Java, так как в силу того что язык имел недавний релиз, а последней версией является версия 1.2, то не все возможности на данный момент реализованы. Решением этой проблемы являются пакеты Java, которые можно использовать в Kotlin.

Scala

Язык Scala часто называют «мультипарадигмальным», поскольку в нем комбинируются элементы, присущие различным направлениям философии программирования. В сущности, это чистый объектно-ориентированный язык с функциональными возможностями. Но его серьезное преимущество над традиционными подходами к программированию заключается в том, что в Scala широко поддерживается параллельное программирование. Кроме того, Scala обладает еще одним уникальным компонентом, отсутствующим во всех других языках, применяемых с JVM – так называемыми «акторами». Это еще один конструкт, предназначенный для параллельного программирования и встроенный в язык. Язык компилируется в оптимизированный байт-код, работающий, в сущности, так же быстро, как и нативный код Java. Однако Scala не получает большого распространения, а официальная поддержка языка Kotlin делает его более привлекательным при выборе инструмента для программирования. Еще одним, но

существенным недостатком является компиляция Scala в байт-код, который отличается от байт-кода Java. Эта особенность негативно влияет на разработку, так как для программирования на этом языке необходима тонкая настройка среды разработки, что усложняет саму разработку.

Вывод

Проанализировав всю вышеперечисленную информацию, а также изучив различные материалы можно сделать следующий вывод по данной теме. Коммуникация – это важная часть жизни каждого человека. Развитие технологий позволяет людям своевременно обмениваться информацией находясь на больших расстояниях. Используя новые технологии в разных способах взаимодействия рабочего коллектива, а в нашем случае это исследование способов взаимодействия персонала и администрации с помощью современных технологий, дает возможность повышать информированность и продуктивность коллектива, а также оперативно решать возникающие сложности. Также необходимо выделить важность новой платформы. Совместимость Kotlin с предыдущим базовым языком, которым по традиции является Java, и решения недостатков, а так же улучшения особенностей языка предыдущего поколения делают серьезное преимущество перед выбором по вопросу использования платформы. Таким образом, исходя из анализа приведенных выше языковых инструментов, в качестве средства разработки мессенджера для реализации взаимодействия персонала и администрации более подходящим языковым инструментом является Kotlin, который является современным языковым решением и обладает большими особенностями и преимуществами над базовым языком разработки под ОС Android, которым является Java.

Литература

1. Климов А. Kotlin [Electronic resource] / Интернет-ресурс. - Режим доступа : [http:// http://developer.alexanderklimov.ru/android/kotlin/](http://http://developer.alexanderklimov.ru/android/kotlin/)
2. Гринштейн С. Какое место занимает язык Scala в ИТ-индустрии [Electronic resource] / Интернет-ресурс. - Режим доступа: [http/ URL: http:// https://habr.com/post/314308/](http://https://habr.com/post/314308/)
3. Internet. 9 лучших языков программирования для JVM [Electronic resource] / Интернет-ресурс. - Режим доступа: [http/ URL: https://dev.by/lenta/main/9-luchshih-yazykov-programirovaniya-dlya-jvm](http://https://dev.by/lenta/main/9-luchshih-yazykov-programirovaniya-dlya-jvm)
4. EverydayTools. 25 Android-библиотек, которыми определенно стоит воспользоваться в начале 2017 года / Интернет-ресурс. - Режим доступа: <https://habr.com/company/everydaytools/blog/322390/>
5. Maksym. Scala хуже, чем Java. Как минимум, для половины Java проектов [Electronic resource] / Интернет-ресурс. - Режим доступа: [http/ URL: https://habr.com/post/134897/](http://https://habr.com/post/134897/)
6. Шалыто А. За что Kotlin так полюбили в Google и кому нужны две тысячи языков программирования [Electronic resource] / Интернет-ресурс. - Режим доступа: [http/ URL: http://news.ifmo.ru/ru/science/it/news/6683/](http://http://news.ifmo.ru/ru/science/it/news/6683/)
7. Arturka. Android архитектура клиент-серверного приложения [Electronic resource] / Интернет-ресурс. - Режим доступа: [http/ URL: https://habr.com/company/e- Legion/blog/265405/](http://https://habr.com/company/e- Legion/blog/265405/)
8. Nerumb. Scala vs Kotlin (перевод) [Electronic resource] / Интернет-ресурс. - Режим доступа : [http/ URL: https://habr.com/post/308562/](http://https://habr.com/post/308562/)