



Anteproyecto:

Aplicación android con calculadora.

01/04/2019

Álvaro Vega Díaz

Instituto: IES Francisco de Goya

Módulo: Desarrollo de aplicaciones multiplataforma

ÍNDICE:

Visión general	Pág.1
Qué lenguaje/lenguajes y/o herramientas se van a emplear	Pág.2
¿Por qué usar esta tecnología?	Pág. 2
Objetivos	Pág. 5
Pros y contras de lo que se va a emplear	Pág. 5
¿Cuáles son los puntos positivos?	Pág. 5
¿Cuáles son los puntos negativos?	Pág. 6

1. Visión general

El proyecto en cuestión consiste en la realización de una aplicación Android que, funcionalmente servirá para automatizar o realizar operaciones matemáticas de diversa índole .

Esta aplicación está pensada para un elevado número de usuarios con el fin de facilitarles trabajo en cuanto a cálculos varios, además se contará con un guardado de cálculos frecuentes elegido e introducido por el mismo usuario.

Dicha aplicación consta de una pantalla principal 3 opciones posibles con sus pantallas correspondientes : Calculadora clásica (Que permitirá realizar cálculos básicos), Calculadora científica (Que hará funciones de la clásica con aliciente de mayor configuración en manejo de resultados y cálculos más complejos), Mis Operaciones (El usuario podrá introducir tanto fórmulas como cálculos con datos fijos en dicho apartado). Asimismo el usuario podrá compartir las fórmulas con otros usuarios que podrán incluirlas en su app.

2. Qué lenguaje/lenguajes y/o herramientas se van a emplear

La calculadora se realizará en el lenguaje de programación Kotlin. Kotlin es un lenguaje de programación de tipado estático que corre sobre la máquina virtual de Java y que también puede ser compilado a código fuente de JavaScript. A través de un plug-in del IDE Android Studio.

Para el guardado de fórmulas y operaciones tendrá una base (aunque mínima debido a la aplicación a desarrollar) de SQL en el cual se podrá guardar y actualizar dicha información.

¿Por qué usar esta tecnología?

Se ha decidido emplearla por los siguientes motivos:

1. **El lenguaje y el entorno están maduros:** *a diferencia de otros lenguajes como Swift, el lanzamiento de Kotlin ha pasado por muchas etapas de maduración antes de liberar la versión 1.0 final.*
2. **El plugin del IDE funciona perfectamente:** *y ya permite muchas de las funcionalidades que existen para Java. Por ello, trabajar con Kotlin es un maravilla en ese sentido.*
3. **Es interesante saber que antes de salir a producción:** *el lenguaje pasó varios años en alpha y luego en beta, y que ya antes de que saliera la beta había gente que lo estaba utilizando en proyectos reales.*

4. **Facilita mucho el desarrollo en Android:** *si has estado leyendo todos estos artículos, lo habrás visto claro en muchas ocasiones.*
5. **Kotlin es simplicidad mezclada con potencia:** *por ello es la única alternativa viable a Java para desarrollar grandes aplicaciones en Android. Otras alternativas como Scala son tremendamente pesadas, y los tiempos de compilación y las herramientas no son comparables.*
6. **La compilación en Kotlin a día de hoy ya es equiparable a la de Java:** *(en algunos tests se ha demostrado que incluso superior), y la librería necesaria para usarlo es muy pequeña, por lo que no genera problemas con el límite de métodos en Android.*
7. **Está perfectamente integrado con Android Studio:** *todo funciona satisfactoriamente desde el minuto cero para Android es una ventaja potencial.*
8. **Funciona igual que se estuviera trabajando con Java:** *puedes ejecutar desde el IDE, depurar sin problema, hacer refactors, utilizar instant run... Todo lo que imagines sigue ahí y se sigue pudiendo usar.*
9. **Su evolución está bien cubierta:** *IntelliJ es la empresa detrás de Kotlin, y ella misma lo usa para sus propios productos, así que están más que interesados en seguir desarrollándose.*
10. **Hablando específicamente de Android:** *sus creadores están muy centrados en que todo funcione perfectamente en esta plataforma.*
11. **Múltiples funcionalidades:** *como las Kotlin Android Extensions, hasta librerías como Anko, pasando por las constantes optimizaciones de tamaño y de tiempos de compilación, todo demuestra que están convencidos de que Kotlin es el futuro de Android.*

12. **Hay empresas muy conocidas que ya lo usan en producción:** *por ejemplo, en Pinterest ya lo están utilizando en las nuevas features de su aplicación de Android. Por otro lado, tienes a Basecamp, donde el 70% del código de su App Android ya está escrito en Kotlin.*

13. **Toda la cantidad de código que nos ahorramos en Kotlin revierte en:**

- a. Menos errores: si hay menos código, es más difícil que ese código falle, y por tanto se vuelve más estable.*
- b. Cuando el compilador te ayuda a detectar esos errores en tiempo de compilación, nunca llegan a producirse en tiempo de ejecución (obvio).*
- c. El código es más fácil de entender cuando lo lee otra persona y, nuevamente, es más difícil que se introduzcan errores.*
- d. Más trabajo realizado en menos tiempo, más estable, y con menos necesidad de dedicar tiempo a errores.*

3. Objetivos

- Controlar las nociones de un nuevo lenguaje de programación.
- Explotar las capacidades de Android al máximo con un lenguaje que lo permita.
- Aprender a desarrollar una aplicación real que se adapte a necesidades reales de un grupo o grupos de usuarios.

- Afianzar conceptos de clase:
 - *SQL.*
 - *Manejo de Excepciones.*
 - *Aplicar estándares de limpieza de código.*
 - *Comentar código.*
 - *Principios de usabilidad y consistencia.*

4. Pros y contras de lo que se va a emplear

¿Cuáles son los puntos positivos?

Kotlin es un lenguaje relativamente simple y fácil de asimilar por su familiaridad con Java. El tema relacionado con el SQL supone menor dificultad debido al conocimiento del mismo.

¿Cuáles son los puntos negativos?

Kotlin es un lenguaje nuevo y va a suponer un coste asimilar los conceptos que sean dispares de los lenguajes con los que se está familiarizado. Por otro lado su asociación con SQL puede suponer problemas por desconocimiento de dicho lenguaje.