

Desarrollo de aplicaciones móviles usando el lenguaje Kotlin

Mobile application development using the Kotlin language

Karen Vanessa Corilla Quispe 

20211318@une.edu.pe

Universidad Nacional de Educación Enrique Guzmán y Valle

Resumen

El presente artículo se enfoca en el manejo del lenguaje de programación Kotlin siendo un lenguaje moderno, multiplataforma y puede ser utilizado para diversas aplicaciones móviles del lado del cliente y del servidor basados en la plataforma de Android, por ser el sistema operativo más usado en la actualidad. El contenido está estructurado con las principales aplicaciones que se desarrollan en Kotlin haciendo uso de las diversas herramientas para el diseño con el lenguaje XML. Asimismo, la parte lógica con el lenguaje Kotlin haciendo uso de sus diversas librerías para la implementación de aplicaciones móviles.

Palabras claves: aplicaciones móviles; desarrollo web; lenguaje Kotlin; tecnología móvil.

Abstract

This article focuses on the management of the Kotlin programming language, being a modern, cross-platform language and can be used for various mobile applications on the client and server side based on the Android platform, as it is the most used operating system in the world. present. The content is structured with the main applications that are developed in Kotlin using the various tools for designing with the XML language. Likewise, the logical part with the Kotlin language making use of its various libraries for the implementation of mobile applications.



Esta obra está bajo una licencia Creative Commons Atribución 4.0 Internacional (CC BY 4.0).

Keywords: mobile applications; Web development; Kotlin language; Mobile technology.

Introducción

La frase “App móvil” es un software de aplicación que es creado para dispositivos móviles, como smartphones y Tablets, el mismo que desde hace una década se convirtió popular (Geddes, 2004). En el año 2021 fue listada como “Word of the Year”, como palabra del año por la American Dialect Society (Ceballos, 2014). Con el uso del internet, estas aplicaciones han evolucionado rápidamente hacia un modelo de plataformas semicerradas basado en aplicaciones que gestionan los datos online y prescinden del navegador (Wolff, 2010).

Cinco años después, el uso de aplicaciones ha crecido de manera exponencial, los dispositivos móviles disputan la posición preeminente con los dispositivos de la televisión o PC (Khalaf, 2014; Meeker, 2014). Teniendo diversas aplicaciones móviles en las plataformas como iOS o Android, los mismo que, según Fransman (2014), tienen como objetivos:

- Garantizar el control de un canal de distribución dominante que les permite articular servicios de valor añadido.
- Producir un entorno de innovación emergente basado en los principios de la evolución cantidad, diversidad y selección.

En la educación, el uso de las TIC’S (tecnologías de la información y la comunicación) ha permitido la innovación en la formación de enseñanza y aprendizaje con el manejo de dispositivos móviles (M-Learning) dentro y fuera del aula (Pascuas, García y Mercado, 2020). Este aprendizaje se puede realizar en cualquier momento y lugar, no sólo para la transmisión de conocimientos, sino también, para el desarrollo de otro tipo de estrategias como: aprendizaje autónomo, trabajo colaborativo, aplicaciones para complementar el aprendizaje (Santiago, 2012; Díez, 2012). Para trabajar en grupo o individual, con mayor o menor nivel de profundidad o responsabilidad. Asimismo, permite al profesor evaluar de forma diferente al estudiante, consiguiendo así una valoración mucho más completa y ajustada de sus logros (Gardner, 2000).



Por otro lado, en este texto se menciona al lenguaje de programación Kotlin, siendo un lenguaje moderno, multiplataforma que puede ser utilizado para varios fines, es por esta razón, que los programadores usan para el desarrollo móvil en Android como en iOS; también es conocido por su diseño práctico y su concisa sintaxis, sin olvidar que provee una gran oportunidad para la reutilización de código y compartir entre desarrolladores el uso de múltiples proyectos (Grin, 2019).

En este sentido, el propósito del presente trabajo es dar a conocer lo que Grin y Castillo (2017), dicen del lenguaje Kotlin, como una curva de aprendizaje muy buena, sencilla y fácil de comprender para la implementación de proyectos de forma rápida y eficiente sin tener muchas líneas de código en las plataformas de Android, y ser el más usado actualmente por la población.

Inicios de Kotlin

En el año 2010, la empresa JetBrains desarrollo el lenguaje de Kotlin liderado por Andrey Breslav, siendo publicado en febrero del 2012 como un lenguaje de código abierto para que cualquier desarrollador pueda utilizar (Luján, 2017). El nombre de denominación de este lenguaje se puso así, en referencia a una isla que se encuentra en el Mar Báltico y cerca de la empresa desarrolladora (Smyth, 2017). Inicialmente Kotlin fue creado para aplicaciones Desktop, actualmente Kotlin es multiplataforma con soporte para Android en un 100%.

Aplicaciones de Kotlin

Kotlin, en la máquina virtual de Java (JVM), puede ejecutar diferentes aplicaciones, tales como el desarrollo de aplicaciones web en el lado del cliente y también en el lado del servidor. Asimismo, Kotlin tiene su mayor popularidad en el desarrollo de aplicaciones para Android (Devic, 2018).

Web en el lado del cliente. Con los lenguajes específicos de dominio (DSL) que tiene Kotlin, permite desarrollar las páginas HTML con los estilos CSS, la cual no es indispensable el uso de otro lenguaje para la creación de sitios web dinámicos (Subramaniam, 2019)

Web en el lado del servidor. Kotlin es ideal para desarrollar aplicaciones del lado del servidor, teniendo un código más conciso y compacto para su fácil comprensión. Asimismo, gracias a su librería se puede crear cualquier tipo de módulos



o web services, ofreciendo interoperabilidad, migración y escalabilidad en los diversos sistemas (Jemerov, 2017; Isakov, 2017)

Android. Kotlin tuvo el respaldo del Google por ser un lenguaje moderno con una experiencia productiva y placentera, además oficial para el desarrollo de Android, por esta razón empezó su popularidad y a muchos desarrolladores le gustó trabajar este lenguaje con un resultado de constante actualización que ayuda a las necesidades del desarrollador (Jemerov, 2017; Isakov, 2017).

Desarrollo con Kotlin

Para el desarrollo de aplicaciones en Android haciendo uso del lenguaje Kotlin, se debe tener en cuenta el sistema operativo para móviles:

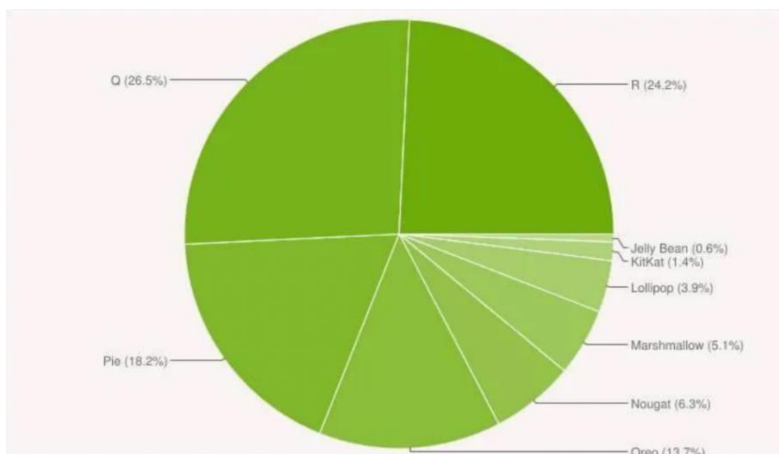
Android:

En 2007 Google se alió con el sector de dispositivos móviles y creó la alianza Open Handset Alliance. Asimismo, se anunció el sistema operativo Android, de código abierto con la finalidad de desarrollar aplicaciones (Karch, 2011)

Actualmente las versiones que se encuentran en los dispositivos Android son Q, R, Pie, con un mayor desarrollo de aplicaciones en estas versiones, porque son los que más usuarios tienen; como muestra el siguiente gráfico de datos recogidos el 24 de noviembre de 2021.

Figura 1

Estadísticas de las versiones de Android. (captura: Business Insider)



Patrón de diseño (M-V-C)

Modelo: En esta fase modelo se encuentran las diferentes clases de los datos que se utiliza en el sistema.

```
data class EmpleadoModel(  
    val id: Int?, val nombres: String, val apellidos: String, val fechaNacimiento: String, val  
    movil: String, val direccion: String, val identificacion: String, val clave: String, val rol:  
    String)
```

Vista: Una vista es compuesta por los lenguajes específicos de dominio de Kotlin con la finalidad de crear contenido mediante las tecnologías HTML y CSS.

Ejemplo: Formulario de inicio de sesión

```
class LoginView: Template<HTML>{  
    val cabecera: Cabecera=Cabecera()  
    val alerta=Placeholder<FlowCenter>()  
    override fun HTML.apply(){  
        insert(cabecera){}  
        head{  
            link(rel="stylesheet", href="/resources/LoginStyle.css") {}  
        }  
        body{  
            attributes["style"]="background-image:URL('resources/fondo.jpg'); background-  
            size:cover;""  
            form(classes = "form-signin", method = FormMethod.post, action =  
            Endpoints.LOGIN.url) {  
                div(classes="card"){  
                    div(classes="card-body"){  
                        h1(classes = "h3 mb-3") { +"Inicio de Sesión" }  
                        br{}  
                        div{ insert(alerta) }  
                        label(classes = "sr-only") {  
                            attributes["for"] = "identificacion"+"Identificación" }  
                        input(classes = "form-control", type = InputType.text, name = "identificacion") {  
                            attributes["id"] = "identificacion"  
                            attributes["placeholder"] = "Identificación"  
                            attributes["required"] = "true"  
                            attributes["autofocus"] = "true" }  
                    }  
                }  
            }  
        }  
    }  
}
```



```
label(classes = "sr-only") {  
    attributes["for"] = "clave"+"Contraseña" }  
input(classes = "form-control", type = InputType.password, name = "clave") {  
    attributes["id"] = "clave"  
    attributes["placeholder"] = "Contraseña"  
    attributes["required"] = "true"  
    attributes["autofocus"] = "true" }  
br{}  
button(classes = "btn btn-lg btn-primary btn-block", type = ButtonType.submit) {  
    +"Ingresar" }  
}  
}  
}  
}  
}  
}
```

Se puede observar que el desarrollo está en el lenguaje kotlin, asimismo para el funcionamiento de la vista se utiliza el lenguaje parecido a HTML.

Por otro lado, aplicar estilos es similar, usando la palabra reservada rule, ejemplo:

```
rule(".btn-search"){  
    borderRadius = LinearDimension("0px .25rem .25rem 0px")  
    border = "2px solid transparent"  
}
```

Controlador: En esta fase se encuentran todas las operaciones que se pueden ejecutar en el sistema, por ejemplo, las diferentes conexiones con la base de datos según la acción que se realice. Ejemplo: Insertar nuevo empleado

```
object EmpleadoManager{  
    fun nuevoEmpleado(empleado: EmpleadoModel: Int){  
        var result=-1  
        return if  
        (buscarEmpleadoPorIdentificacion(empleado.identificacion).isEmpty()){  
            transaction{  
                result=EmpleadoTable.insert{  
                    it[nombres]=empleado.nombres  
                    it[apellidos]=empleado.apellidos
```



```
it[fechaNacimiento]=empleado.fechaNacimiento
it[movil]=empleado.movil
it[direccion]=empleado.direccion
it[identificacion]=empleado.identificacion
it[clave]=empleado.clave
it[rol]=empleado.rol
}[EmpleadoTable.id]
}
result
}else
-2
}
}
```

Conclusiones

El lenguaje de programación Kotlin mediante la investigación que se ha realizado se obtiene que es un sistema donde sus códigos son abiertos, limpios, ordenados y sobre todo entendible, cuyo objetivo es ir evolucionando de tal manera que la plataforma de Android tenga un crecimiento sobre el uso de este lenguaje, con una fuerte alianza de la empresa GOOGLE. Este lenguaje demuestra ser competente, gracias a la sintaxis que maneja, por ende, vale la pena aprender y desarrollar cualquier sistema en él.

Referencias

- Al-Emran, M., Arpaci, I., & Salloum, S. A. (2020). An empirical examination of continuous intention to use m-learning: An integrated model. *Education and Information Technologies*, 1-20. <https://link.springer.com/article/10.1007/s10639-019-10094-2>
- Breslav, Andrey (2012). *Language of the Month: Kotlin*. drdobbs.com
- Devic, M. (2018). *Kotlin Quick Start Guide*, Packt Publishing



- Furió, D., Juan, M. C., Seguí, I., & Vivó, R. (2015). Mobile learning vs. traditional classroom lessons: a comparative study. *Journal of Computer Assisted Learning*, 31(3), 189-201. GitHub JetBrains/Kotlin
- Grin, T. (2017). Kotlin programming language. Available. <https://kotlinlang.org/assets/kotlin-media-kit.pdf>
- Heiss, J. (2013). The Advent of Kotlin: A Conversation with JetBrains' Andrey Breslav. oracle.com. Oracle Technology Network.
- Hsieh, W. M., & Tsai, C. C. (2017). Taiwanese high school teachers' conceptions of mobile learning. *Computers & Education*, 115, 82-95.
- Krill, P. (2011). JetBrains readies JVM language Kotlin. infoworld.com. InfoWorld.
- M. Karch, (2011). Android Tablets Made Simple, Apress, "Android," [Online]. Available: https://www.android.com/intl/es_es/everyone/enabling-opportunity
- Orozco, A. G. (2021). Que es Kotlin y Características. Obtenido de OpenWebinars: <https://openwebinars.net/blog/que-es-kotlin/>
- Pascuas, Y., Garcia, J. y Mercado, M. (2020). Dispositivos móviles en la educación: tendencias e impacto para la innovación. *Revista Politécnica*, 16(31), 97-109. <https://doi.org/10.33571/rpolitec.v16n31a8>
- RebelLabs (2013). "JVM Languages Report extended interview with Kotlin creator Andrey Breslav". <http://zeroturnaround.com/>
- Shafirov, M. (2017). Kotlin on Android. Now oficial. Today, at the Google I/O keynote, the Android team announced first-class support for Kotlin.
- Subramaniam, V. (2019). *Programming Kotlin*. Pragmatic Bookshelf
- Tomaschko, M., & Hohenwarter, M. (2017). Integrating mobile and sensory technologies in mathematics education. In *Proceedings of the 15th International Conference on Advances in Mobile Computing & Multimedia* (pp. 39-48). <https://doi.org/10.1145/3151848.3151866>
- Waters, J. (2012). *Kotlin Goes Open Source*. ADTmag.com/. 1105 Enterprise Computing Group.

