



# **Actividad [#3] - [Menú de Áreas de Figuras Geométricas]**

**[Desarrollo de Aplicaciones Móviles III]**

**Ingeniería en Desarrollo de Software**

**Tutor: Sandra Luz Lara Dévora**

**Alumno: Alan David López Rojas**

**Fecha: 12/07/2023**

# Índice

|                           |         |
|---------------------------|---------|
| Introducción.....         | pág. 3  |
| Descripción.....          | pág. 4  |
| Justificación.....        | pág. 5  |
| Desarrollo.....           | pág. 6  |
| Codificación.....         | pág. 6  |
| Prueba de aplicación..... | pág. 8  |
| Conclusión.....           | pág. 11 |
| Bibliografía.....         | pág. 12 |

## Introducción

En el ámbito de la programación, es frecuente encontrarse con la necesidad de calcular áreas de figuras geométricas en diversas aplicaciones. Estas aplicaciones pueden ser utilizadas en distintos campos, como la arquitectura, la ingeniería o la física, por mencionar algunos ejemplos. Por lo tanto, resulta importante contar con herramientas que permitan realizar estos cálculos de manera eficiente y precisa.

En este contexto, se plantea el desarrollo de una aplicación en lenguaje Swift que cumpla con el objetivo de calcular áreas de distintas figuras geométricas. La aplicación contará con un menú de opciones que permitirá al usuario seleccionar la figura de su interés, entre las cuales se encuentran el cuadrado, el rectángulo, el triángulo y el círculo.

Una vez que el usuario elija la figura deseada, se le solicitarán los datos necesarios para el cálculo del área correspondiente. Por ejemplo, en el caso del cuadrado se pedirá ingresar el valor del lado, mientras que en el caso del triángulo se requerirá la base y la altura. La aplicación realizará los cálculos correspondientes utilizando las fórmulas matemáticas adecuadas y mostrará el resultado al usuario.

Esta aplicación ofrece una solución práctica y conveniente para aquellos que necesitan calcular áreas de figuras geométricas de manera frecuente o esporádica. La interfaz de usuario amigable y el menú de opciones intuitivo facilitan su uso, incluso para aquellos sin conocimientos avanzados en matemáticas. Además, al automatizar el proceso de cálculo, se reducen los errores humanos y se agiliza la obtención de resultados precisos.

## Descripción

El contexto plantea la necesidad de desarrollar una aplicación en lenguaje Swift que permita calcular áreas de distintas figuras geométricas. La aplicación contará con un menú de opciones donde el usuario podrá elegir la figura de su interés: cuadrado, rectángulo, triángulo o círculo. Una vez seleccionada la figura, se le solicitarán los datos necesarios para realizar el cálculo del área correspondiente, como el lado del cuadrado, los lados del rectángulo, la base y altura del triángulo, o el radio del círculo.

El objetivo principal de la aplicación es brindar una herramienta práctica y eficiente para calcular áreas de figuras geométricas. Esto resulta útil en diversos campos, como la arquitectura o la ingeniería, donde el conocimiento de las áreas de las figuras es fundamental. La aplicación busca simplificar el proceso de cálculo, evitando errores humanos y agilizando la obtención de resultados precisos.

La interfaz de usuario de la aplicación es amigable y presenta un menú intuitivo que permite al usuario elegir la figura y proporcionar los datos necesarios para el cálculo del área. Una vez ingresados los datos, la aplicación realizará los cálculos correspondientes utilizando las fórmulas matemáticas adecuadas y mostrará el resultado al usuario de manera clara y comprensible.

## **Justificación**

La utilización de una aplicación para el cálculo de áreas de figuras geométricas ofrece una serie de beneficios significativos que respaldan su empleo en la actividad propuesta.

En primer lugar, esta solución permite obtener resultados precisos y confiables. Al contar con algoritmos y fórmulas predefinidas, la aplicación garantiza la exactitud de los cálculos, evitando errores humanos que podrían ocurrir al realizar los cálculos manualmente. Esto es especialmente relevante en el contexto de las figuras geométricas, donde la precisión es esencial para obtener resultados confiables y tomar decisiones informadas.

Además, el uso de una aplicación agiliza el proceso de cálculo. Al proporcionar una interfaz intuitiva y fácil de usar, los usuarios pueden ingresar los valores necesarios para cada figura geométrica de manera rápida y sencilla. La aplicación se encarga de realizar los cálculos de forma automática, eliminando la necesidad de realizar operaciones matemáticas repetitivas y ahorrando tiempo y esfuerzo.

Otro aspecto importante es la portabilidad y accesibilidad de la aplicación. Al estar desarrollada en lenguaje Swift, la aplicación puede ser utilizada en dispositivos móviles como smartphones y tablets, lo que brinda la flexibilidad de realizar cálculos de áreas en cualquier lugar y en cualquier momento. Esto resulta especialmente útil para profesionales y estudiantes que necesitan calcular áreas de figuras geométricas en situaciones de campo o en movimiento.

El uso de una aplicación para el cálculo de áreas de figuras geométricas ofrece ventajas en términos de precisión, eficiencia y accesibilidad. Proporciona resultados confiables, simplifica el proceso de cálculo y brinda flexibilidad para realizar los cálculos en cualquier momento y lugar. Su implementación en la

actividad propuesta mejorará la efectividad y productividad al calcular áreas de figuras geométricas de manera más rápida y precisa.

## Desarrollo

### ❖ Codificación

```
import Foundation

func calcularAreaCuadrado(lado: Double) -> Double {
    return lado * lado
}

func calcularAreaRectangulo(base: Double, altura: Double) -> Double {
    return base * altura
}

func calcularAreaTriangulo(base: Double, altura: Double) -> Double {
    return (base * altura) / 2
}

func calcularAreaCirculo(radio: Double) -> Double {
    return Double.pi * radio * radio
}

func mostrarMenu() {
    print("Menu de areas")
    print("1- Area del Cuadrado")
    print("2- Area del Rectangulo")
    print("3- Area del Triangulo")
    print("4- Area del Circulo")
    print("Por favor introduce una opcion (numero):")
}

func obtenerOpcion() -> Int {
    if let input = readLine(), let opcion = Int(input) {
        return opcion
    } else {
        return 0
    }
}

func obtenerDouble() -> Double {
    if let input = readLine(), let valor = Double(input) {
        return valor
    } else {
        return 0.0
    }
}
```

```

}

func calcularArea() {
  mostrarMenu()
  let opcion = obtenerOpcion()

  switch opcion {
    case 1:
      print("Introduce el valor del lado:")
      let lado = obtenerDouble()
      let area = calcularAreaCuadrado(lado: lado)
      print("El area del cuadrado es: \(area)")
    case 2:
      print("Introduce el valor del lado 1:")
      let lado1 = obtenerDouble()
      print("Introduce el valor del lado 2:")
      let lado2 = obtenerDouble()
      let area = calcularAreaRectangulo(base: lado1, altura: lado2)
      print("El area del rectangulo es: \(area)")
    case 3:
      print("Introduce el valor de la base:")
      let base = obtenerDouble()
      print("Introduce el valor de la altura:")
      let altura = obtenerDouble()
      let area = calcularAreaTriangulo(base: base, altura: altura)
      print("El area del triangulo es: \(area)")
    case 4:
      print("Introduce el valor del radio:")
      let radio = obtenerDouble()
      let area = calcularAreaCirculo(radio: radio)
      print("El area del circulo es: \(area)")
    default:
      print("Opcion invalida")
  }
}

calcularArea()

```

Este código define varias funciones para calcular el área de diferentes figuras geométricas: cuadrado, rectángulo, triángulo y círculo. Luego, se implementa una función `mostrarMenu()` para mostrar el menú de opciones y obtener la opción elegida por el usuario. Dependiendo de la opción seleccionada, se solicitarán los valores necesarios (lados, base, altura, radio) y se llamará a la función correspondiente de cálculo del área. Finalmente, se muestra el resultado.

Primero, se definen cuatro funciones `calcularAreaCuadrado`, `calcularAreaRectangulo`, `calcularAreaTriangulo` y `calcularAreaCirculo`, que reciben los valores necesarios para cada figura y calculan el área correspondiente.

La función `mostrarMenu()` se encarga de imprimir en la consola las opciones del menú.

La función `obtenerOpcion()` se encarga de obtener la opción elegida por el usuario desde la entrada estándar y devuelve un entero. Si el valor ingresado no es válido, se retorna 0.

La función `obtenerDouble()` se encarga de obtener un valor decimal ingresado por el usuario desde la entrada estándar y devuelve un número de tipo `Double`. Si el valor ingresado no es válido, se retorna 0.0.

La función `calcularArea()` es la función principal del programa. Primero, llama a `mostrarMenu()` para mostrar las opciones disponibles. Luego, obtiene la opción elegida por el usuario utilizando `obtenerOpcion()`. A continuación, se utiliza una estructura `switch` para manejar cada una de las opciones.

Dependiendo de la opción elegida, se solicitarán los valores necesarios utilizando la función `obtenerDouble()`. Luego, se llama a la función correspondiente de cálculo de área y se almacena el resultado en una variable llamada `area`. Finalmente, se imprime el resultado utilizando la función `print()`.

Si el usuario elige una opción inválida, se imprimirá el mensaje "Opción inválida".

Al final del código, se llama a la función `calcularArea()` para iniciar la ejecución del programa.

El flujo del programa es el siguiente:

Se muestra el menú de opciones.

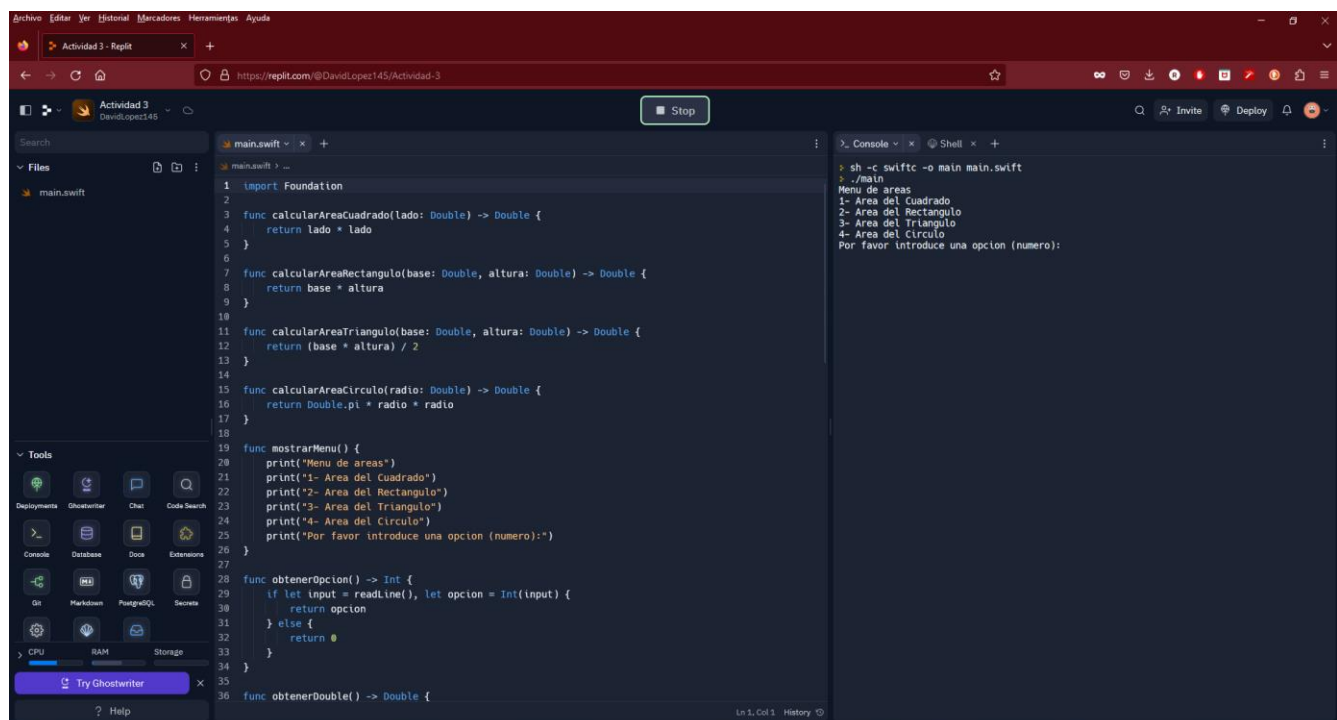
El usuario selecciona una opción.

Se solicitan los valores necesarios según la opción elegida.

Se realiza el cálculo del área correspondiente.

Se muestra el resultado en la consola.

## ❖ Prueba de la aplicación



The screenshot shows a Replit environment with a Swift file named `main.swift` and a terminal window. The code defines functions for calculating the area of a square, rectangle, triangle, and circle, along with functions to display a menu, get user input, and get double values. The terminal output shows the menu being displayed and the user being prompted to enter an option.

```
1 import Foundation
2
3 func calcularAreaCuadrado(lado: Double) -> Double {
4     return lado * lado
5 }
6
7 func calcularAreaRectangulo(base: Double, altura: Double) -> Double {
8     return base * altura
9 }
10
11 func calcularAreaTriangulo(base: Double, altura: Double) -> Double {
12     return (base * altura) / 2
13 }
14
15 func calcularAreaCirculo(radio: Double) -> Double {
16     return Double.pi * radio * radio
17 }
18
19 func mostrarMenu() {
20     print("Menu de areas")
21     print("1- Area del Cuadrado")
22     print("2- Area del Rectangulo")
23     print("3- Area del Triangulo")
24     print("4- Area del Circulo")
25     print("Por favor introduce una opcion (numero):")
26 }
27
28 func obtenerOpcion() -> Int {
29     if let input = readLine(), let opcion = Int(input) {
30         return opcion
31     } else {
32         return 0
33     }
34 }
35
36 func obtenerDouble() -> Double {
```

The terminal output shows the following text:

```
> sh -c swiftc -o main main.swift
./main
Menu de areas
1- Area del Cuadrado
2- Area del Rectangulo
3- Area del Triangulo
4- Area del Circulo
Por favor introduce una opcion (numero):
```



## Área del cuadrado

The screenshot shows a Swift playground with the following code in `main.swift`:

```
1 import Foundation
2
3 func calcularAreaCuadrado(lado: Double) -> Double {
4     return lado * lado
5 }
6
7 func calcularAreaRectangulo(base: Double, altura: Double) -> Double {
8     return base * altura
9 }
10
11 func calcularAreaTriangulo(base: Double, altura: Double) -> Double {
12     return (base * altura) / 2
13 }
14
15 func calcularAreaCirculo(radio: Double) -> Double {
16     return Double.pi * radio * radio
17 }
18
19 func mostrarMenu() {
20     print("Menu de areas")
21     print("1- Area del Cuadrado")
22     print("2- Area del Rectangulo")
23     print("3- Area del Triangulo")
24     print("4- Area del Circulo")
25     print("Por favor introduce una opcion (numero):")
26 }
27
28 func obtenerOpcion() -> Int {
29     if let input = readLine(), let opcion = Int(input) {
30         return opcion
31     } else {
32         return 0
33     }
34 }
35
36 func obtenerDouble() -> Double {
```

The console output shows the menu and the result of the first calculation:

```
> sh -c swiftc -o main main.swift
./main
Menu de areas
1- Area del Cuadrado
2- Area del Rectangulo
3- Area del Triangulo
4- Area del Circulo
Por favor introduce una opcion (numero):
1
Introduce el valor del lado:
4
El area del cuadrado es: 16.0
```

## Área del rectángulo

The screenshot shows the same Swift playground as before, but with the console output updated to show the calculation of a rectangle's area:

```
> sh -c swiftc -o main main.swift
./main
Menu de areas
1- Area del Cuadrado
2- Area del Rectangulo
3- Area del Triangulo
4- Area del Circulo
Por favor introduce una opcion (numero):
2
Introduce el valor del lado 1:
5
Introduce el valor del lado 2:
6
El area del rectangulo es: 30.0
```

## Área del triángulo

```
1 import Foundation
2
3 func calcularAreaCuadrado(lado: Double) -> Double {
4     return lado * lado
5 }
6
7 func calcularAreaRectangulo(base: Double, altura: Double) -> Double {
8     return base * altura
9 }
10
11 func calcularAreaTriangulo(base: Double, altura: Double) -> Double {
12     return (base * altura) / 2
13 }
14
15 func calcularAreaCirculo(radio: Double) -> Double {
16     return Double.pi * radio * radio
17 }
18
19 func mostrarMenu() {
20     print("Menu de areas")
21     print("1- Area del Cuadrado")
22     print("2- Area del Rectangulo")
23     print("3- Area del Triangulo")
24     print("4- Area del Circulo")
25     print("Por favor introduce una opcion (numero):")
26 }
27
28 func obtenerOpcion() -> Int {
29     if let input = readLine(), let opcion = Int(input) {
30         return opcion
31     } else {
32         return 0
33     }
34 }
35
36 func obtenerDouble() -> Double {
```

```
> sh -c swiftc -o main main.swift
> ./main
Menu de areas
1- Area del Cuadrado
2- Area del Rectangulo
3- Area del Triangulo
4- Area del Circulo
Por favor introduce una opcion (numero):
3
Introduce el valor de la base:
12
Introduce el valor de la altura:
8
El area del triangulo es: 48.0
```

## Área del círculo

```
1 import Foundation
2
3 func calcularAreaCuadrado(lado: Double) -> Double {
4     return lado * lado
5 }
6
7 func calcularAreaRectangulo(base: Double, altura: Double) -> Double {
8     return base * altura
9 }
10
11 func calcularAreaTriangulo(base: Double, altura: Double) -> Double {
12     return (base * altura) / 2
13 }
14
15 func calcularAreaCirculo(radio: Double) -> Double {
16     return Double.pi * radio * radio
17 }
18
19 func mostrarMenu() {
20     print("Menu de areas")
21     print("1- Area del Cuadrado")
22     print("2- Area del Rectangulo")
23     print("3- Area del Triangulo")
24     print("4- Area del Circulo")
25     print("Por favor introduce una opcion (numero):")
26 }
27
28 func obtenerOpcion() -> Int {
29     if let input = readLine(), let opcion = Int(input) {
30         return opcion
31     } else {
32         return 0
33     }
34 }
35
36 func obtenerDouble() -> Double {
```

```
> sh -c swiftc -o main main.swift
> ./main
Menu de areas
1- Area del Cuadrado
2- Area del Rectangulo
3- Area del Triangulo
4- Area del Circulo
Por favor introduce una opcion (numero):
4
Introduce el valor del radio:
10
El area del circulo es: 314.1592653589793
```

<https://replit.com/join/twwbnfooqg-davidlopez145>

<https://github.com/AlanDavidLR/Desarrollomovil3.git>

[https://drive.google.com/file/d/1QyQGV9daDXAE6eacy\\_8OL4uFEHsEHxMN/view?usp=sharing](https://drive.google.com/file/d/1QyQGV9daDXAE6eacy_8OL4uFEHsEHxMN/view?usp=sharing)

## Conclusión

La realización de la actividad de crear una aplicación en lenguaje Swift para calcular áreas de figuras geométricas tiene una gran relevancia tanto en el campo laboral como en la vida cotidiana, en el ámbito laboral, esta solución proporciona una herramienta eficiente y precisa para profesionales que trabajan con cálculos de áreas, como arquitectos, ingenieros, diseñadores y otros campos relacionados. La aplicación les permite ahorrar tiempo al realizar los cálculos de manera automática, evitando posibles errores humanos y brindando resultados confiables. Además, al estar disponible en dispositivos móviles, pueden llevar la aplicación consigo a las obras, reuniones o cualquier lugar donde necesiten realizar cálculos de áreas, mejorando su productividad y facilitando su trabajo.

En la vida cotidiana, esta aplicación también tiene un impacto significativo. Muchas personas pueden encontrarse en situaciones en las que necesitan calcular áreas de figuras geométricas, como al realizar proyectos de bricolaje, planificar el diseño de un jardín o incluso al hacer compras de materiales para renovaciones en el hogar. La aplicación les permite realizar estos cálculos de manera rápida y precisa, sin la necesidad de recurrir a fórmulas complejas o realizar cálculos manuales.

A lo largo de las tres actividades presentadas, he podido aprender y aplicar diferentes conceptos y habilidades relacionadas con la programación en lenguaje Swift y el desarrollo de aplicaciones móviles. Cada actividad ha abordado un objetivo específico y ha requerido el uso de distintos elementos y enfoques para lograrlo.

En conclusión, las tres actividades han proporcionado un enfoque práctico para el aprendizaje de Swift y el desarrollo de aplicaciones móviles. Se han abordado aspectos esenciales como el uso de estructuras de control, la manipulación de objetos y clases, el manejo de datos y la creación de interfaces interactivas. Estas habilidades y conocimientos adquiridos son fundamentales para la programación y el desarrollo de aplicaciones en diversos contextos laborales y cotidianos.

## **Bibliografía**

Stack Overflow Contributors, Aprendizaje Swift Language, 2019, Manual-informática, En línea:  
<https://manual-informatica.com/download-file.html>