

DISEÑO Y LAYOUT DE INTERFACES GRÁFICAS

NOTAS:

- En una vista no puede haber 2 textos, por eso lo que se puede hacer es combinar ambos en una Vstack.
- Puedo embeber varios objetos en una misma vista seleccionándolos en el canvas y pulsando la tecla command + el botón izquierdo del ratón y aparece un desplegable y puedo embeberlos en la pila que quiera (Vstack, Zstack, lista...)
- Refactorizar texto: Seleccionar el código con la tecla command pulsada y hacer clic y en el desplegable seleccionar ExtractSubview y se crea una nueva estructura a la que le podemos cambiar el nombre y la cual es llamada en nuestro stack pero su código queda aparte para que no moleste.

En el ejemplo de las dos Vstack que hemos utilizado para los dos planes del ejemplo (el plan básico y el plan carrera), nos damos cuenta que hay muchas cosas iguales y solo algunas que cambian, como el tipo de plan, el precio, la descripción y el valor del texto, y el color de fondo. La idea general es que se vea de los 15 ó 20 líneas de código, salvo 3 ó 4 es todo lo mismo. Así que hay que intentar siempre la REUSABILIDAD, reutilización de código.

Haciendo clic en la tecla Command y en una de las 2 Vstack seleccionamos extractSubview y creamos abajo una nueva estructura a la que le podemos cambiar el nombre y a su vez a la instancia de la misma. Así que luego añadimos una serie de variables a la estructura antes del body.

- El Zstack lo utilizamos igual que un .overlay para superponer una vista encima de la otra pero es algo más completo.
- Los opcionales en swiftUI con map: en los clousures el \$0 hace referencia a la primera variable, en este caso del ejemplo a "icon" ya que es la única.
- Espacios en blanco – **Spacer** (Spacer()): Es como echar goma espuma en contra de donde queremos mover el elemento en cuestión. En nuestro ejemplo si queremos mover la cabecera hacia la izquierda por ejemplo tendremos que cambiar la vstack en la que está alojada en una Hstack y le ponemos la goma espuma al final (Spacer()) y el texto se nos va a la izquierda. Por el contrario si lo ponemos al principio del Hstack nos desplazará el texto a la derecha.
- Podemos hacer el contorno de una CardView con un .overlay(RoundedRectangle(cornerRadius:10).stroke(Color.grey, .opacity: 0.2), lineWidth:2))

```
import SwiftUI
struct ExerciseView: View {
    var body: some View {
        VStack{
            Header1View()

            Pricing1View(title: "Plan Básico", subtitle: "Un curso incluido", price: "9,99", textColor: .white, backgroundColor: .green, icon: "star")
                .padding(.horizontal)

            ZStack {
                Pricing1View(title: "Plan carrera", subtitle: "Una carrera completa", price: "29,99", textColor: .black, backgroundColor: Color(red: 200/255,
green: 200/255, blue: 200/255), icon: "timer")

                Text("El mejor para empezar")
                    .font(.system(.caption, design: .rounded))
                    .foregroundColor(.white)
                    .fontWeight(.black)
                    .padding(6)
                    .background(Color(red: 240/255, green: 180/255, blue: 50/255))
                    .cornerRadius(50)
                    .offset(x: 0, y: -80)
            }

            .padding(.all)
            .offset(x: 0, y: -80)

            ZStack{

                Pricing1View(title: "Definitivo", subtitle: "Todos los cursos online", price: "99.99", textColor: .white, backgroundColor:Color(red: 95/255,
green: 95/255, blue: 95/255), icon: "lightbulb")

                .padding(.horizontal)

                Text("Conviértete en un máster del Universo")
                    .font(.system(.caption, design: .rounded))
                    .foregroundColor(.white)
                    .fontWeight(.black)
                    .padding(6)
                    .background(Color(red: 240/255, green: 180/255, blue: 50/255))
                    .cornerRadius(50)
                    .offset(x: 0, y: -80)

                .padding(.horizontal)
```

```

        }.padding(.vertical)
        .offset(x: 0, y: -180)
        Spacer()
    }
}
}
}
struct ExerciseView_Previews: PreviewProvider {
    static var previews: some View {

        ExerciseView()
    }
}
struct Header1View: View {
    var body: some View {
        HStack {

            VStack(alignment: .leading, spacing: 3){
                Text("Elige tu itinerario")
                    .font(.system(.largeTitle, design: .rounded))
                    .fontWeight(.black)

                Text("de aprendizaje")
                    .font(.system(.largeTitle, design: .rounded))
                    .fontWeight(.black)

            }.padding()
            Spacer()
        }
    }
}
struct Pricing1View: View {

    var title: String
    var subtitle: String
    var price: String
    var textColor: Color
    var backgroundColor: Color
    var icon: String?

    var body: some View {
        ZStack {
            VStack{

                icon.map({
                    Image(systemName: $0) // $0 = icon
                        .font(.largeTitle)
                        .foregroundColor(.white)
                })

                Text(title)
                    .font(.system(size: 25, design: .rounded))
                    .fontWeight(.bold)
                    .foregroundColor(textColor)

                Text(price)
                    .font(.system(size: 35, weight: .heavy, design: .rounded))
                    .foregroundColor(textColor)

                Text(subtitle)
                    .font(.system(size: 14))
                    .foregroundColor(textColor)

            }
        }.frame(minWidth: 0, idealWidth: .infinity, maxWidth: 120)
        .padding()
        .background(backgroundColor)
        .cornerRadius(15)
    }
}
}
}

```