Universidad de los Andes

ISIS-1106 Lenguajes y máquinas
TP 1
Fecha: February 14, 2023

Para esta primera parte del taller vamos a definir la gramática para la definición de Views para la definición de interfaces gráficas de Swift en SwiftUI. Views corresponde a un protocolo que puede ser extendido desde cualquier estructura `struct`. Dentro de la clase, los views se caracterizan por tener un atributo `body` definido como un bloque asociado a una variable de tipo `var body : some View {...` dentro de la implementación de `body` es posible tener uno o mas elementos como: otros Views, componentes, o elementos estructuradores.

- Los `Views` corresponden a estructuras que implementan `View` previamente definidas y están dadas por el nombre de la estructura seguido de paréntesis.
- Los componentes corresponden a elementos predefinidos como `Image` o `Text` las cuales se pueden definir con un parámetro de tipo String. Adicionalmente se puede usar el componente `Spacer`, que no recibe ningún parámetro y no implementa ningún método.
- Los elementos estructuradores consisten en elementos para agrupar otros elementos verticalmente (*i.e.,* `VStack`) y horizontalmente (*i.e.,* `HStack`) los cuales definen un bloque (delimitado entre corchetes) que agrupa otros elementos .

Adicionalmente, cada uno de los elementos que aparecen en la definición del view pueden llamar métodos de formato (excepto `Spacer`).

La definición de los métodos es la siguiente.

- `offset` es aplicable para todos los elementos y recibe como parámetros opcionales las coordenadas del plano (`x:` y `y:` separadas por comas) y un número
- `padding` es applicable para todos los elementos y recibe como parámetro (opcional) la dirección desde la cual se toma el espacio (`.top`, `.bottom`, `.left`, `.right`) con el valor correspondiente del padding, separados por comas
- `font` es aplicable para los elementos estructuradores y `Text` y recibe por parámetro el tipo de font a utilizar (*e.g.,* `.title`, `.title2`, `.title3`. `.body`
- `foregroundColor` aplicable para los elementos estructuradores y `Text` y recibe por parámetro el color a utilizar (*e.g.,* `.green`, `.red`, `.background`, `.foreground`

Task 1. Su trabajo es definir la gramática en EBNF que pueda generar interfaces gráficas basadas en Views. Dentro de la gramática debe definir cadenas de dígitos (para definir números) y las cadenas de caracteres (para definir los nombres de las clases). Adjunto a la actividad encontrará ejemplos de programas definiendo views