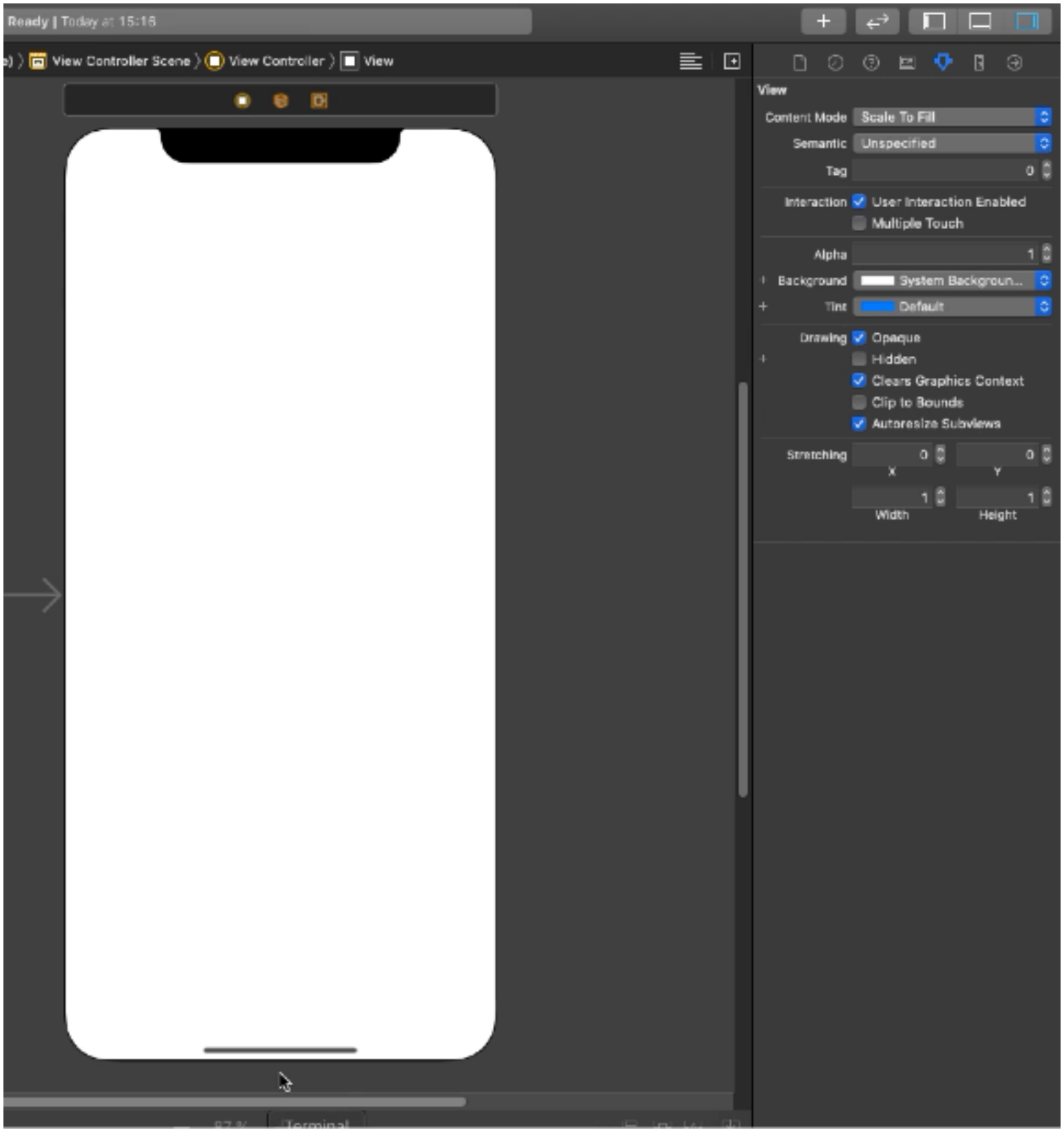


SF Symbols

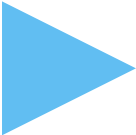
sf symbols deprecate storyboard

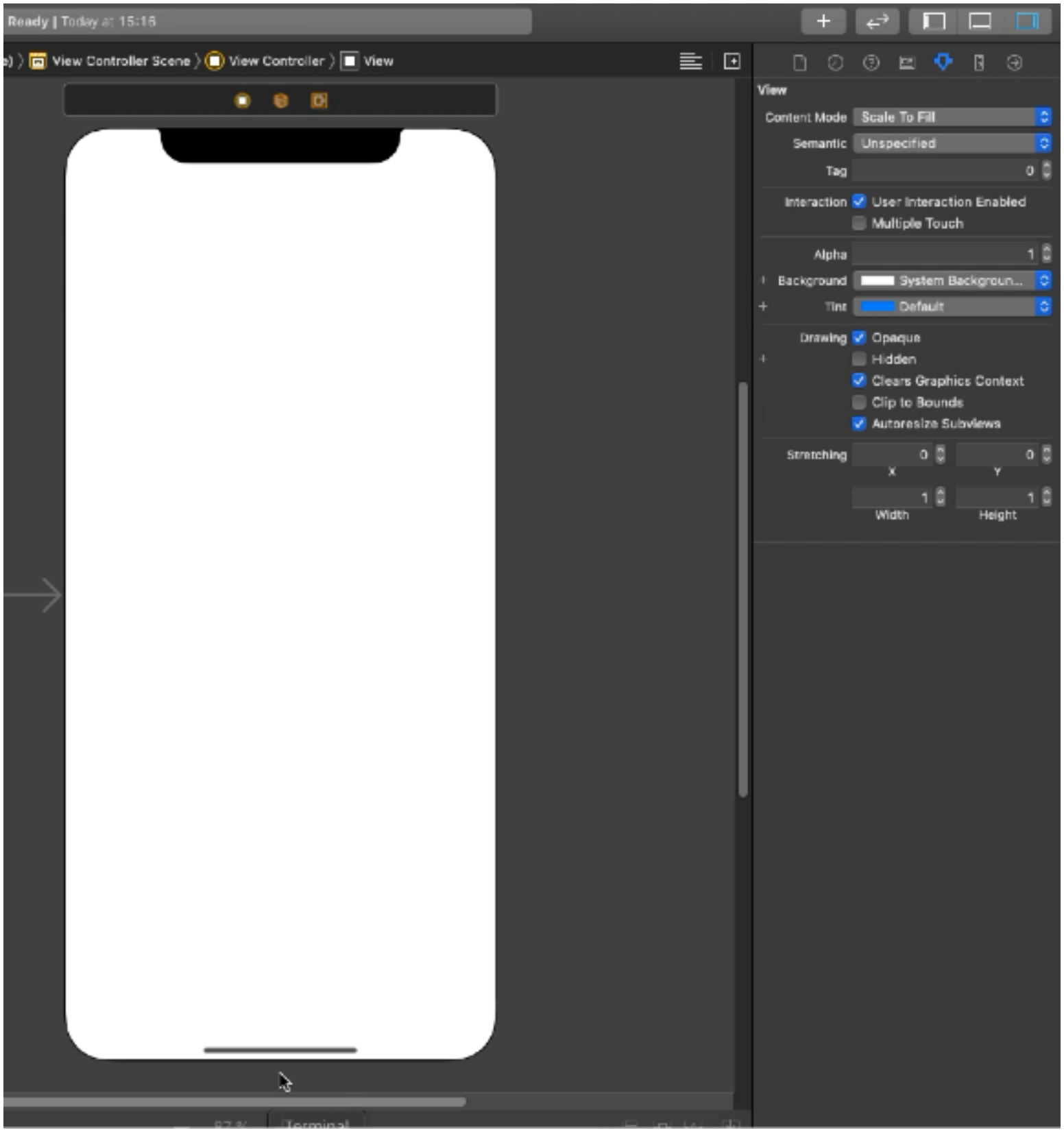
Dans le storyboard (donc si on ne se sert pas de SwiftUI) on peut directement ajouter les logo de SF Symbols depuis la propriété image (concerne les UIImageView et les buttons

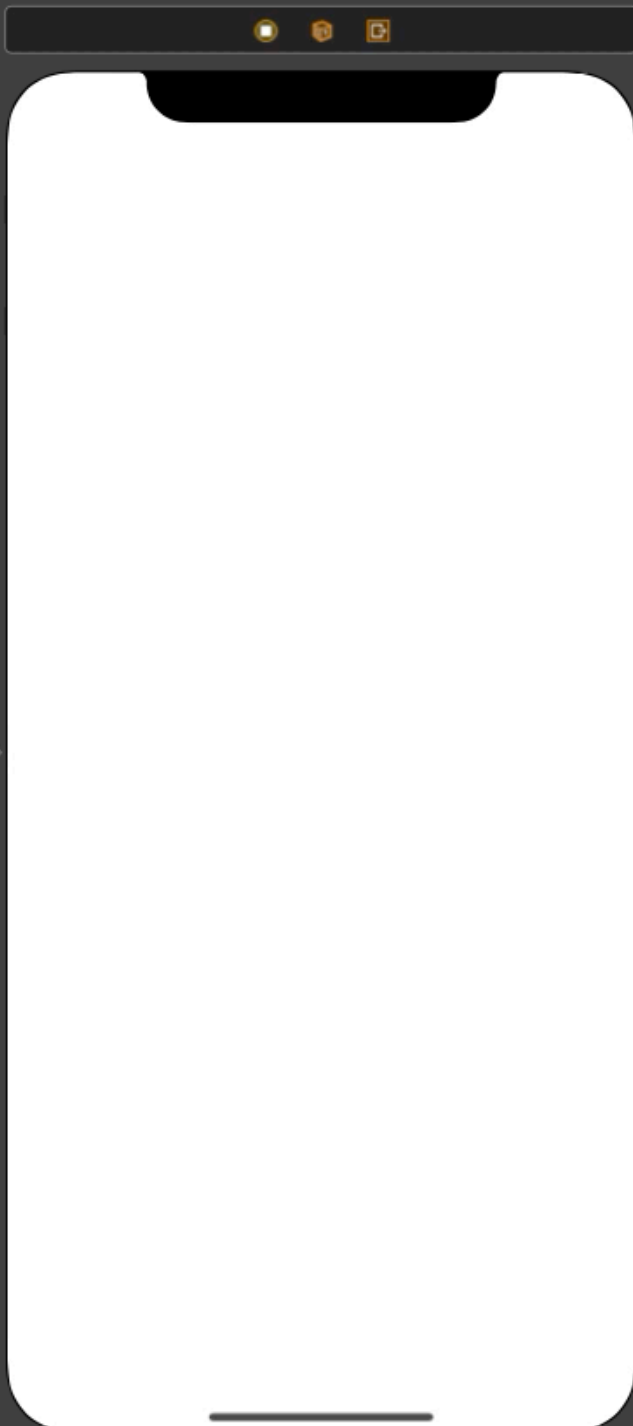


Pour ajouter ou modifier l'image d'un bouton dans le code :

```
trashButton.setImage(UIImage(systemName: "trash"), for: .normal)
```







View

Content Mode **Scale To Fill**Semantic **Unspecified**

Tag 0

Interaction ☒ User Interaction Enabled☐ Multiple Touch

Alpha 1

+ Background **System Backgroun...**+ Tint **Default**Drawing ☒ Opaque+ ☐ Hidden☒ Clears Graphics Context☐ Clip to Bounds☒ Autoresize Subviews

Stretching 0 0

X

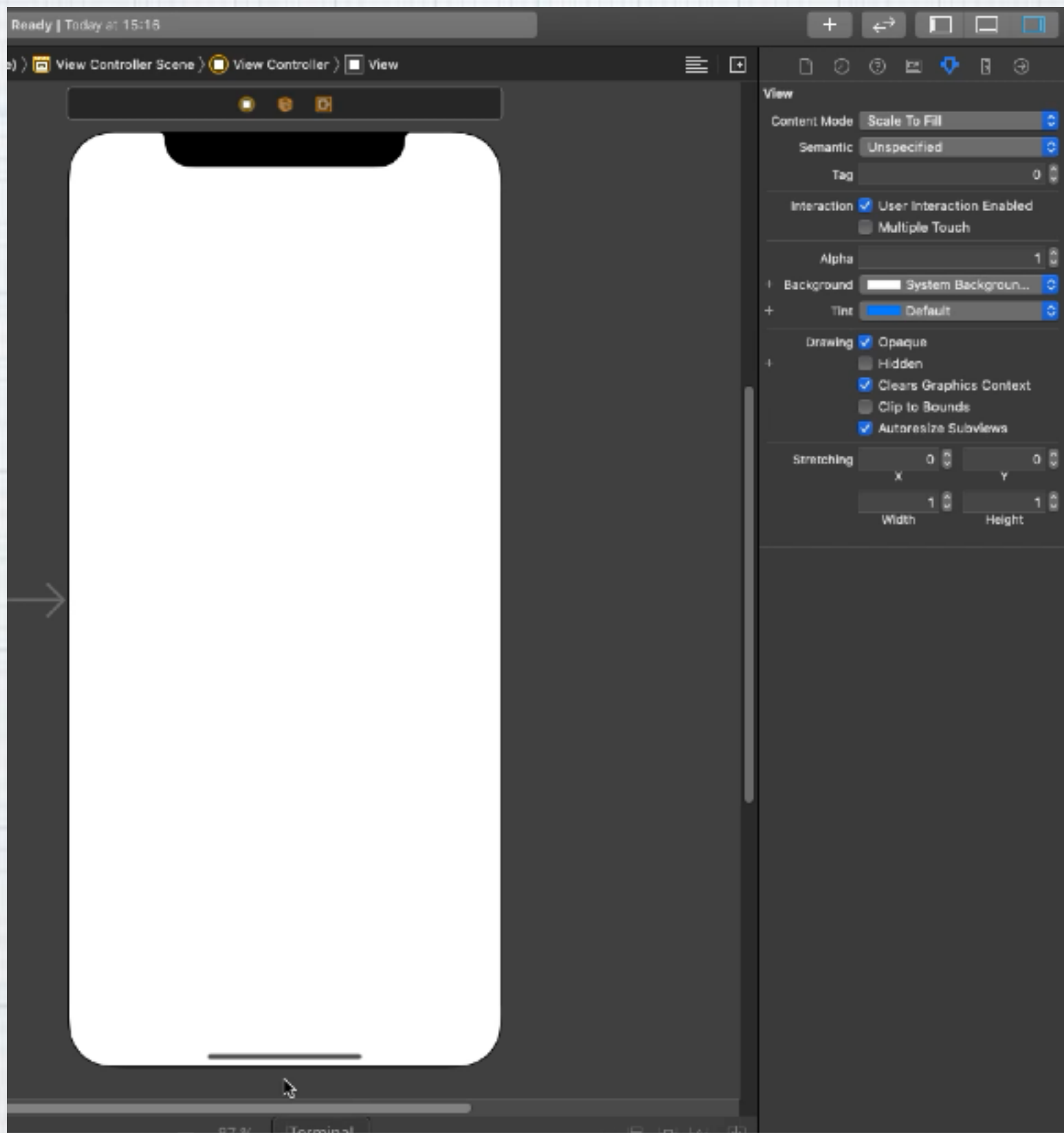
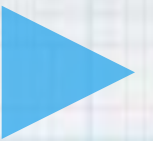
Y

1 1

Width

Height

SF Symbols depuis le storyboard



Dans le storyboard (donc si on ne se sert pas de SwiftUI) on peut directement ajouter les logo de SF Symbols depuis la propriété image (concerne les imageView et les buttons

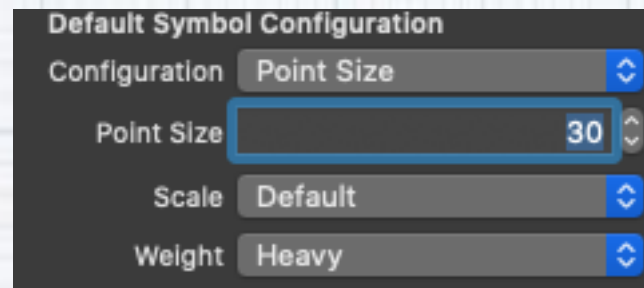
Pour ajouter ou modifier l'image d'un bouton directement dans le code :

```
trashButton.setImage(UIImage(systemName: "trash"), for: .normal)
```

SF Symbols

Personnaliser l'affichage

On peut choisir la police du symbole directement dans le storyboard avec le panneau Default Symbol Configuration



On peut modifier la police du symbole dans le code, en créant une constante contenant la configuration voulu et en l'appliquant aux endroits voulus, c est valable pour les boutons et les images.

```
let configurationDuSymbole = UIImage.SymbolConfiguration(pointSize: 30.0, weight: .heavy, scale: .default)
```

```
trashButton.setPreferredSymbolConfiguration(configurationDuSymbole, forImageIn: .normal)
```

Pour conclure, SF Symboles est très pratique car les icônes sont directement utilisable dans Xcode. On peut les utiliser dans le storyboard ou/et dans le code. Finalement cela permet d'améliorer l'interface de nos applications.