Afin d'illustrer un paradigme plus procédural voici un autre exemple qui concerne la vue d'authentification (première vue que l'utilisateur voit en lançant l'application). Il faut comprendre qu'il y a une variable globale nommée "loginState" définie en tant qu'entier (Integer) qui peut prendre plusieurs valeurs en fonctions des évènements :

@State var loginState:Int = 0

// 0 : Login

// 1 : Settings

// 2 : Working Settings

// 3 : Error saving settings

// 4 : Settings recorded

// 5 : Try To authenticate

// 6 : Success

// 7 : Error authentication

Chaque évènement, par exemple une clique sur un bouton, peut déclencher une modification de la vue. Dans ce cas, une fonction à été définie et appelée par cet évènement afin de modifier la vue :

//Switch view

func loginSwitchToMain(){

loginState = 0

}

func loginSwitchToSettings(){

loginState = 1

}

func loginSwitchToWorkingSettings(){

loginState = 2

}

func loginSwitchToErrorSettings(){

loginState = 3

}

func loginSwitchToSuccessSettings(){

loginState = 4

}

func loginSwitchToTryAuthenticate(){

user.login = userLogin

user.saveUser()

loginState = 5

}

func loginSwitchToSuccessAuthenticate(){

loginState = 6

}

func loginSwitchToErrorAuthenticate(){

loginState = 7

}

Voici un exemple de bouton, celui-ci permet de lancer la procédure d'identification :

Button(action: loginSwitchToTryAuthenticate){

Text("LOGIN")

.fontWeight(.semibold)

.foregroundColor(Color.white)

.background(Color("Turquoise"))

.frame(width:315, height: 32)

.overlay(

RoundedRectangle(cornerRadius: 12)

.stroke(Color.white, lineWidth: 0)

)}