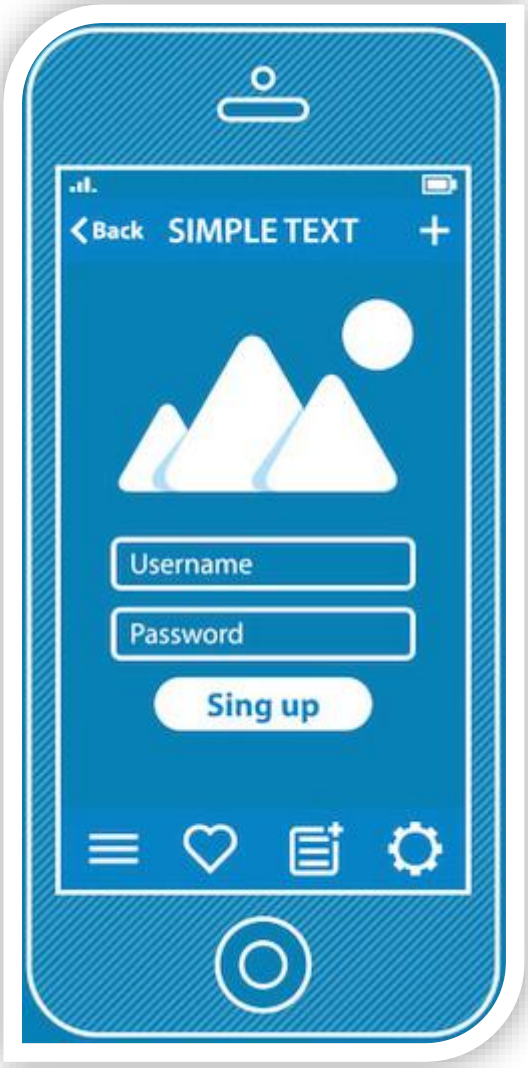
Diseño de Interfaces de Usuario



Nombre: Zabai Armas Herrera

Curso: 4º

Práctica 2. Monitorizar eventos del ratón.

Deslizador y Barra de Progreso.

Base:

- Como se indica en la ficha de práctica, he inicializado todos los componentes del JFrame con un valor adecuado para su correcta visualización en pantalla. En otras palabras, los campos de texto por defecto a "---" hasta que ocurra algún evento relacionado con el ratón y el deslizador. Este último al iniciar el programa se encuentra a la mitad de su máximo valor, por lo que la barra de progreso también.

Mejora y añadidos:

- Para hacerlo más cómodo he decidido que al iniciar el programa, lo haga con su ventana centrada a la pantalla. De esta manera, mayoritariamente, se evita obligar al usuario a mirar a otro sitio ya que por defecto la ubica en la esquina superior izquierda de la pantalla.

- Como se propone en la práctica, he implementado un botón (llamado "RGB") que hace visible otro JFrame con el sistema RGB por deslizadores y campos de texto.

- En la realización de esta mejora hay que tener en cuenta que: al tratarse de "crear" colores se necesita precisión. Así que se puede cambiar cada componente de color tanto por el deslizador como por campo de texto. Al modificarlo de cualquier manera se actualizarán los parámetros, es decir: si cambias el color por entrada de texto, el deslizador cambiará al valor introducido y viceversa.

- Al introducir el ratón en el deslizador se cambia exclusivamente el color del texto respecto al deslizador actual e informa del estado del ratón.

- A parte de la estructura propuesta, como tengo cierta familiaridad con las interfaces de software de diseño gráfico (Adobe Photoshop/Illustrator, Krita, Affinity, …) me parece indispensable que tenga un campo que indique el valor hexadecimal del color actual. Además de indicarlo, da la posibilidad al usuario de introducir uno en hexadecimal ya que también actualizará todas las variables. Sólo hay que respetar el formato "#RRGGBB".

- Si el usuario cierra la ventana no sale de la aplicación, sino que la deja invisible, guardando así el estado en el que se dejó. Nota: no he contemplado los fallos de entrada, lo que se traduce a poder introducir un número fuera del rango RGB (0-255) o hexadecimal (#000000 - #FFFFFF).

Capturas: