

Componente

Índice

1. Proceso de elaboración.....	3
1.1. Parte común.....	3
1.2. Elaboración de la funcionalidad.....	3
1.3. Prueba del componente.....	4
2. Decisiones de diseño.....	4
3. Conclusión.....	4

1. Proceso de elaboración

Este ejercicio se basa en la creación de un componente para interactuar con fechas, con una interfaz que muestre tres campos para introducir día, mes, y año. El componente debe incluir una funcionalidad aplicable a las fechas introducidas.

1.1. Parte común

Para una mayor eficiencia a la hora de testear todos los componentes se ha realizado una selección entre todos los miembros de la clase sobre unas funciones que deben añadir todos los componentes, incluyendo los parámetros que reciben y los que devuelven. Estas funciones son:

- Constructor vacío sin una fecha predeterminada, ni recibe ni devuelve nada.
- Constructor con fecha predeterminada, recibe tres números enteros que corresponden a día, mes, y año; y no devuelve nada.
- Función “setFecha”, recibe tres números enteros que corresponden a día, mes, y año; y no devuelve nada.
- Función “getFecha”, no recibe nada y devuelve un `localDate`.
- Excepción “FechaException”, es una excepción que recibe un `String`; se usa para controlar cualquier error que pueda darse en el componente.

1.2. Elaboración de la funcionalidad

Una vez conocida la funcionalidad que debía incluir el componente había que definir el código que hace que funcione. En primer lugar decidimos creamos la clase del componente, con los parámetros que contiene; luego procedimos a crear los constructores, el vacío simplemente crea el componente llamando a otra función privada que explicare mas adelante, el constructor que recibe una fecha predeterminada establece los valores introducidos a los parámetros de la clase y llama a esa función privada. El “setFecha” llama a una función privada que comprueba que la fecha sea correcta y la añade a los parámetros de la clase, esta función puede devolver la excepción que habíamos definido comúnmente. El “getFecha” devuelve un `LocalDate` con los valores de los parámetros que se han establecido previamente. La función que comprueba la fecha, comprueba que sea una fecha válida que exista en el calendario y que sea posterior al 1900, si no es así devolverá una excepción.

La función privada “setComponent” definen lo que va a formar el componente, con sus atributos que va a tener en pantalla y como se va a visualizar.

La función “dayDifference” se realiza sobre una fecha recibiendo otra con la que se quiere comparar y se devuelven los días de diferencia.

1.3. Prueba del componente

Para probar el componente se ha pedido que realicemos una aplicación en la que se llame al componente dos veces para establecer dos fechas y que al accionar un botón se llame a una función del componente en la que se devuelva los días de diferencia entre estas dos fechas, esto se mostrará en pantalla con un alert.

2. Decisiones de diseño

Para el diseño hemos querido realizar algo sencillo basándonos en las indicaciones que habíamos recibido por parte de el profesor. Estas son los tres campos de texto establecidos horizontalmente con sus respectivas indicaciones del nombre de cada campo sobre los mismos.

3. Conclusión

Como conclusión, al ya tener mas experiencia desarrollando el código sin la asistencia de un IDE con múltiples ayudas, el trabajo ha resultado mas sencillo en ese aspecto; sin embargo la dificultad en este recae en realizar la funcionalidad del componente aislada del resto de código de manera que no haya problema al interactuar con diferentes ejemplos.