IES de Teis
Avda. de Galicia, 101
36216 – Vigo
Tfno: 886 12 04 64
e-mail: ies.teis@edu.xunta.es
http://www.iesteis.es

EXPLICACION PRÁCTICA 4

Funcionamiento

- Ejecutar la aplicación
- Muestra la ventana principal en la que el usuario introduce los datos
- Al pulsar el botón **Guardar**, se comprueba que ningún cuadro de texto está vacío. Para ello accede a todos los hijos del panel en el que están los cuadros de texto. Si alguno está vacío no permite almacenar los datos. Se pueden añadir más validaciones para comprobar que los datos cumplen con las especificaciones, por ejemplo, número de dígitos, etc. Si los datos están validados, e crea el objeto alumno con los datos del alumno y se crea un arraylist de módulos. Al elegir los módulos se debe comprobar que no se supera el número de créditos permitidos. En el caso de que se superen sale un Alert con un mensaje indicando al usuario el Error y desactiva el último modulo elegido, así como quitar de la lista de modulos ese módulo y restar los créditos
- Al pulsar el botón **Cancelar**, limpia todos los cuadros de texto, limpia la lista de módulos y pone el cursor en el cuadro de texto del nombre (**No consigo que aparezcan los checkbox desactivados en la interfaz, aunque si que los desactiva)**
- El botón **Salir** finaliza la aplicación
- Los datos de matriculación de cada alumno se deben mostrar en una nueva ventana. Para ello se utilizan los métodos getUserData y setUserData. Debido a esto se necesita un botón RecuperarDatos en la ventana secundaria, necesario para poder acceder al Stage de la aplicación y a través de ese Stage obtener los datos con getUserData.
- **TableView** –> El método **SetCellValueFactory** –> Método que especifica como rellenar todas las celdas de una columna. Para mostrar los datos de un objeto, deben estar implementados los métodos get y set de cada propiedad. Además para poder acceder a las propiedades de las clases, esta debe ser pública
- Tampoco funcionan los Tooltip con los checkbox. Una posible solución es añadir un botón INFO que muestre los créditos de cada módulo (**Pendiente de implementar)**
- **Pendiente de implementar (Modificar el aspecto con css)**
- **ObservableList –>** Es una lista de JavaFX permite a los listener escuchar los cambios cuando se producen. Es necesaria para mostrar los datos en la TableView