**Monitor device assignment**

I did 2-5 sections and solved one bug related to the add item button which didn’t work.  
I would do the other authority sections if I had more time.

Explanation about the sections:

Section 2, Done by deleting the object from the list of device monitor and updating the device list view.

section 3, I created two classes “SoftwareDevice” and “HardwareDevice” that represents Device and inherits from GeneralDevice. New kinds of devices will be built by adding one class that represents them and inherits from generalDevice.   
The application supports the two types of devices because those devices are also represent GeneralDevice so a few changes were needed.

Section 4, The user can choose the type of device when adding new one by choosing the desired option in the combox.  
after choosing the desired option the appropriate containers will be shown.

Section 5, the option to show the device details will be attached to “remove” option in the MenuItem, after choosing the details option an alert window will be opened and show the device details according to its type.

**Things I would add or change if I add more time**

* Add validation to the user input
* Handling better errors.
* Use more constants instead of creating such as strings used for UI labels or button labels, into separate static final fields to make them easier to manage and update.
* Would change the logic in the showMainScene() to get back to the DeviceMonitorApp class.
* I would pay attention to separation of concerns and modularization theme,

considering breaking down the code into smaller classes or modules to improve separation of concerns. For instance, to separate classes for UI management, device handling, and dialog creation it also allows reusability.

* Another important theme is the dependency injection - instead of creating instances of UI elements within the controller methods, I would use dependency injection to pass these elements as parameters to the methods. This can make your methods more testable and decoupled.
* Handling the problem that happens after saving new item by checking if there is something to improve in the code so that the devicemonitor data will be saved when getting back to the DeviceMonitorApp class
* Change the installation date and time text field to datepicker and the SoftwareDevice class data member respectively
* Changing the location text field to combo box using specified locations
* Implement Singleton for the controller because one instance is enough.