

## **Grade2 Fix**

This is actually identical to my Grade4 attempt, but since learning how to do my own observer in workshop 3 I understood that I could remove all observable list and properties. I initially thought it would be ok to use since the model wasn't bound to JavaFX. But when I saw the comments about killed kittens in slack I understood that encapsulation was violated and if not sending the list out the model it was no point of keeping it a Observable list. The multiplicity was actually fixed after the peer, but unfortunately I had changed the name of the diagram and both the old and the new one was in the folder. At least I hope you saw the wrong one, since I think it was ok to only add multiplicity for the associations that is not 1 to 1, otherwise I think it would add to much clutter.

## **Login**

hardcoded - user: admin, password: admin

## **UML**

With the help of the controller I could remove or rather move some dependencies from the ViewClasses and model. Which now instead are from controller to view and from controller to model. The views still has some dependencies to the model though.

In the DesignClassDiagram I have excluded all dependencies from CRUDController to the views, and instead added a dependency arrow to the package view and showed the operations that open new views in the class instead.

filter is in separate class diagram.

As discussed in grade3 I think that the design is a little bit more messier than it had to be because moved from MV to MVC when the view started to feel to low cohesion instead of designing for that in the first place.

All relations are 1-1 if no multiplicity is added.