Allgemeine Design-Entscheidungen

Wir haben uns dazu entschieden Datenbankobjekte von den Businessobjekten abzukapseln, weil die Datenbankobjekte automatisch generiert werden. Außerdem kann so objektbezogene Businesslogik in diesen Objekten untergebracht werden.

Für das GUI haben wir uns dazu entschieden die Businessobjekte mit einem Interface zu wrappen, damit nur die für das GUI benötigten getter und setter sichtbar sind.

UseCase 1)

In unserer Patientensuche ist es möglich nach Vorname, Nachname, SVN und Geburtsdatum (dd.mm.yyyy) zu suchen. Alles wird in einem String eingegeben der von unserer Logik interpretiert wird und die dazugehörigen Patienten liefert.

Damit das GUI während der Suche nicht blockiert, implementiert der PatientSearchController das Runnable Interface und der BussinessLogicalDelegationController startet einen neuen Thread, welcher laufend nach Patienten sucht. Das GUI wird über Events informiert, wenn der PatientSearchController neue Patienten gefunden hat und aktualisiert entsprechend.

Jede Änderung in dem Textfeld wird im PatientSearchController auf einen Stack gelegt, damit dieser immer die neuste Eingabe analysiert.

UseCase 2)