

Meeting Agenda

Group: Super duper Omega gruppen

Date: 2020-10-08 15:15

Chair: Sebastian

Participants: Samuel, Oskar, Erik, Behroz, Sebastian, Pelle Evensen

Objectives (5 min)

Fråga 1

Om man skriver @author på någon som t.ex. bara skrivit några rader, kommer inte det göra så att nån som skrivit 90% av filen får mindre cred?

Fråga 2

Behöver man ha author/purpose till test filer?

Fråga 3

Hur ska vi implementera TowerInfo? Ska man skapa ett torn med sin lambda för att läsa dess klass/range?

Fråga 4

Är visitor överkill i Wave Command? Borde man bara ha instanceof istället?

Fråga 5

Prata om ConcurrentModificationException i View och i Controller (kopia av listor till view, Command pattern till controller?)

Fråga 6

Kolla peer review

Discussion items (135 min)

Fråga 1:

Lägg till @author på allt, även om man bara lägger till en getter i en fil.

Fråga 2:

@author i test filer "om vi vill".
Skit i test kom vi överens om.

Fråga 3:

selectedTower borde heta selectedTowerType

Abstract Factory

Pelle skeptisk till TowerInfo. Pratar nånting om Proxy. Dessa borde vara i Controller.

TowerInfo borde skapas i Controller, inte ha en path. Sen väljs rätt bild till varje TowerInfo i View med map. OCP handlar mest om model.

Fråga 4:

Att använda en visitor är ok. Snyggare och mer typ säkrare än instanceOf.

Fråga 5:

Använd bara en kopia eller Collections.unmodifiable.

Använd synchronized block på kopia av collection, samt alla ställen där man använder collectionen.

<http://www.evensen.org/softcon/Evensen-SoftwareConstraints-HW3.pdf>

Länk till exempel på thread safety.

```
synchronized(this) {  
    o1.call(this);  
    o2.call(this);  
}
```

Fråga 6:

Outcomes and assignments (5 min)

Ny regel: lägg till @author även om du knappt har gjort något.

Refaktorisera towerInfo, kolla fråga 3.

Fixa concurrentModificationException med nya systemet beskrivet i fråga 5.

Erik lägger till @author i test filerna.

Wrap up

Next meeting: 2020-10-12 15:00