

Lab 2: wat ging er zoal fout?

- Fully dressed use case ging meestal prima/goed
- Maar sommige zaken verliepen uiterst moeizaam
 - >> Domain modeling
 - >> GRASP & responsibilities
 - >> UML class diagram
 - >> Patterns
- Doe er je voordeel mee t.b.v. lab 3

Domain modeling

- Dit heb je al gedaan bij Datamodelleren
- Toen gebruikte je ER-modellen
- Hier gebruik je UML light:
 - Geen methods
 - Lijntjes met cardinaliteiten en rolnamen
 - Nog geen connectoren (aggregatie etc.)
 - Wel ISA

Domain modeling: observaties

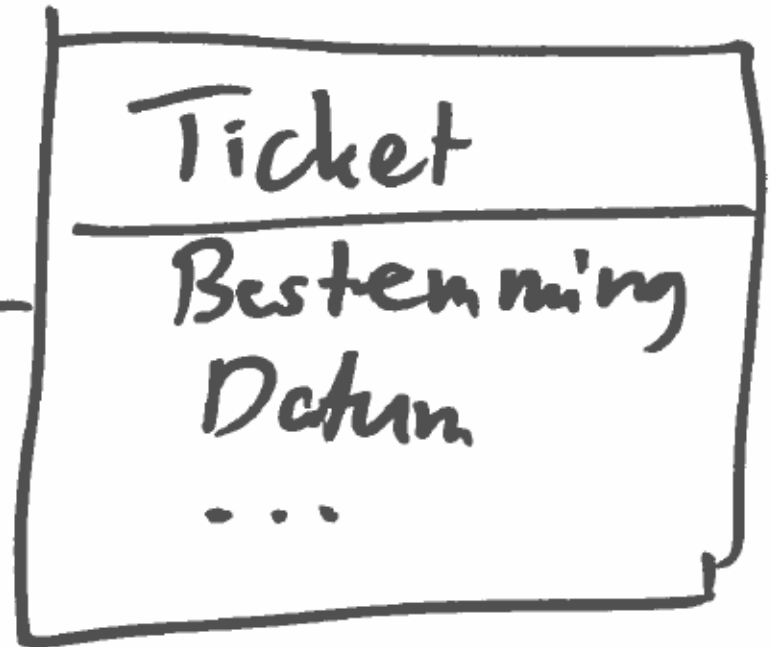
- De *klant*: ik kan mij voorstellen dat je die als entiteit in je domain model zet
- Maar bedenk dat je vaak (cash!) *niets* van een klant weet
- ... hooguit een kortingskaart
- De entiteit *klant* zal dus geen echte rol spelen in je verdere ontwerp

Domain modeling: observaties

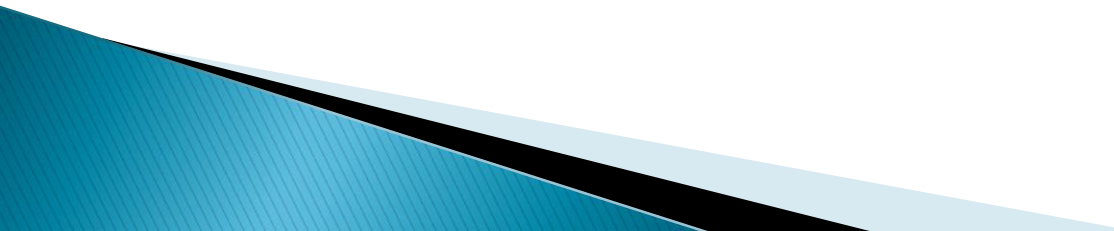
- Ook in je domain model heb je ISA hiërarchieën!
- Maar die voor de verschillende betaalmethoden, reisvarianties, kortingen ontbraken vaak
- Dit zijn relevante statische aspecten
- “Zwervende verbanden”



krijgt



GRASP & responsibilities

- Ik hoef geen uitleg wat GRASP in het algemeen inhoudt
 - Doel van de vraag: laat zien dat je iets van de principes begrijpt
 - “Entity *Transactie* (of *Ticket*) kent de start en bestemming en heeft toegang tot de afstandentabel; deze bepaalt de ritprijs” (INFORMATION EXPERT)
 - De gangbare drie principes (Info Expert, Creator, Controller) werden vaak niet genoemd
 - Of er worden zaken genoemd die irrelevant zijn (Encapsulation)
- 

GRASP & responsibilities

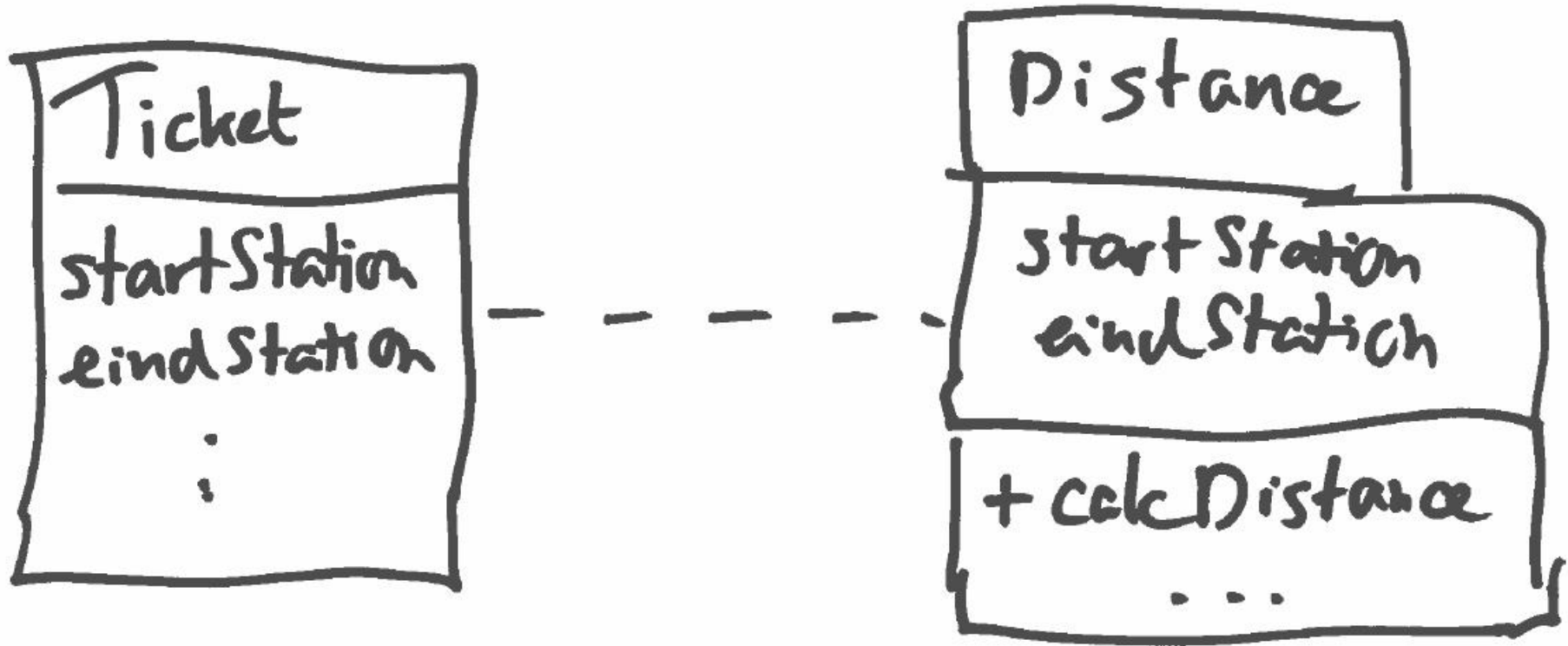
- Sommigen legden de verantwoordelijkheid voor het starten van een transactie bij entiteit Klant
- Maar in je uiteindelijke class diagram spelt de Klant eigenlijk geen rol van betekenis
- Wat in de werkelijke wereld gebeurt wordt niet altijd direct afgebeeld op de structuur van je software

UML class diagram

- Geen UML light meer
- Methoden toevoegen
- Connectoren in plaats van lijntjes: aggregation, composition
- Problemen:
- >> kopiëren van attributen
- >> kopiëren van methoden
- >> onjuiste plaatsing van methoden:
 - Class Sale wil afstand weten om ritprijs te bepalen; waar zet je `getDistance(st1,st2)`? In Sale??
- >> (mede daardoor) vage benamingen van methoden: “giveInformation”

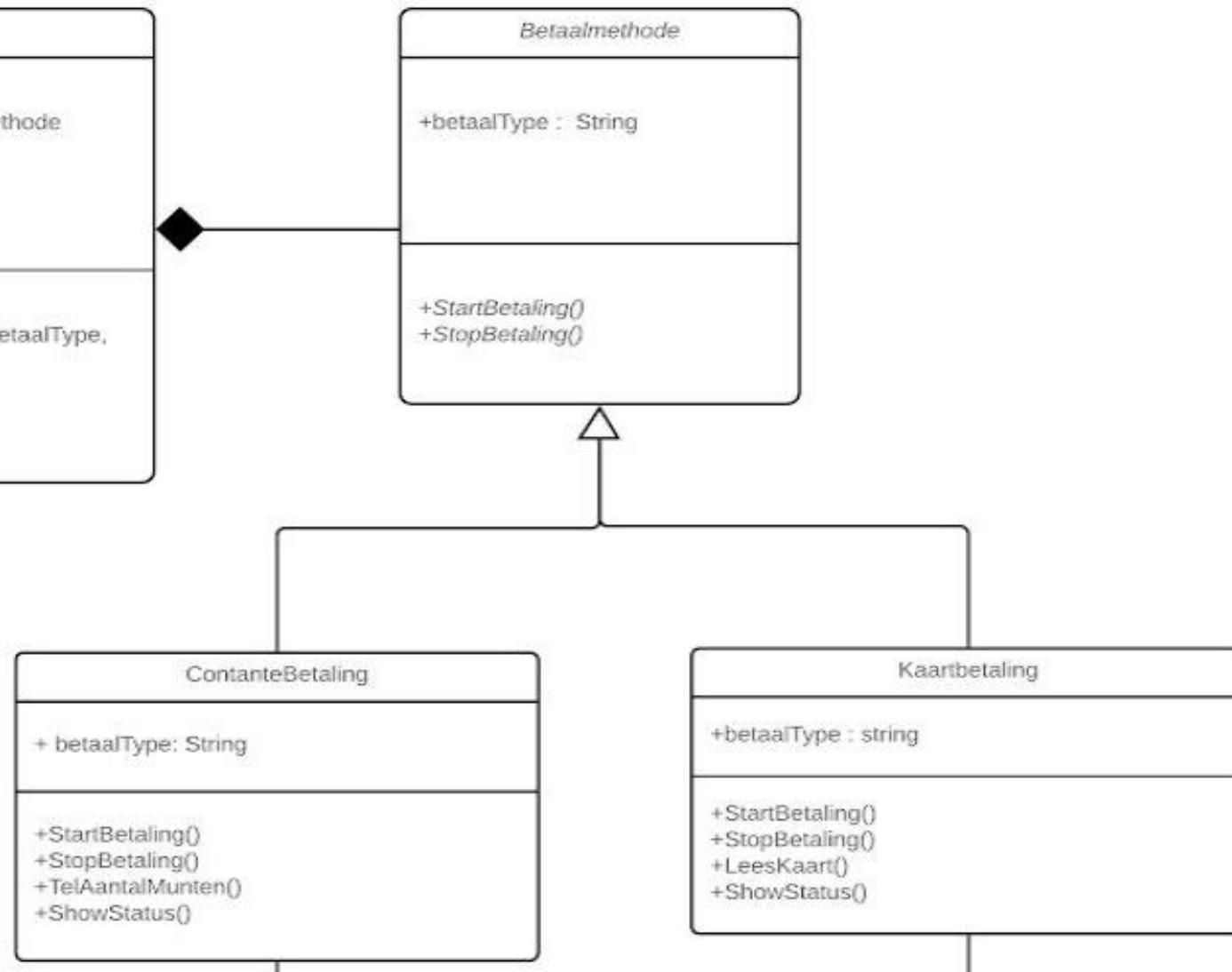
UML class diagram

Wat gebeurt hier?



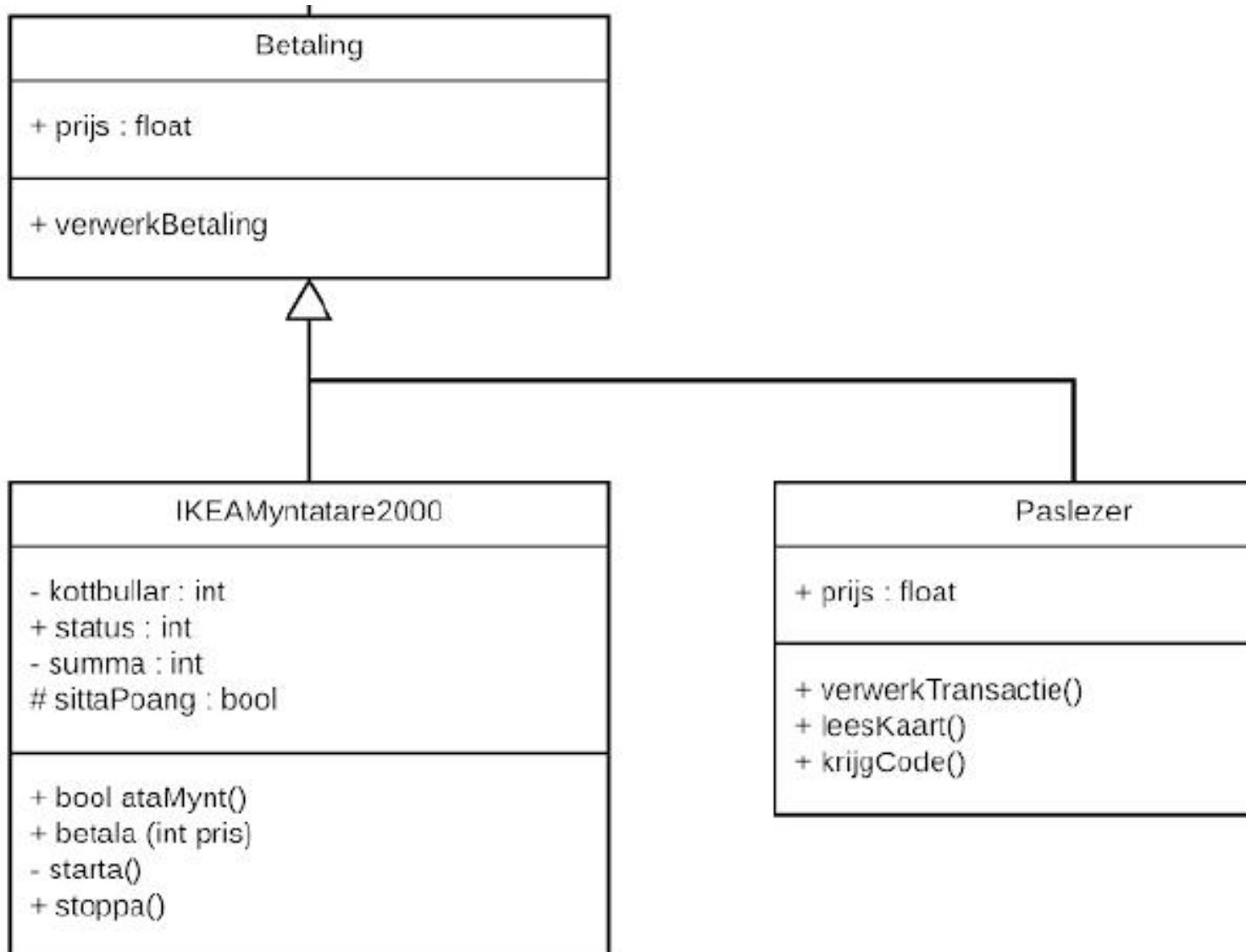
UML class diagram

Wat gaat hier goed?
Wat gaat hier fout?



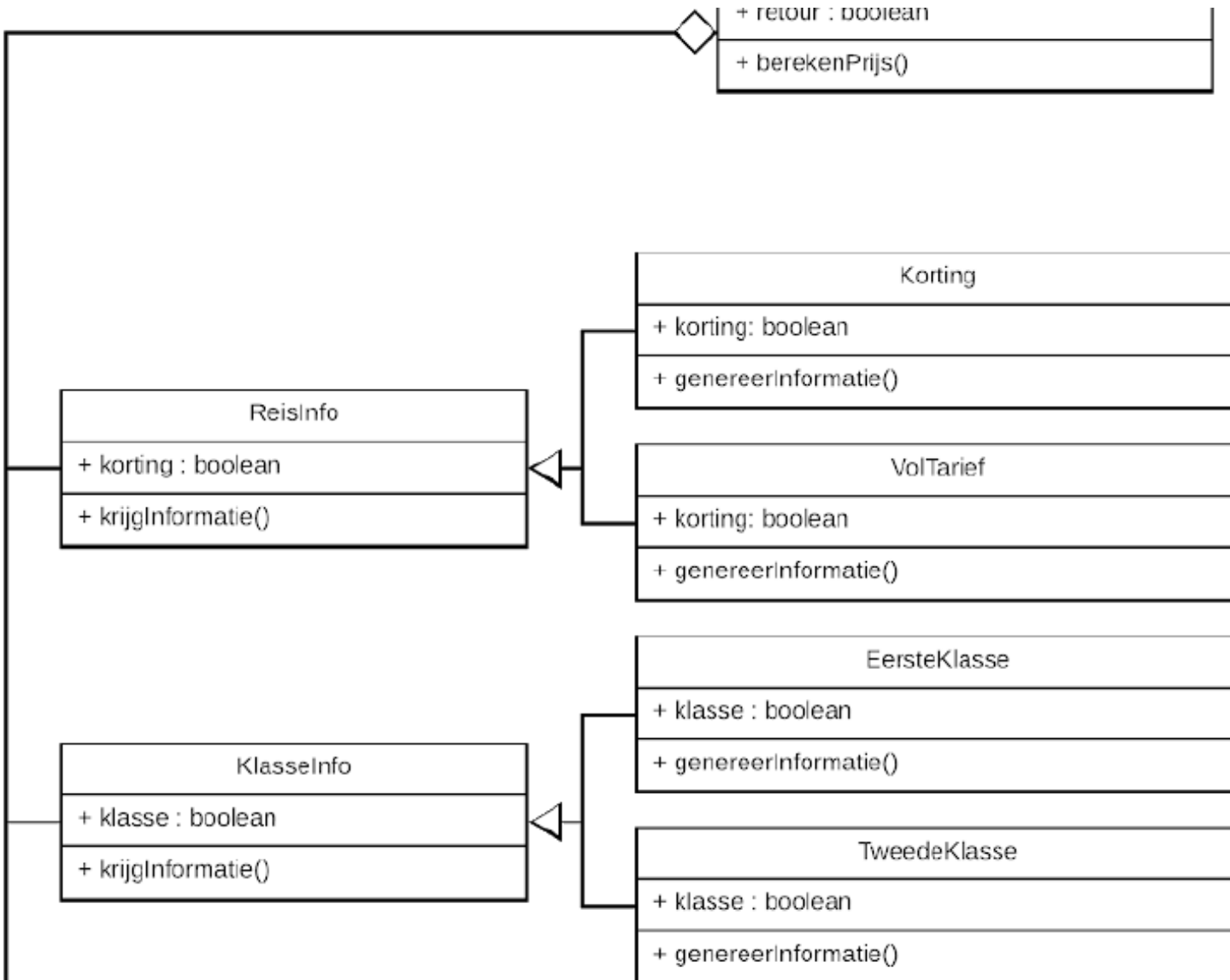
UML class diagram

Wat gebeurt hier?



UML class diagram

What's in a name?



Patterns

- Bestudeer de stof voordat je een practicumopgave maakt
- Benoem je patterns en ...
- Doe er iets mee!
- Een ISA hierarchie is niet automatisch een Strategy!

Tot slot

- Zie lab 2 als een (soms heftig) leermoment
- Maak iets moois van lab 3
- Succes!