

Objektorienterad utveckling

Programmering för språkteknologer 2

Sara Stymne

Delvis baserat på material från Tommy Olsson, LiU

2013-09-18

OO[ADP]

- ▶ OOA - objektorienterad analys
 - ▶ Analysera problemet
 - ▶ Skapa användningsfall – vad ska programmet kunna göra?
 - ▶ Bestäm vilka klasser som behövs och hur de ska samarbeta
- ▶ OOD – objektorienterad design
 - ▶ Förfinna och detaljera analysen
 - ▶ Bestäm metoder och variabler som ska finnas i klasser
- ▶ OOP – objektorienterad programmering
 - ▶ Implementera designen i något objektorienterat språk

OOA/OOD – möjlig arbetsgång

- ▶ Ta fram objekt kandidater
- ▶ Grovrensa bland objekten
- ▶ Klassificera objekten
- ▶ Gruppera och beskriv relationer mellan klasser
- ▶ Ta fram användningsfall och validera systemet
- ▶ Rita klassdiagram
- ▶ Detaljdesigna klasser

Ta fram oobjekt kandidater

- ▶ Brainstorming
- ▶ Substantiv-verb metoden
 - ▶ Substantiv – möjliga objekt
 - ▶ Verb – möjliga metoder
- ▶ Checklista
 - ▶ Faktiska ting (bil, hus)
 - ▶ Platser (rum, trädgård)
 - ▶ Begrepp (bankkonto, transaktion)
 - ▶ Roller (kund, spelare)
 - ▶ Händelser (landning, avbrott)
 - ▶ ...

Grovrensa bland objekten

- ▶ Är några objekt egentligen samma, men med olika namn?
- ▶ Är några tydligt orimliga, och kanske snarare borde vara variabler, t.ex.

Klassificera objekten

- ▶ Jobba till exempel med CRC-kort
 - ▶ C – class, klassens namn
 - ▶ R – responsibility, klassens ansvar
 - ▶ C – collaborators, klasser som klassen samarbetar med
- ▶ CRC-korten kan flyttas runt, uppdateras, slängas efterhand

Gruppera och beskriv relationer mellan klasser

- ▶ Vilka klasser hör ihop?
 - ▶ Lämpligt med arv/gränssnitt?
 - ▶ Lägga ihop till paket?

Användningsfall och validering

- ▶ Användningsfall
 - ▶ Vad ska programmet kunna göra?
- ▶ Validering
 - ▶ Kan klasserna användas för att utföra användningsfallen
 - ▶ Har vi rätt klasser?
 - ▶ Har vi hittat alla klasser som behövs?
 - ▶ Vilka klasser samarbetar, och hur?

Rita klasssdiagram

- ▶ Baserat på analysen

Detaljdesigna klasser

- ▶ Vilka metoder behövs
 - ▶ Framförallt icke-privata metoder – klassens gränssnitt
- ▶ Vilka variabler behövs
 - ▶ Vilka datastrukturer är lämpliga att använda?

Lab 4

- ▶ Textbaserat äventyrsspel
- ▶ Skriv egen kod från grunden
 - ▶ Strukturera upp koden i lämpliga klasser
 - ▶ Använd lämpliga datastrukturer
- ▶ Läs igenom instruktionen och kraven ordentligt innan ni sätter igång!

Kommande veckor

- ▶ Lab 3: deadline måndag 14/10
- ▶ Lab 4
 - ▶Handledning 14/10, 21/10
 - ▶Deadline 29/10
- ▶ Nästa föreläsning: 4/11
 - ▶Repetition
 - ▶Övningstenta
 - ▶Era frågor – maila gärna innan!
- ▶ Det mesta schemalagda är slut
 - ▶ Mycket eget arbete!
 - ▶ Hör av er om ni behöver hjälp/undrar något!
- ▶ Glöm inte anmäla er till tentan!

Jobba själv

- ▶ Gör labbar
- ▶ Gör programmeringsövningar
 - ▶ Från boken
 - ▶ Labbar från tidigare kursomgångar
- ▶ Läs/tentaplugga
- ▶ Gå igenom gamla tentor