

Alex Olsson har bidragit till projektet:

- Designskelet för Labb 1
- Bakgrund-, vädericon- och textfärgsändring för Labb 1
- Del av bokningssidan på Websidan
- Design av menysidan på Orderapplikationen
- Popupfönster för tillagningsgrad
- Delar av Labb 3

Command Pattern

Command Pattern eller Action Pattern är ett beteendemässigt designmönster som kapslar in en operation i ett objekt, vilket gör det möjligt att köra, köa och ångra operationer. Detta används då program vill kunna köa exekveringar, göra så de körs under senare tid eller vill kunna ångra en körning.

![[UML diagram av command pattern]](umlcom.png)

Fördelar med Command Pattern är att det frånkopplar sender och reciver objekten då de inte behöver "känna" varandra direkt samt gör det möjligt att lagra och ångra exekveringarna. Nackdelarna är att command pattern kan introducera onödig komplexitet om det räcker med vanliga funktionsimplementeringar.

![[Bild på kodexempel för ett command pattern i Java]](kodcom.png)

I koden ovan ser man att command interface definierar common metoden execute() som alla kommandon måste implementera. Under i Concrete Command implementeras kommandot och anropar den korresponderade metoden från Reciever. Reciever är den som gör den faktiska logiken då kommandot exekveras. Invokern lagrar Command objektet samt funktionen som exekverar kommandot. Client sätter upp kommandot och tilldelar det till invokern som kallar på kommandot vilket i sin tur kallar på reciever som printar ut att ljuset är på.

Diskussion Alex Olsson

I helhet skulle jag säga att projektet gjorde sin uppgift i att introducera applikationsutveckling i java i form av grupparbete. Vi skapade flertal förfrågade produkter till klienten det bästa vi kunde men jag tror vi skulle kunnat fått produkten klar tidigare om vi inte tog på oss för mycket jobb i onödan. Men vi tog oss igenom det och fick fungerande produkter.

Jag gillade att vi började dela upp arbetet i olika delar som alla jobbade på vilket förenklade arbetet men jag tycker att sättet vi delade upp det kunde förbättras. Det är alltid bra att ha en ledare i grupparbeten men det gör också att hen får mindre implementeringsjobb att göra samt att dela upp det i front-end och back-end gjorde att vissa bara gjorde samma saker under hela arbetet och för vissa var de saker som redan visste/var bekväm med. Det är inget fel med det men om man istället delar upp så alla får implementera lite back-end och front-end samt att bara göra så att man har en ledare i början för att dela upp arbetet skulle göra att alla lär sig nya saker utan att någon lämnas utanför.

Det var två saker jag inte gillade, en var att vi skrev upp onödigt många delar att jobba med som i slutändan inte implementerades då de var ja, onödiga och överkomplicerade och det andra var att rapporten sköts upp ganska rejält och sättet vi skrev den på var förvirrande och tog onödigt lång tid. Det bästa, då vi ändå är sju personer, skulle varit att alla skrev i ett Google dokument eller Word dokument som alla har tillgång till och har skrivit i förut där alla kan se vilka delar som är skrivna och inte istället för att dela upp så vissa skriver på andra ställen.

Labb 1 var en bra introduktion till kursen för gruppen, att designa delar av appen var roligt och jag fick göra lite java kodning för att få till så bakgrunden på väderappen ändras efter väder vilket andra sedan kopplade till API:et. Front-end är mer lockande för mig men javakodningen var roligt att göra också. Dock tycker jag att vi som Civilingenjörer i datateknik borde ha fler kurser i java än denna då för oss som var mer front-end fokuserade inte använde så mycket java kod och majoriteten av det var tillsammans med xml kod. Webbdesignen på bokningssidan blev i slutändan ändrad men de generella delar jag implementerade var kvar i slutet som tidsbokning, namn, epost och informationstext. Designen av orderapplikationen specifikt menysidan var det roligaste då det var utmanande och intressant. Appen hade mest javamoment för mig som front-end då en massa RVadapters och andra funktioner och objekt skapades för att lagra och hantera informationen från databasen.

Jag tror att med bättre tidsplanering skulle arbetet kunna implementera de delar som valdes att skippas och i framtiden skulle hemsidan och appen kunna användas i en riktig restaurang då både orderappen, köksappen, webbsidan och databasen var fungerande (med några brister).