

I denna peer review ska jag försöka fånga de övergripande positiva och negativa delarna med er design och arkitektur för grundkraven i workshop 2.

Efter att följt lämnade instruktioner kunde jag lätt starta programmet med kommandot "java -jar thehappypirate.jar". Det var lätt att använda och man fick bra hjälp med hur man skulle "röra" sig i programmet. Kraven för funktionerna som skrivs under workshop 2 är uppfyllda och jag tycker att Martin löst lagringen av medlemmar och båtar på ett smidigt sätt i json-format.

Klass-diagrammet visar en tydlig bild som till största del stämmer överens med den implementerade arkitekturen. Det är ett par relationer som jag tycker saknas, "User" har dependency till "Console" då Console används som argument och "Console" har dependency till "Member" och "Boat" då metoder i klassen har argument med klasserna.

Dessa relationer är lite dolda av sin typ, och dependencyt mellan "Console" och "Member"/"Boat" gör att arkitekturen inte följer Model-View-Controller separations principen, som Martin vill följa. Dock uppfyller den kravet för Model-View separations principen, enligt Larmac som skriver: "The Model-View Separation principle states that model (domain) objects should not have direct knowledge of view (UI) objects, at least as view objects." [Ch 13.7], vilket är bra då det var ett utan grundkraven för workshop 2.

Kravet på skapandet av unika idn till medlemmar uppfylls genom att returnera en sträng från ett "randomly generated UUID", dock är jag inte säker på om det finns någon sannolikhet för duplicering av idn. Generellt tycker jag att implementationen är av mycket god kvalitet gällande kod standarder, namngivning osv.

Designen håller också en hög kvalitet, där jag inte kan hitta något mer fel än de relationer som jag tidigare nämnt. De tre diagrammen i övrigt ser bra ut, det hjälper utvecklaren att förstå relationerna mellan klasserna och en del attribut och metoder inom de olika klasserna. Sekvens diagrammen visar tydligt under en tidsrymd hur dessa use cases exekveras.

Enligt mig tycker jag att både den design och implementation som har skapats uppfyller kraven för betyg 2. Det som jag skulle rekommendera att ändra är att få bort dependencyt mellan "Console" och "Member"/"Boat", vilket gör att MVC principen uppfylls. Och även rätta till relations pilarna i klass-diagrammet där efter. Annars ser det ut som ett super bra jobb som jag blir imponerad på.

1. Larman C., Applying UML and Patterns 3rd Ed, 2005, ISBN: 0131489062