

Missade tyvärr live-föreläsningen från FortNox men har nu kikat på inspelning och skriver här ett referat med lite kommentarer.

Första delen av föreläsningen kändes som en introduktion till varför testdriven utveckling är viktigt. Jonas tog upp exempel på projekt där en testsuite kunde ha besparat mycket arbete och flera buggar kunde undvikits. Han talade om evolution, om människans hjärna och kognitiva beteenden och begränsningar som orsaker till varför vi inte klarar av att hålla ett helt projekts innefattande kod i huvet, fatta korrekta beslut och i slutändan skriva felfri kod från scratch. Han sågade även vattenfallsmetoden och talade om dess negativa konsekvenser och tog upp en del statistik över applikations utvecklingens historia.

Föreläsningens del två hade mer fokus på TDD och därmed testdriven utveckling rent konkret med många exempel. Jonas menade att eftersom man endast skriver kod som är direkt kopplad med ett test och endast skriver tester som är direkt kopplad till ett krav så fungerar testerna även som en dokumentation som förklarar varför man skrivit den kod man skrivit. En av fördelarna med testdriven utveckling enligt Jonas är att, eftersom det är de publika delarna av applikationen som körs genom testerna så kan man i ett senare skede av utvecklingen enkelt ändra på strukturer och design mönster i de privata delarna av applikationen. På det sättet kan man ändra på de viktiga beslut som gjorts i ett tidigt skede som senare behöver förbättras.

Sammanfattningsvis var föreläsningen en hel del repetition av det vi redan lärt oss om agil utveckling och testdriven utveckling men även en nyttig uppdatering från någon som arbetar med dessa metoder och tillvägagångssätt dagligen. Tiden på slutet för att svara på frågor på ett uppriktigt och ärligt sätt uppskattades verkligen. Han visar förståelse för och tycker själv också att TDD kan kännas som att man skriver kod flera gånger och bara gör extra arbete men menar att statistiken för företag som lyckats med dessa metoder talar för sig själv. En viktig poäng som togs upp i slutet är även att många utbildningar lär ut specifika delar separat och att det är viktigt att man som programmerare lär sig att använda allting tillsammans, tex kombinationen av säker, kommenterad och objekt-orienterad kod. Där tycker jag tex att vi under detta första år av webbprogrammerare programmet endast har lärt oss testdriven utveckling i teorin, vi har lärt oss skriva testfall och kravspecifikationer men i praktiken endast skrivit ett fåtal unit-tester i c#-kursen. När vi nu i våra individuella mjukvaruutvecklingsprojekt blir uppmanade att skriva tester för våra applikationer står vi helt utan erfarenhet av hur detta görs i praktiken. Allt som allt var det en bra och sevärd föreläsning, bilderna kunde varit lite mer sammanhängande med ämnet dock.