



Datavetenskap

Opponent(er):

Erik Olsson Haglund & Fredrik Häggbom

Respondent(er):

Mattias Hedman Fallquist & Fredrik Johansson

**Utvärdering av enhetstestning för Palasso på
Logica**

1 Sammanfattat omdöme av examensarbetet

Enhetstester ligger i tiden då systemen blir allt större och mer komplexa. Det är då viktigt att ha bra metoder för att genomföra kontinuerlig testning under utvecklingen, för att förhindra skrotning av projekt.

Arbetet som utförts har varit intressant och tog upp relevanta aspekter för läsaren. Alla målen som sattes upp genomfördes och en slutsats, om än tvetydig, kunde dras.

Tyvärr blir arbetet lidande på grund av mycket språkfel i rapporten, vilket gör den svårläst och öppen för tolkningar.

2 Synpunkter på uppsatsen knuten till examensarbetet

Rapporten beskriver arbetet som har utförts med utvärderingen av testverktyg och hur enhetstestning ska införas i Palasso-systemet, så smärtfritt som möjligt. Strukturen är generellt sett bra, men vi saknar den röda tråden vid vissa tillfällen. Till exempel så förespråkar vi mer grundlig bakgrundsfakta om Palasso-systemet och många av de komponenter som nämns i texten. Det är lätt för en oinsatt att tappa bort sig i läsningen.

2.1 Titel

Vi anser att man bör ha med att Palasso är ett administrationssystem i titeln. Samtidigt är det inte relevant att veta att arbetet utfördes på Logica. Mycket av rapporten behandlar också en utvärdering av de olika ramverken för enhetstestning, detta ska reflekteras i titeln.

2.2 Uppsatsens disposition

Generellt är den bra. Den största bristen i rapporten är bakgrundsinformationen. Vid flera tillfällen så nämns uttryck utan att ha beskrivit dem tidigare. Exempel på detta är Mock-objekt, Palasso, dependency injection, singleton, IDE, XML, Exception, Agila metoder och Waterfall-modellen.

2.3 Begreppsapparat

Ett flertal begrepp används okonsekvent genom hela rapporten. Exempel:

- Uttrycken metod och funktion blandas vid exempelvis fig. 46
- Påhittade svengelska ord används, exempel: fakeas, fakeVariant, castas
- Vid extern källförteckning kan både författarens förnamn och efternamn användas vid ett tillfälle medan det senare i rapporten anges endast efternamnet.
- Samma uttryck används på olika sätt, exempel på dessa uttryck är:
 - Klient/server (Klient-server. Klient/server samt klient server)
 - Komponenttestning och enhetstestning blandas
 - Dataförsörjare (dataförsörjare samt dataproviders)
- Figurtexten är ibland på ovansidan och ibland på undersidan av figurerna/tabellerna.

2.4 Argumentering och slutsatsdragning

Slutsatsen är tvetydig, vi har tolkat deras slutsats efter flera omläsningar som att enhetstester ska införas i hela Palasso-systemet men det ska ske successivt. Om detta är korrekt bör det stå i en konkret mening som inte kan misstolkas.

Vid argumenteringen så upplever vi att motiveringen av varför valen har gjorts är bristfällig. Exempel på det är vid jämförelsen mellan testverktygen (Tabell 2), där motiveringen av urvalet av egenskaper är "... vilka funktioner som vi anser vara viktiga ...".

2.5 Sammanfattningen

Sammanfattningen är relevant och man får som läsare en bra uppfattning om vad rapporten ska handla om. Det som saknas är främst referenser både interna och externa till MVC, Java, klient/server, enhetstest etc.

Vi ställer oss frågande till om resultatet av experimentet ska presenteras i sammanfattningen.

2.6 Språkbehandling

Rapporten innehåller mycket språkfel såsom sårskrivningar, stavfel, talspråk, påhittade uttryck ("svengelska") samt felaktiga meningsuppbyggnader. Tyvärr blir läsandet och förståelsen av rapporten lidande av dessa fel. Vi tror att många av felen, speciellt de felaktiga meningsuppbyggnaderna, hade kunnat åtgärdas om författarna hade korrekturläst rapporten noggrant innan inlämning. Detta beror på att tolkningsfel kan uppstå vid sårskrivningar samt att man inte fick något flyt vid läsandet på grund av meningsuppbyggnaden.

2.7 Referat och källförteckning

Källförteckningen består av relevanta källor från både tryckt facklitteratur och Internet-adresser. Det finns få interna korsreferenserna i dokumentet. Detta får till följd att överblicken vid läsande blir påverkad då det inte blir en klar bild över vart mer information kan hittas.

Bilagan – Saknar titel och är ostrukturerad, det gör den svår att läsa. Som den ser ut nu är den inte relevant, då den ställer mer frågor än svar. Bilagan måste struktureras om samt korrekturläsas ordentligt, då den innehåller mycket språkfel.

Vid utskrift av rapporten eller uppdatering av referenser skapas redundans till källförteckningslistan. T.ex. Scrum[7][7]

2.8 Övriga kommentarer

Vi anser att utvärdering av enhetstester är av stor vikt för företaget och det är en viktig del att ta med sig i framtida utveckling av mjukvara. Generellt är arbetet väl utfört och tar upp relevanta aspekter.

Vi tycker också att det är bra att kod finns med, så att man lättare kan få en uppfattning om hur testet skrivs och exekveras.

3 Genomgång av uppsatsen kapitelvis

3.1 Kapitel 1

Kapitel 1 har titeln ”Översikt”, och innehåller en sammanfattning av rapporten samt dokumentstrukturen. Kapitlet är bra och ger en översikt över vad som gjorts i projektet samt vad rapporten innehåller. Vi saknar dock en del referenser i sammanfattningen, både externa referenser till tekniska uttryck samt interna referenser.

3.2 Kapitel 2

Kapitel 2 tar upp bakgrunden till projektet. Där beskrivs de olika testerna som utförs, hur de vävs in i utvecklingsmetoder samt hur test används på Logica idag.

De externa referenser som finns i kapitel 2.1 bör istället vara interna referenser till stycke 2.1.1 och stycke 2.2.2. De externa referenserna bör istället finnas i dessa två stycken.

I detta kapitel saknar vi mer beskrivning av termer som används genom rapporten. Vi saknar bland annat beskrivningar av mock-objekt, Palasso, dependency injection, Agila metoder samt Waterfall-modellen. Dessa uttryck nämns upprepade gånger i rapporten utan att ha någon förklaring av uttrycken. Antaganden har gjorts med ”Traditionell utvecklingsprocess” är waterfall-metoden. Om så är fallet, bör ”Traditionell” bytas ut mot ”Waterfall”.

Vi skulle också vilja ha en översiktsbild över hur de olika testnivåerna förhåller sig till varandra (komponent-, integration-, system- samt acceptanstestning).

3.3 Kapitel 3

Kapitel 3 innehåller information om den förstudie som gjordes i projektet. I detta kapitel presenteras och jämförs fyra olika testverktyg utifrån de krav som ställs på testverktyget i detta fall. Ett av testverktygen väljs sedan ut och används i övriga delar av projektet.

Två jämförelser görs, där egenskaperna i den första jämförelsen motiveras. I den andra jämförelsen saknas dock motivering till varför egenskaperna valts.

Det står också att TestNG kan exekvera test parallellt, men ingen förklaring ges till hur det sker.

I stycke 3.4 nämns setup- samt teardown-funktioner utan att ge referens eller förklaring till vad det är. Även i stycke 3.5 saknas referenser till uttryck som inte nämnts tidigare, t.ex. EasyMock, Spring, Hibernate, Java Persistence API.

3.4 Kapitel 4

I detta kapitel ges en översikt av experimentet samt en kort beskrivning av Palasso-systemet. Med tanke på att Palasso-systemet är en så stor del av rapporten bör beskrivningen vara mer utförlig så att läsaren får en bra uppfattning av vad Palasso är, samt vad det kan användas till.

Vi anser att Figur 8 inte är helt korrekt då vyn och modellen bör ha en egen relation, alltså inte enbart via kontrollern.

Figur 9 och 10 ska redigeras så att texten i rutorna hamnar på samma rad.

3.5 Kapitel 5

Kapitel 5 tar upp metod 1 av experimentet. I denna del visas hur programvaruutveckling kan ske med hjälp av TDD och enhetstestning för att säkerhetsställa kodkvalitén samt hur utvecklarna av Palasso kan använda enhetstestning.

Vi saknar en motivering till varför modulen "Organisation" valdes. Den motivering som ges nu, är att det är "den enklaste". På vilket sätt är det den enklaste modulen?

Butler nämns, men ingen tydlig förklaring finns.

I stycke 5.2.2 tas fördelarna med klient/server upp, inte nackdelar.

Figurerna och kodexemplen som finns i detta kapitel är relevanta och ökar förståelsen för metoden.

3.6 Kapitel 6

Kapitel 6 tar upp metod 2 av experimentet. Här tas metoden för att skriva test på redan skriven kod upp.

I Figur 41 har pilarna bytt riktning i jämfört med andra figurer. Vi antar att pilarna är felaktigt riktade i denna figur och bör därför göras om. I övrigt är detta kapitel bra skrivet.

Figureerna och kodexemplen som finns i detta kapitel är relevanta och ökar förståelsen för metoden.

3.7 Kapitel 7

I kapitel 7 ges en diskussion till experimentet. I det första stycket diskuteras det valda ramverket, TestNG. I de två nästkommande styckena finns en diskussion för de två olika metoderna som användes i experimentet.

Detta kapitel var bra, relevant och tog upp det viktigaste som gjorts tidigare i projektet.

I stycke 7.3.3 står att den enskilde betraktaren kan göra tolkningar om anledningen till tidsförbättringarna i samband med införandet av enhetstester. Vi anser att författarnas åsikt bör anges här.

3.8 Kapitel 8

I detta kapitel finns en slutsats för experimentet.

Vi anser att slutsatsen i denna rapport var något abstrakt, då den inte ger något konkret svar på frågan om Logica rekommenderas att införa tester på Palasso eller inte. I samband med att positiva aspekter anges bör också de negativa aspekter som finns anges.

3.9 Övriga kommentarer

Mängden språkfel gör att vi inte skriver en lista med synpunkter utan bifogar istället rapporten utskrivna med kommentarer för rättningen.

4 Slutliga kommentarer

Arbetet är väl utfört och vi tror att Logica kommer att få nytta av resultatet som har visats. Bilderna samt koden som finns i rapporten underlättar läsningen samt förståelsen. Rapporten behöver dock justeras språkmässigt innan inlämning.

Framsidan bör bytas ut till den mall som tillhandahålls av Karlstads Universitet.