

Halvtidsrapport

Niclas Olofsson - 1 april 2014

Statusrapport

Examensarbetet har hittills gått i stort sett enligt planeringen i den tidigare inskickade planeringsrapporten. Arbetet inleddes med att sätta sig in i uppdragsgivarens nuvarande applikation. Ett möte med en av uppdragsgivarens kunder hölls för att ta reda på vilken typ av ny funktionalitet som efterfrågades. Detta efterföljdes av en litteraturstudie, där flera olika aspekter av testning samt automatisk generering av sådana studerades. Delar av slutrapportens teoridel påbörjades och skrevs samtidigt som litteraturstudien pågick. En testimplementation för automatisk generering av tester påbörjades.

Arbetet övergick sedan i en fallstudie, som inleddes med att implementera nya samt komplettera de befintliga, till stor del icke-fungerande, tester som fanns i applikationen. Detta gav dels en bra aspekt på testning, men var även en förutsättning då testdriven utveckling skulle komma att användas i examensarbetets senare implementationsdelar, vilket verkade kunna bli ganska jobbigt om vissa tester inte ens gick att köra.

En systemdesign av den nya funktionaliteten gjordes sedan. Denna diskuterades igenom med uppdragsgivarens handledare och implementation av den nya funktionaliteten påbörjades. Implementationsfasen har hittills gått överlag bra. En del tid ägnades i början åt att lära sig Rails, KnockoutJS samt testramverken RSpec och Jasmine, men det känns som att jag nu fått såpass mycket erfarenhet av dessa ramverk att utvecklandet börjar gå smidigt.

Förändringar jämfört med planering

Den största förändringen som gjorts jämfört med planeringsrapporten är att den planerade delen om automatisk generering av tester har tagits bort. Efter diskussion med handledare kom vi fram till att den problemformulering som var aktuell i planeringsstadiet behövde bli mer koncis och mindre spretig. Således borde problemformuleringen fokusera på antingen automatisk generering av tester, eller på hur testning kan göras i uppdragsgivarens webbapplikation. Valet kom att falla på det sistnämnda, varvid målet att skapa en testimplementation av en programvara för

automatisk generering av tester inte längre var relevant.

Implementationsfasen har kommit längre än planerat, dels eftersom att arbetet gått fortare än väntat, men även eftersom att implementationen av ett verktyg för automatisk generering av tester uteblev. Om implementationsfasen blir klar tidigt är tanken att börja skriva tester för fler delar av uppdragsgivarens applikation, för att få mer erfarenhet av att skriva tester samt se hur TDD kan tillämpas i detta sammanhang.

Utvärdering av planerade halvtidsmål

A test implementation of a software to derive lower-level tests from high-level tests should be done. Alternatively, parts of the report explaining why such software would be implausible to build within the scope of this project should be accomplished.

Detta mål är inte längre aktuellt och har därför inte uppfyllts helt, se diskussion ovan.

Draft of report parts regarding web application testing theory should be completed.

Ett utkast till slutrapport finns bifogat. Teoridelen är inte helt färdig, vilket dock inte heller var målet. Jag tycker att teoridelen är ungefär så färdig som jag planerade.

A plan for the implementation of the new system functionality should be finished.

Designdokument finns bifogat. Implementationsfasen har kommit längre än planerat.

Framtid

Implementationsfasen kommer att pågå ytterligare ett par veckor. Därefter kommer mycket arbete att läggas åt att färdigställa fler delar av slutrapporten. Arbetet kommer sedan att övergå till att utöka befintliga delar av uppdragsgivarens system med tester i den mån tid finns, samt utvärdera dessa med code coverage. Framläggning i början av juni känns som ett rimligt mål.

Frågeställning

Den huvudsakliga frågeställningen för examensarbetet är att undersöka hur automatiserad testning kan tillämpas i uppdragsgivarens applikation på ett bra sätt, med syfte att förhindra buggar i produktionskoden. Mer resonemang om detta återfinns i utkastet till slutrapport.

Referenser

En fullständig referenslista återfinns i utkastet till slutrapport. Nedan listas de viktigaste litterära källorna. Som komplement till litterära källor har bland annat även videor från utvecklarkonferenser använts.

- Kent Beck. Test-Driven Development by Example. Addison-Wesley Longman, 2002. ISBN 9780321146533.
- Dorota Huizinga and Adam Kolawa. Automated defect prevention: best practices in software management. Wiley-Interscience, IEEE Computer Society, 2007. ISBN 9780470042120.
- Shari Lawrence Pfleeger and Joanne M. Atlee. Software engineering: theory and practice (international edition). Pearson, 2010. ISBN 9780138141813.