

DETTA ÄR ETT FÖRSÄTTSBLAD

Projektnamn	Kontinuerlig leverans av system under test
Beställare	Sigma

Sammanfattning

Här görs en sammanfattning av dokumentet som ska rymmas på denna sida.

Revisionshistorik

Datum	Revision	Anmärkning	Signatur
12-04-02	1	Rapport skapad	ML

Innehållsförteckning

Inledning	3
Bakgrund.....	3
Syfte.....	3
Projektmål.....	3
Teori.....	4
Metod.....	4
Genomförande.....	4
Informationsinsamling.....	4
Lösningsförslagen.....	4
Historik.....	4
Problem.....	4
Lösningen (Resultat).....	4
Uppfyllnad.....	4
Måluppfyllnad och förväntningar.....	4
Tidutfall mot plan.....	4
Funktion/kvalitet mot kravspecifikation.....	4
Acceptans av projektet.....	4
Rekommendationer.....	4
Lärdomar, erfarenheter och slutsatser.....	5
Personliga reflektioner.....	5
Slutsatser.....	5
Referenser.....	5
Bilagor.....	5

Inledning

Bakgrund

Kontinuerlig leverans

Att vid utvecklingen av en produkt regelbundet bygga hela systemet för verifiering av funktionalitet är ofta önskvärt. Utvecklare får snabb feedback på de delar som inte integrerats korrekt med övriga systemet och kan då rätta till detta direkt. Om systemet fungerar korrekt fortsätter ofta utvecklingen och med korta intervaller sätts hela systemet ihop för testning av funktionalitet.

Ett positivt resultat av testningen av hela systemet bör innebära att teamet har en fullt fungerande produkt. Ändå är det ofta med mycket långt intervall som produkten levereras till kund så att denne kan ta del av ny funktionalitet. Tanken med kontinuerlig leverans är att inte bara testa ny funktionalitet regelbundet utan även leverera ny funktionalitet regelbundet. [1]

Exekverbara specifikationer

Det språk som används i en kravspecifikation är talspråk och definierar ofta en produkt utan

konkreta exempel. Det finns inte heller något som säger att det som är angivet verkligen uppfylls av produkten. Med exekverbara specifikationer minskar gapet mellan beställarens och utvecklingsteamets språk och de specifikationer som är angivna består av konkreta exempel som kan förstås av beställaren och exekveras av utvecklaren.

Syfte

Syftet med det här projektet är att ta ett helhetsgrepp om utvecklingen av en webapplikation, från idé till slutanvändare. Vid projektets slut ska en modell finnas för kontinuerlig leverans av en automatiskt testad produkt till molntjänst.

Teori

Git

Git är ett versionskontrollsystem som använts i detta projekt på grund av den utbredda användningen samt kopplingen till community GitHub(se nedan).

GitHub

GitHub är ett community där utvecklare kan ladda upp projekt utvecklade med Git revisionskontrollsystem.

Selenium

Selenium är ett verktyg för analys av innehållet på en hemsida. Givet namn eller css-id kan värden på attribut erhållas vilka sedan kan användas för testning.

Cucumber

Cucumber erbjuder ett ramverk för högnivåtest genom reguljära uttryck. Uttrycken har formen:

1. Givet att - följt av angivna värden av användaren."
2. När – följt av operationen användaren utfört när testet ska utvärderas.
3. Då – följt av ett booleanskt uttryck som är det uttryck som ska utvärderas.

Java

JSF

JUnit

Molntjänst

Heroku

Heroku är en molntjänst integrerad med git för kontinuerlig leverans.

Mockito

Metod

Genomförande

Till en början var lärokurvan brant. Första veckan ägnades främst åt informationsinhämtning i syfte att skapa större förståelse för projektet.

Informationsinsamling

Informationsinhämtningen bedrevs genom sökningar på de verktyg som presenterats av handledare.

Lösningförslagen

Om det fanns alternativa lösningar, beskriv dessa och hur man valde.

Historik

Lista de viktigaste händelserna under projektets gång.

Problem

Resultat

Version angiven i pomen viktigt vid molntjänst
HERoku fungerar för jsf-applikationer?

Uppfyllnad

Måluppfyllnad och förväntningar

Funktion/kvalitet mot kravspecifikation

Acceptans av projektet

Beskriv acceptanstestets utfall, tillvägagångssätt, när det genomfördes och vem som var kundens representant.

Rekommendationer

Förslag på förbättringar och ärenden som kan vara intressanta i framtiden.

Lärdomar, erfarenheter och slutsatser

Detta är ett mycket viktigt avsnitt. Vad har vi lärt oss under projektet – positivt och negativt.

Personliga reflektioner

Slutsatser

Ett sammanfattande avsnitt som bygger på problemformuleringen, resultaten och erfarenheter. Glöm inte bort att komma fram till en slutsats om era lärandemål – har de uppfyllts?

Referenser

[1] Conitnues Delivery Jez Humble and David Farley.

Bilagor

Dokument, diagram, etc. som inte ”platsar” i den löpande texten i rapporten. T.ex. kravspecifikationen, avtal, o dylikt.