

Sammanfattning

I Karlstad utvecklas det marknadsledande löne- och PA-stytemet för den statliga sektorn Palasso. För att få så hög kvalitet på Palasso som möjligt genomförs omfattande testning. Testning genomförs för att säkerställa att mjukvaran gör det den ska göra. Dock har testningen varit på en högre nivå. För att fortsätta utvecklas har gruppen som utvecklar Palasso börjat att snegla på de nya utvecklingsmetoderna som har blivit populära inom dataindustrin de senaste åren. Dessa utvecklingsmetoder bygger på att använda sig av test på så låg nivå som möjligt, så kallade enhetstest.

För att undersöka möjligheterna för Palasso att använda enhetstest, har två olika delar undersökts av detta arbete. Först genomfördes en studie av de befintliga ramverk som finns för enhetstest. Därefter valdes det ramverk som var bäst lämpad för Palasso. Därefter genomfördes en experimentdel för att se om det var möjligt att införa enhetstest på Palasso. Detta gjordes med det valda testramverket. Denna del delades upp i två metoder. I metod 1 togs en modul av Palasso och skrevs om för att visa på hur koden måste vara uppbyggd för att enhetstestning ska kunna införas. I metod 2 infördes enhetstest på befintlig kod för att bevisa att det var möjligt att införa enhetstest och att små ändringar gör koden mer testbar.

Det ramverk som valdes till det bäst lämpade för Palasso blev TestNG. Detta för att det var lätt att använda och att det var skalbart. Genom experimentdelen kunde slutsatsen dras att det var möjligt att införa enhetstest på Palasso. Det ska dock införas i samband med nyutveckling eller när ändringar av den befintliga koden ska genomföras. Införandet av enhetstest ger många fördelar därför är slutsatsen att det vore Palassosystemets kvalitet till gagn att använda sig av enhetstest.

Abstract

The market-leading wage and PA-system for the Swedish public sector Palasso is developed in Karlstad. In order to achieve the highest quality of code , the developing team behind Palasso has been conducting extensive testing. Testing is done to ensure that the software does what it should do. However, the testing has been done at a high level. In order to continue to evolve, the group that develops Palasso has started to look at the new development methods that have become popular in the computer industry in recent years. These methods are based on the use of tests as low as possible so-called unit tests.

To examine the possibility for Palasso to use unit testing, the study has been divided into two parts. First, a study of existing frameworks to find the best suited one for Palasso. Then, by using the selected framework, a second study is conducted to see if it is possible integrate unit testing with the Palasso system. This part was divided into two methods.

In the first method a existing functionality was redeveloped to show how the code should be structured and implemented to be suitable for unit tests. In the second method unit testing was introduced on legacy code to prove that it was possible to introduce unit test and that small changes can make the code more testable.

The best suited framework for Palasso was TestNG. TestNG was chosen because it was easy to use and it had scalability. The second study showed that it is possible to integrate unit tests with the Palasso system. But it should be introduced when new modules are developed or when existing code is modified. The introduction of unit testing provides many advantages. It is concluded that the quality of the Palassos system would benefit from using unit testing.