

Executable documentation

Wojciech Biela¹ and Krzysztof Jelski²

¹ ExOrigo Sp. z o.o., Krucza 50, 00-025 Warsaw, POLAND
Wojciech.Biela@exorigo.pl, <http://www.biela.pl>

² ExOrigo Sp. z o.o., Krucza 50, 00-025 Warsaw, POLAND
Krzysztof.Jelski@exorigo.pl

Abstract. Most of the effort is put into *making the software right*. Experience shows that much less consideration and effort is put into *making the right software*. This differentiation is even more accurate and important if we look at software engineering from the business side of the fence. In this presentation the authors, basing on their own as well as others' experience, will talk about an approach to maximise the chance of writing the right software. This may be accomplished by specifying acceptance tests early in the development process, thus using them as requirements documents and by engaging the client into the whole process of specifying and using this documentation in various ways. Automated acceptance tests also do a good job of documenting the system under development and allow for fast regression tests. The presentation will be divided into three parts: introduction to the field of acceptance tests, setting-up a project around acceptance tests, and a practical presentation of code examples. We would like the second and third part to be filled with questions from the audience (and answers) and a lively discussion around the topics we will be touching. The practical part will contain both a presentation of a complex example we have developed using one of the techniques and a presentation of other tools available and our experiences with them. In this presentation we will also address the hot topic GUI vs non-GUI tests as well as the controversial agile documentation issues.

References

1. Mugridge, R., Cunningham, W.: Fit for Developing Software: Framework for Integrated Tests Prentice Hall (2005)
2. Crispin, L.: Testing Extreme Programming. Addison-Wesley (2002)
3. Astels, D.: Test-Driven Development - A Practical Guide. Prentice Hall (2003)
4. Hunt, A., Thomas, D.: The Pragmatic Programmer: From Journeyman to Master. Addison-Wesley (1999)