

# A twenty one-month agile prototype

Hugo Corbucci

## Abstract

Prototyping is a well known and widely accepted technique to quickly present possible solutions to the stakeholders and, therefore, help them make a more conscious choice between the possibilities.

Agility is strongly associated with maintaining high quality all the time. To achieve such goal, it is necessary to invest a reasonable amount of time to polish the project's source code, its development environment and its own development process.

What happens when one tries to mix both approaches? This report presents a project that started as a simple series of quick prototypes which evolved during twenty one months to a full featured product. It shows how this project tried to keep the balance between developing prototyped incomplete features and keeping the quality of the code to allow for further changes.

## 1 Introduction

With the growth of investments in software development, it is not uncommon to encounter companies not related to software development that want to build better tools for their business. Usually those companies are not confident that the investment is good and wish to invest as little as possible to understand whether the project might be successful or not.

Due to the agile principle of delivering working software frequently and the quest to quickly understand if a project is going to fail, agile processes are very good candidates to develop those spike projects. If the spike is successful, clients expect to evolve the spike project to a full featured project without starting it all over.

On a prototyping development, adding spike features (ignoring several known issues) and improving code quality to ensure a decent user experience are antagonistic forces.

Investing in the code quality only repays after some time when maintenance is a considerable part of the work. However, such investment is useless if the project is aborted when the stakeholders discover the project will not be lucrative. In such situation, stakeholders pressure for new spike features. On the other hand, developers feel that investment in quality is absolutely essential but have trouble justifying it because maintenance might never be necessary.

Once the project moves from spike/prototype project to a full feature project, developers and stakeholders join to improve quality and ensure all corner cases are covered. However such transition is usually not very well delimited and the mindset change for the stakeholders is quite fuzzy. Not to mention that if the first balance is not fine tuned, the project might never reach the transition stage. This

report presents one case where the project came close to the limit of complexity but managed to obtain the quality investment and moved to a full featured project after eighteen months.

## **2 Proposed report structure**

1. Starting a prototype project
2. Producing faulty features consciously
3. Changing the prototype's objective
4. Increasing complexity
5. Moving to a full featured project
6. Decreasing complexity and improving quality

## **3 About the author**

My name is Hugo Corbucci and I am a masters student at the University of São Paulo (USP) in Brazil currently working to delineate the proximity between open source software development and agile software development. I am also one of the founders of Agilbits, a three years old software development company in São Paulo, Brazil.

I can be contacted through my email at [hugo@agilbits.com.br](mailto:hugo@agilbits.com.br), mobile phone at +55 11 8636 9029 or through post at:

Rua Trajano Reis, 185 - Ap 94 Bl 4  
Jardim das Vertentes - São Paulo - SP  
CEP: 05541-030  
Brazil

I was first in touch with agile software development in 2005 and have kept my interest ever since. From 2006 to 2009 I was involved in the extreme programming laboratory course at USP first as a student then as a teaching assistant. In 2007, I founded Agilbits with four colleagues and a year later we got involved with the mentioned project with a four people team keeping three to four weeks releases but with hourly builds.

In July 2007, I co-founded the Coding Dojo São Paulo and have been maintaining it ever since. Such work resulted in an experience report published at Agile 2008 which I attended as well as Agile 2009. I am also involved in the main agile event in São Paulo and I am working with the Brazilian community to organise a national event in 2010.