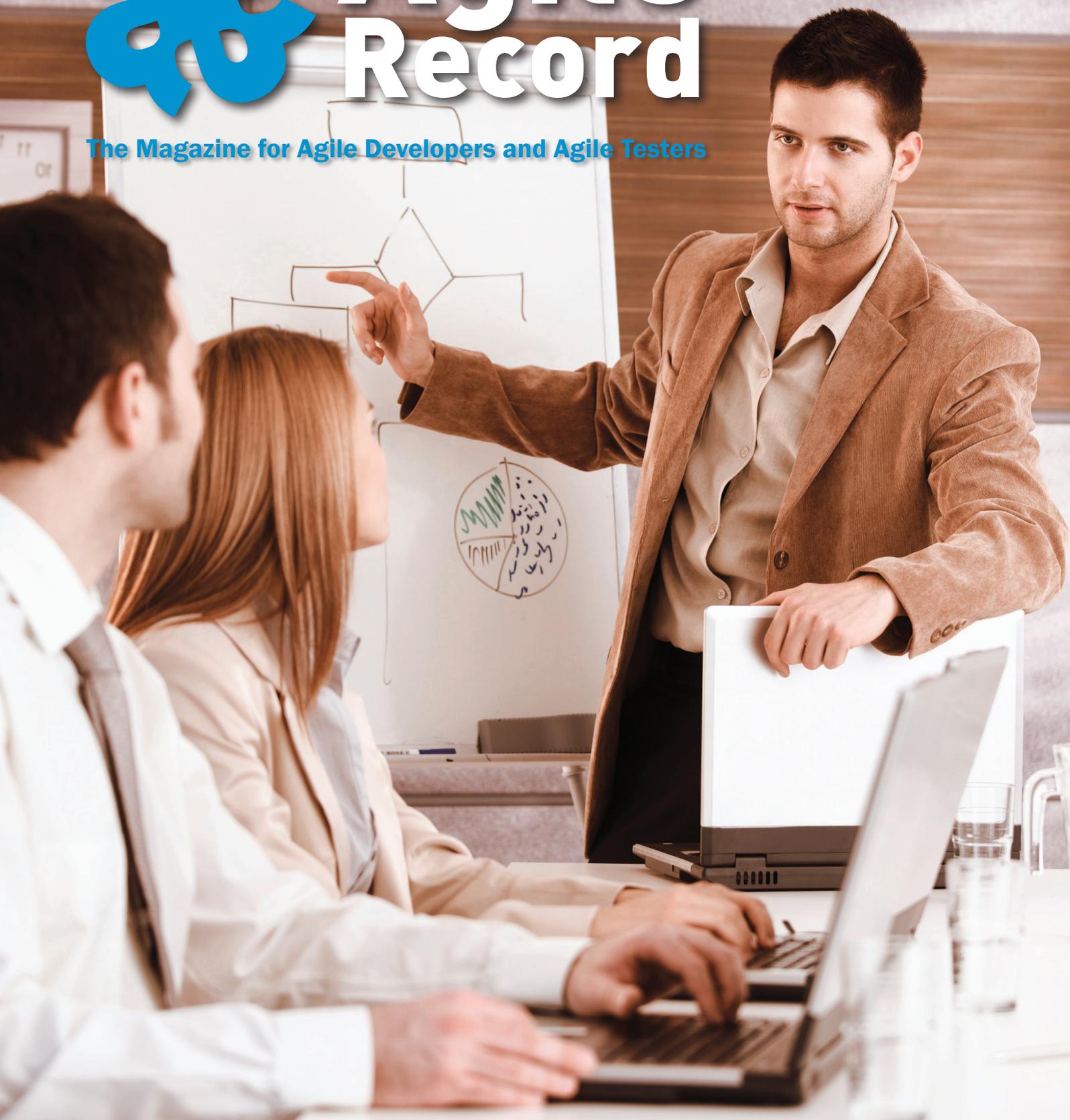


Agile Record

The Magazine for Agile Developers and Agile Testers



Collaboration and Building Teams

www.agilerecord.com

free digital version

made in Germany

ISSN 2191-1320

November 2012

issue 12



Column

Software development is all about people

by Huib Schoots

The theme of this agile record is an interesting one. To me collaboration and great teams are the most important aspects of working. Agile or non-agile, collaborating makes things easier and often more fun! Team building is improving the performance of a team. Interesting stuff, but how does this work? How can we make successful teams and make them work collaboratively? I have read a lot about this topic and I have developed my own opinion about it as well. To me technical skills or knowledge do not matter in building great teams. Collaboration does! But how do you make teams collaborate better?

Steve Jobs

In Jonah Lehrer's book "Imagine, how creativity works" the author explains how collaboration helped Pixar. When Steve Jobs worked at Pixar he needed people to collaborate more. Pixar wanted the computer guys to work closely together with the animators. So Steve Jobs decided to put everybody in the same building. But he took this even further: he insisted that there would be only one group of bathrooms for the whole studio. He forced people to run into each other, even if it was in the bathrooms. Jobs needed people to mix, he knew that mixing skills makes better movies and the natural tendency is to stay isolated and only talk to people that are just like you. And it worked! The result was instant. The communication improved and with that the collaboration became much better.

Daniel Pink

In his book "Drive" management guru Daniel Pink describes three success factors for teams. He mentions: autonomy, purpose and mastery. Autonomy gives the team the freedom and the tools to do what they need to do. Purpose helps the team feel that they are making a difference and they are important. This will feed their motivation and dedication. And last but not least mastery, the will of the team to be really good at what they do. If you have the will to excel, your chances of succeeding increase.

Challenge, passion and room to excel

People who want to excel need two important things: a challenge and passion. Vineet Nayar says in his article "The Key Ingredients of a Successful Team" (source: <http://blogs.hbr.org/hbr/nayar/2012/06/three-things-that-drive-teams.html>) that it is about the journey, not the end result. When teams face huge challenges, finding solutions becomes exciting, sometimes even obsessive. The passion drives the team to go on, infects others and works as a snowball to get the team to the next level. The last essential

characteristic for successful teams is room to experiment: real innovation and improvement comes from trying new things.

These factors resemble the ones described by Daniel Pink: autonomy makes room to excel, passion feeds mastery and a challenge gives purpose.

My view on collaboration and building teams

Teams working together are most important in achieving goals. I remember when Feyenoord, my favorite soccer team in the Netherlands, won the UEFA cup in 2002 with a team that wasn't the best in the UEFA cup competition. They had just average players compared to other teams in Europe. But they were a real team and the whole team was dedicated and committed to win. Feyenoord won, not because of the individual skills, but because they were a team, the collaboration and dedication made the difference.

But I dare to take this even further: give me a team of people that are really willing to make something work and we will do the job, any job! September 10th I visited a talk about specification by example by Gojko Adzic at the Federation of Agile Testers in the Netherlands (FAT-NL) and during the drinks we discussed what is needed to excel in agile. I stated that DEWT (11 fellow software testers who get together regularly in a peer workshop to learn from each other) would be able to build let's say a salary system within reasonable timelines if we were asked to. Probably just a couple of us can code a bit but still I believe we will make it work. And if needed we will hire a real developer to do the really complicated stuff. Why do I believe that we will pull this off? Because we are dedicated, perseverant, willing to learn, willing to help each other and also able to work together. We know each other quite well and we will give each other room to make mistakes. That will make the perfect team! The dedication and perseverance will help us make up for the lack of coding skills and knowledge. We will learn fast and make it work.

Successful teams

If you google 'successful teams' or 'building teams' you will find many lists with team characteristics that make a team successful. The lists differ quite a bit and that is interesting. This might have something to do with the context in which the teams work. Three aspects are in every list: clear team goals, clear communication and defined responsibilities. Some others popup often but not in

every list, like balanced participation, trust, participative leadership, effective decision making and helping each other.

What I miss in a lot of these lists is passion and perseverance. During the “all things digital conference” in 2007 Steve Jobs says to be successful you need to have passion for what you do. To be successful is often very hard and to sustain that over a period of time, you really have to work hard and love what you do to persevere. Normal people, he says, give up. Successful people persevere.

Building teams

It takes time to build successful teams. At CAST Test Coach Camp in San Jose last summer Michael Larsen (link: <http://www.mkltesthead.com/2012/07/day-2-of-test-coach-camp-live-blog.html>) introduced the Tuckman stages of team development to me which were published in 1965: Forming, Storming, Norming and Performing. Every team has to go through these stages, which takes time and in the end only a few reach the performance stage.

In the forming stage (why are we here?) the team forms itself but exists of individual members getting to know each other and finding their places. In the storming stage (can we work together?) the members have to find ways to work together. Conflicts between the members arise about things like the mission and the approach. In the norming stage (how will we work together?) issues get resolved. All members take the responsibility and have the ambition to work for success. In the performing stage (how can we work smarter?) the team becomes a high-performing team. They are accomplishing the work smoothly and effectively without inappropriate conflict or the need for external supervision. In 1977 Tuckman added an extra stage: adjourning, in which the project ends and the group is abandoned.

High-performance teams?

So how many high-performance teams are there anyway? According to Steve Demming, high-performance teams constitute only 2% of all teams in the workplace. I think there could be much more if management would understand that the biggest challenge of software development is not about technique but about humans! Building successful teams is all about people! Teams will succeed if the people trust each other, are persistent and have passion and the willingness to help each other. This will lead to great collaboration. Once you have built a great team, their technical skills or knowledge do not matter that much anymore...

> about the author



Huib Schoots

Huib has 15 years experience in IT and software testing. After studying Business Informatics he became a developer. Soon he discovered that development was not his cup of tea and software testing is fun.

Huib has experience in various roles such as tester, test coordinator, test manager, trainer, coach, but also in project management. He is currently agile test consultant at codecentric. He tries to share his passion for testing with others through coaching, training and giving presentations on different test subjects.

Huib sees himself as a context-driven tester. He is curious, passionate and has (unsuccessful) attempted to read everything published on software testing ever written. He is a board member of TestNet, the association of testers in the Netherlands. He is a member of DEWT (Dutch Exploratory Workshop on Testing), student in the Miagi-Do School of Software Testing and maintains a blog on magnifiant.com. He is co-author of the recently published TestNet jubilee book about the future of software testing.

SWE Guild

The Software Engineering Guild

www.sweguild.com