



Agile Record

The Magazine for Agile Developers and Agile Testers

Management and Leadership in Software Organisations

www.agilerecord.com

free digital version

made in Germany

ISSN 2191-1320

May 2012

issue 10



Risk Poker: Risk based testing in agile projects

by *Jurian van de Laar*

About risk based testing

Testing is an important supporting activity to achieve ‘working software’ in agile projects. By finding defects and providing insight in the quality of the software, testing plays a role in providing feedback to software developers, in satisfying acceptance criteria and in adhering to the Definition of “Done”.

Testing activities can be seen as mitigating product risk. A product risk is defined as a risk which is directly related to a potentially failing product. Risk based testing is an approach for developing and prioritizing tests based upon the impact and likelihood of failure of the functionality to be tested. Likelihood is the chance that the software contains defects (caused by for example poor programming, high complexity, etc.). Impact is an indication of the consequences when the software fails.

Risk based testing with “Risk Poker” in agile projects

One of the most frequently asked questions about testing, both in traditional and Agile projects, is: “How much testing should be done”? In some traditional projects managers may want the team to ‘test everything’. They want to be absolutely sure that the system is completely tested before it is released into the market, to prevent problems – or even claims – in production. However, testing the entire system in every possible way is impossible. No organization is willing to spend sufficient resources for ‘exhaustive testing’ and pressure on budget and release schedule will not allow for the required effort.

James Bach, leading proponent of Exploratory Testing, introduced the concept of ‘good enough testing’ in 1997. This concept is helpful in understanding the risk based testing approach. Agile projects are usually not striving to develop ‘the absolute perfect software’. The concept of ‘a potentially useful version of working product’ in Scrum actually means that the software is working ‘good enough’ to take it into production. ‘Good enough’ in this context is defined as: providing sufficient benefits, having no critical problems and the benefits of releasing now outweigh both the consequences of

non-critical problems and delaying the project for further testing.

Risk Poker is an approach for product risk based testing in agile projects. The process of Risk Poker is similar to the way that Planning Poker is done, e.g. in Scrum, except that it will result in risk identification and risk analysis rather than estimations and story points.

What are the reasons for applying risk based testing with Risk Poker in agile projects – and what are the benefits?

1. Most agile methods and frameworks – like Scrum – are time-boxed. Iterations have a fixed duration, so both development and testing activities are by definition limited to a pre-defined timeframe. Risk based testing provides an excellent answer to the problem ‘how much testing’ by ensuring that the most important testing has been done within the available time. Therefore risk based testing is a very suitable approach for time-boxed development methods.
2. Just like Planning Poker, Risk Poker is a team-based activity and decisions are made by achieving consensus.
3. In the Scrum process, Risk Poker can be easily combined with Planning Poker in the Planning meeting. They complement each other, because information about business value from the Product Owner (PO) will provide input for the impact component of product risk, and the question-and-answer game about product risks will be input for estimating the testing effort in Planning Poker.
4. User stories are very suitable entities to be used as ‘risk items’ in a product risk analysis. In agile projects, risk identification comes down to identifying user stories.
5. Agile is all about ‘working software’, Risk Poker is a lightweight approach to achieve the most appropriate balance between sufficient quality and acceptable risk – within the available constraints in time and resources.

In our practice, working in agile projects, we discovered that the 'original' product risk management approach as we had described it in our white paper in 2006 required some modifications to better fit to an agile way of working. We needed a 'lightweight' variant, one that could be easily integrated with Risk Poker in the planning meeting and sufficiently intuitive for the team to quickly understand it and get fast results. As an example, Risk Poker uses 'traffic-light'-colored 'poker cards' instead of the conventional quantitative approach using risk factors.

Risk Poker: How does it work?

Traditional risk assessment starts with identifying product risks. In Scrum, the backlog items (e.g. user stories) are considered risk items. So the question for each backlog item will be: "What risk is associated with this user story?" This question is addressed in the conversation about the user story during the sprint planning session.

Prior to the planning meeting, the team should determine which factors influence the quality of the delivered software. Typical factors for likelihood are complexity, new development (level of re-uses), interrelations (number of interfaces), size, technology and (in-)experience of team. The team itself decides which factors for likelihood are to be taken into account during the Risk Poker.

Concerning the impact, influencing factors can be business importance (i.e. selling item), financial damage, usage intensity, external visibility and legal sanctions. The PO will decide (together with stakeholders) which of these factors for impact are to be taken into account. Risk Poker is ideally played prior to Planning Poker. This is because the risk determined for each item should determine the test effort involved in every backlog item. It is a consensus-based technique for estimating risk.

During the planning meeting, next to their planning poker cards, every estimator is given a set of risk poker cards, which contains 4 colored cards (green, yellow, orange and red). The colors represent the perceived risk level (both for likelihood and impact). The meaning of the colors is similar to story points: 'red' for the highest risk level, 'green' for the lowest risk level. First the moderator or the PO provides an overview of the User Story (US). The team is given the opportunity to ask questions, discuss the US and bring all issues concerning risk into play. While discussing, colors must not be mentioned, since this can influence every individual team member. When all is clear for the Scrum team, every team member will choose the colored card they think represents the risk (for likelihood of failure) best. Everyone shows their card simultaneously by throwing them on the table. When estimated risk colors are not all identical, team members with different colors are asked to clarify their choice of color. When all estimators have explained their color, the risk estimation process is repeated. If all thrown colors are still not equal, the PO (or moderator) can ask the estimators again for clarification and repeat the process, or the PO can decide which color is picked (usually the color representing the highest risk). Concerning the impact, the PO will give a color representing this risk (impact component) best and explains this to the team. Because the PO role in Scrum is typically responsible

for both prioritizing user stories and assigning business value, the impact component of product risk is determined by the PO solely without striving for consensus by playing the risk poker cards with the team. The explanation to the team, however, is important to get the team's support, and the team is also encouraged and challenged to ask questions. After Risk Poker has been played for a US, the Planning Poker can take place for that same US following its own rules.

Applying the risk based approach

When putting the User Stories to the Scrum board, the colors for risk likelihood and impact should be made visible together with the total number of points.

User Story 11

As a *Forum User* I want to *Upload photos* so that *I can share these photos with other forum members*

Points: 8

Risk Likelihood: Orange

Risk Impact: Green

The colors chosen represent a certain thoroughness of testing. Red means very thorough testing, whereas green means less (or maybe even no) testing. To define this thoroughness, a wide variety of actions can be taken. One can think of code reviews, unit tests (e.g. level of code coverage), acceptance tests, regression tests, the way testers are involved in testing items and the usage of testing techniques. For every combination of colors, a set of actions is defined. The measures become stricter if the colors are indicating more risk. Some examples:

- Likelihood is green and impact is yellow: Unit test with decision coverage (e.g. 70%) and acceptance test with use case technique (only basic flow). Tests may be created and executed by every team member.
- Likelihood is orange and impact is green: Unit test with condition determination coverage (e.g. 80%) and acceptance test with only exploratory testing.
- Likelihood and impact are both red: Code reviews, unit test with multiple condition coverage (e.g. 80%) and acceptance test with use case technique (basic, alternative and exceptional flow). Tests must be created and executed by test team member.

Risk likelihood is more or less coupled with unit testing, whereas the risk impact has a close relation with acceptance testing.

Differences with traditional product risk analysis

There are some differences with Risk Poker compared to a 'normal' risk assessment. For instance, the PO will represent all stakeholders and should have good knowledge of the US, so the PO can judge the impact of the risk.

Risk Poker is only useful for user stories which are to be picked up in the upcoming sprint.

The approach of risk poker is less detailed than in the traditional way, by using 4 colors instead of calculations with numeric scores and weighting factors.

Alternative

The set of risk cards can either exist of 3 (green, orange, red) or 4 cards (green, yellow, orange, red). When choosing 3 colors, there is the chance that the middle (orange) color will be chosen as a safe standard estimation. When choosing 4 colors, estimators are forced to make a decision. When playing with 4 colors however, the incremental step from one color to the next is smaller compared with the 3 color deck of risk cards. For the estimators it might be more difficult to pick their choice or to reach consensus.

Acknowledgements

This article is based on the white paper 'Risk Poker explained' (2012), written by my colleague Jurriën Seubers, with contributions from Pascal Maus. The principles of risk based testing as explained in this article are based on the white paper 'Product Risk Based Testing' (2006), written by Erik van Veenendaal, Ruud Cox, Stephanie van Dijck and Jurian van de Laar.

> About the author



Jurian van de Laar

is senior consultant, trainer and Agile Testing coach at Improve Quality Services B.V. in The Netherlands. He has practical working experience since 1994 as software developer, team leader, tester and test consultant. He consulted various companies, including

TomTom, Philips, Triodos Bank and DHL. Jurian is teacher of courses in Agile testing, requirements engineering and structured testing (ISTQB). Jurian is a Certified Professional Scrum Master (Scrum.org) and certified in Prince2, TMap, CAT, ISTQB and IREB. As accredited trainer of the international Certified Agile Tester program (CAT) Jurian was the teacher of the first CAT training in The Netherlands. He contributed as author to various publications (e.g., Testing Experience) and he is a regular speaker at (inter-) national conferences (EuroSTAR 2009, Swiss Requirements Day 2010, Belgium Testing Days 2011). Jurian is vice chair of the Belgium and Netherlands Testing Qualifications Board (BNTQB).



Follow us @ar_mag