

Organised by



Event report



18 and 19 February 2016

Hosted by

GUNNEBO[®]

For a safer world

Thanks

The participants would all like once again to thank **GUNNEBO®** For a safer world for its excellent welcome and the efforts to which it went in order to organise this event.


In particular, we would like to thank:

- Philippe Schlatter, for his time and generosity in presenting **GUNNEBO®** For a safer world to us and in organising our visit of the site,
- Patrick Ditz, for the warm welcome he gave us and his lively approach throughout the event,
- Audrey Giacalone, for her help in organising the event,
- The teams at the Tir stand for the high-quality information that they passed on during the demonstration,
- The catering teams, for the high quality of the service they provided, the meals and their availability,
- Everybody else who was involved in any way, shape or form in organising and running this event.

Contents

Event aims.....	4
Event programme	5
Companies in attendance	6
Subject 1: Agile project:.....	7
Introduction: What is an agile project?	7
How can this methodology be implemented in a company environment?.....	9
Can the agile method only be used for software development?	9
How can an agile method be applied with a multi-project team?.....	9
Is the end of <i>Sprint</i> pressure a source of stress for the <i>Team</i> ?.....	9
What tools are there for applying this methodology?.....	10
Conclusion	10
Subject 2: Automatic Tests.....	11
When should the automation test be used?.....	11
How can an automatic test's return on investment be calculated?	11
Conclusion	12
Subject 3: Does the criticality of the test necessary depend on the criticality of the function?	13
Introduction.....	13
Criticality of the function	13
Criticality of the test.....	13
Definition of criticality	13
Criticality concept field of application	13
Criticality of the test.....	14
Conclusion	15
Summary of answers to the satisfaction questionnaire.....	16
A few photos.....	17

Event aims

The aim behind this event for  **Val** was to bring together everybody who shares its passion for...

Validation

This roundtable event aspired to being a forum for discussion about the methods and procedures used by the various organisations represented.

It was an opportunity for those in attendance to realise that a particular method that one particular organisation uses may be profitable and capable of boosting the performance of another one.



... and all in a fun and friendly environment

Without actually being a training programme about validation, the Roundtable often provided solutions to some of the problems encountered by the companies represented – particularly thanks to the cross-functional nature of what validators do!

Event programme

Thursday 18 February 2016

This year, we wanted to get the guests talking more about issues to do with validation. These subjects were chosen by all the participants and were:

- Agile project: How to supervise the project.
- Automatic test: What is the return on investment?
- Criticality: Does the criticality of a test necessary depend on the criticality of the function?

9:30 AM: Welcome with coffee

10 AM: General presentation & round-the-table introductions

11 AM: Site visit

12 PM: Lunch and drinks reception

1:30 PM Presentation of the "Get together – Discussion table" workshop

2 PM: Tackling of subjects and summaries based on the following rotation

	ROTATION 1	ROTATION 2	ROTATION 3	ROTATION 4	SUMMARY
Start time	2 PM	2:40 PM	3:20 PM	4 PM	4:40 PM
GROUP 1	Subject 1	Subject 2	Subject 3	Tir stand	Subject 1
GROUP 2	Subject 2	Subject 3	Tir stand	Subject 1	Subject 2
GROUP 3	Subject 3	Tir stand	Subject 1	Subject 2	Subject 3

6 PM:

7 PM: Workshop end

Friday 19 February 2016

The second half day was spent summarising the work that had been done in groups. Each group was tasked with summarising one of the subjects following the discussions that had taken place the day before, the aim being to allow everybody to express their point of view in relation to each subject.

- 8:30 AM: Welcome with coffee
- 9 AM: Summary of each subject
- 12 PM: Lunch and drinks reception
- 1:30 PM: Summing-up of the event
- 3:30 PM: End of the event

Companies in attendance



ALE (previously ALCATEL-LUCENT), ALSTOM, BRUKER BIOSPIN, EUROMEDEX, FAURECIA, GAGGENAU, GUNNEBO, HAGER, MERCK MILLIPORE, NTL, ONEFIT MEDICAL, POLE VEHICULE DU FUTUR, PUNCH POWERGLIDE, SIEMENS, SOCOMEC, SOMFY, YGGVAL

Subject 1: Agile project:

Introduction: What is an agile project?

"An agile method is an iterative and incremental approach which is adopted with a collaborative ethos, and just the right amount of formalism. It results in a high-quality product, while at the same time factoring in changes in clients' requirements". (Véronique Messenger-Rotta, 2011).

The structure of an agile method is incremental, iterative and adaptive. The client's satisfaction is its priority, and they are the focus of its approach. The basic principle involves recognising that it is counter-productive to plan and specify the tiniest detail of a product before developing it. Changes and uncertainties only result in problems and frustrations during the development phase.

When an agile approach is adopted, the deliverable is not a document, but instead a piece of the product. This methodology therefore keeps the number of superfluous documents to a minimum, with the focus being on delivering the product, piece by piece, iteration by iteration. Each iteration is considered a small project, ranging from the preliminary design phase through to delivery once all of the development phases and tests are complete.

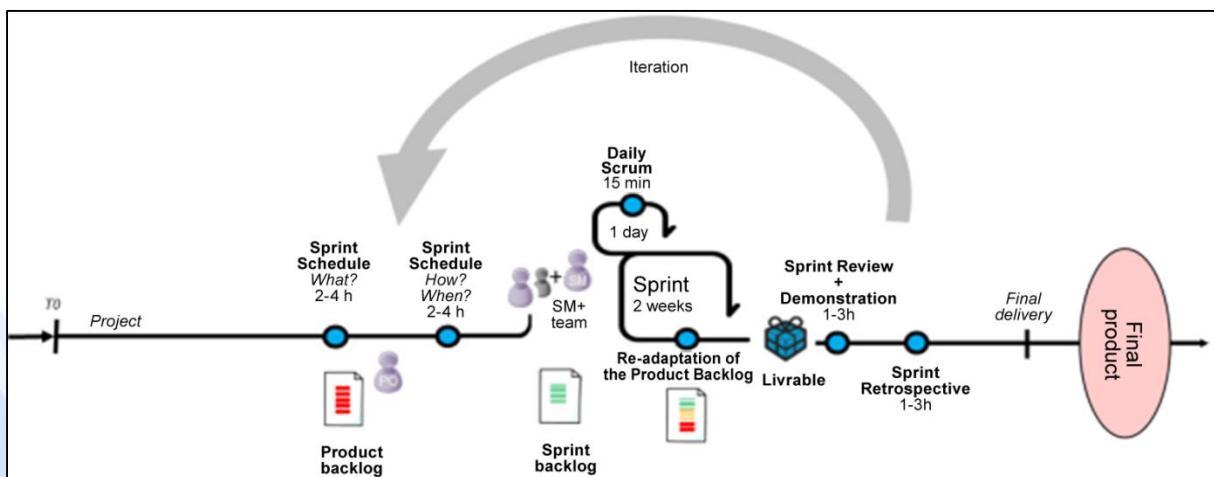
These methods are mainly used in the development of IT applications and require a high level of involvement on the part of the client in the project. It is their responsibility (or their representative's responsibility) to draw up a list of the product's functions and features – which is why the term "product management" is used, instead of "project management".

There are many advantages in adopting agile methods:

- Aims are clearly identified in the short term. If they are not met, they will be modified for the following iterations.
- Communication is central to all concerns.
- Visibility is improved.
- Quality is assessed on an on-going basis.
- Costs are controlled.
- Risks are limited, detected and quickly corrected.
- Client satisfaction is a priority.

Of all the various agile methods, the Scrum method is still the most widespread. In Scrum project development, there are various *Releases* of the product in question, made up of *Sprints*. Each *Sprint* is characterised by a well-defined framework and starts and ends with meetings. The end of the *Release* heralds the end the project and is identified by the end of the last *Sprint* after which the product's final version is delivered.

The documents are drafted for technical purposes, for development and functionality tests. The "*Product Backlog*" is a grouping together of all requirements, described as "*User Stories*". Each user story is ranked and prioritised based on complexity, cost and the velocity needed for its development. Often, they include writing the function, writing the unit test and reviewing the associated code. The figure below diagrammatically represents the way in which this method works.



The www.agiliste.fr website uses a metaphor that effectively represents the differences between Cycle V type predictive methods and agile type incremental methods:

The predictive method is comparable to "*planning a trip from Paris to southern France by car and avoiding all main roads. Each town and village through which you pass is specified, together with the associated time, each road driven down in the urban environments, the litres of petrol used, the miles covered, etc. And there will be lots of unexpected events as well: traffic jams, deviations, roadworks, traffic direction being reversed, even breakdowns, etc. The result is that all of your planning and your specifications will very quickly become obsolete*". Agility involves "*setting an initial short-term objective (a major town, for example) and then getting going without delay. Once you have reached this first objective, you take a*

short break and then adapt your itinerary based on the current situation. Then you continue thus until you have reached your final destination”.

How can this methodology be implemented in a company environment?

Often, incremental methodology is already used in organisations, but not in a formalised way. In fact, many development teams structure their approaches iteratively in a straitjacket that is modelled by Cycle V.

In order to formalise the switch from a predictive method to an adaptive method, ideally one should start by holding daily meetings – the famous *Stand-up* meetings associated with Scrum methodology.

This way, the development teams will get into the habit of communicating more and talking about how much progress they've made, the difficulties they've encountered, etc.

To start with, it might be interesting to introduce the idea of a "Talking stick".

Can the agile method only be used for software development?

Admittedly, software development is better suited to agile methodologies than the development of complex systems which combine hardware and software. However, it is not completely impossible to use this methodology for managing the development of a *hardware* product. It's also possible to blend different methods, stagger the *Sprints*, etc.

How can an agile method be applied with a multi-project team?

For compliance with good practice, a team can only be attached to a single agile project. However, the *Sprint* can be seen as a period (defined by the Team's velocity) to be filled with *User Stories* from various projects. The end of *Sprint* deliverable would therefore relate to several projects. However, there is the risk of difficulties being created when managing multiple projects.

Is the end of *Sprint* pressure a source of stress for the *Team*?

It's up to the *Team* to decide what it considers feasible during the *Sprint*. So it's up to the project initiator to make undertakings in relation to its production capacity. Only very rarely are their “eyes bigger than their stomach”. In all circumstances, the daily *Stand-up* meetings can be used to make adjustments to the launch.

What tools are there for applying this methodology?

Ice Scrum, Redmine and Tuleap are applications that can be used to apply agile methods.

Conclusion

Adopting an agile method does not mean dogmatically implementing it – as would a guru. Instead, it means adopting certain good practices, incorporating them into one's own project management philosophy in order to come out best.

There can be no doubt that the agile methodology has many advantages. But it should be implemented in such a way as to gradually bring about change in the teams.

However, it can be difficult to incorporate validation phases into *Sprints*. Several techniques can be used: the *Sprints* can be staggered or validation phases can be integrated into the development *Sprint*.

In all circumstances, the whole *Team* needs to be made aware of this new approach in order for an agile methodology to be permitted. Training programmes are now available to provide operatives with support and introduce them to the advantages of such a project management method, as well as areas that they need to watch out for.



Subject 2: Automatic Tests

When should the automation test be used?

The automatic test is not a wish or a miracle remedy. It is only developed as part of a test strategy. For this, the resources needed to implement the tests need to be determined beforehand.

A long-term view has to be adopted for an automatic test. For this, various factors need to be taken into account, such as:

- The product's life-cycle
- The automatic test's long-term life
- The automatic test's profitability

How can an automatic test's return on investment be calculated?

To calculate an automatic test's profitability, its cost must be known. This cost is not just the cost of the hardware needed to properly carry out the test, or the time it takes to carry it out. All of the additional hidden costs associated with an automatic test also have to be taken into account, such as:

- The time taken to build and implement the test platform
- The time required to carry out the test
- Maintaining the test platform
- The time required to analyse the results
- etc.



Conclusion

Calculating the return on investment for an automatic test is complicated: variables have to be carefully selected, qualified and quantified. Support is currently available for gauging the return on investment and – by extension – the profitability of test automation to a relatively high degree of accuracy.



Subject 3: Does the criticality of the test necessary depend on the criticality of the function?

Introduction

The dependency between the criticality of the test and the criticality of the function is certainly the most philosophical of the three subjects tackled during the event and made life difficult for our validator teams. Let's start by defining what the criticality of the function is.

Criticality of the function

In companies, it's often up to the marketing department (or the client's representative) to define the criticality of each function. Consequently, the idea of the function's importance is often confused with the idea of its criticality. So it is up to the client to decide whether this function is critical, major or minor for the system.

Criticality of the test

Again, the test usually depends on the criticality of the function: so all of the tests associated with a critical function are themselves critical.

Definition of criticality

When defining the level of criticality of a function or test, under no circumstances should the test cost and test execution time criteria be taken into account.

Similarly, when it comes to validation, the concept of criticality is completely different from the concept of zero bugs.

There is no single weighting method for criticality. Simple concepts can be adopted – using High, Medium and Low criticality levels. Or more complex ones can be adopted, such as Risk x Frequency or any other combination which is appropriate in a given context.

Criticality concept field of application

The criticality concept is mainly used for optimising non-regression tests. The tests to be repeated following modifications to the system are defined by an impact analysis, cross-referenced with the test or function criticality concept.

The second use for criticality is prioritising the tests to be conducted. It is easy to understand that by starting with the tests that have a high level of criticality, faults affecting the system's key parts can be detected first of all: this is test sequencing. However, it should be accepted that certain zones might not be covered because of the impact filter + criticality.

Criticality of the test

None of the previous information can be used to answer the question. But it can be used to define the context.

One of the examples mentioned during the discussion sessions is a vehicle's braking function.

Several tests are conducted on it:

Braking → Test to check that the ABS system is triggered

→ Test to check that the braking lights go on

→ Tests to ensure that the vehicle is stopped

If braking is a high-criticality function, then do all the associated tests necessarily have the same level of criticality?

Intuitively, one deduces that the braking lights going on and even the triggering of the ABS system are not necessarily tests that are as important as the vehicle being stopped.

The automotive sector provided us with an additional concept. An event which is feared.

By analysing the tests from this new perspective, it's obvious that the event that people fear in relation to the braking function is the vehicle failing to stop. Submitting all of the functions to this analysis should still make it possible to analyse criticality, while at the same time retaining a factual approach.

The other points which emerged from discussions are as follows:

A test cannot have a higher criticality level than the function with which it is associated. Obviously, the exception to this is tests which are more encompassing which are attached to several functions with different levels of criticality. In such cases, the test would not be able to have a level of criticality that was higher than the function with the highest level of criticality.

Conclusion

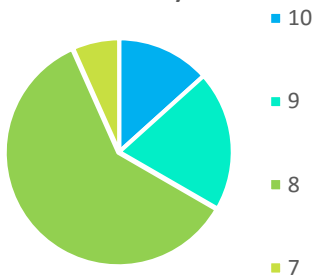
Yes, the criticality of the test does – to an extent – depend on the criticality of the function. As a general rule, the criticality of the test can be equal to the criticality of the function, but no more. But there is no miracle solution.

The other issue which could be looked into is the concept of inheritance between the criticality of the user requirement – towards functional requirement and detail requirement – and the dependency of these levels of criticality on the bottom-up part of the cycle. But that might be the subject of a future Validation Roundtable discussion...

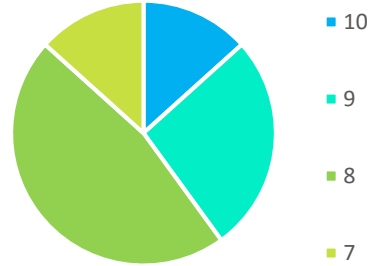


Summary of answers to the satisfaction questionnaire

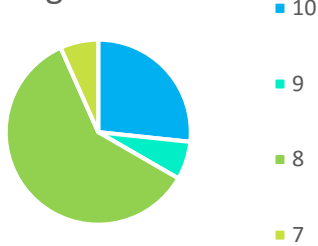
Structure of Thursday



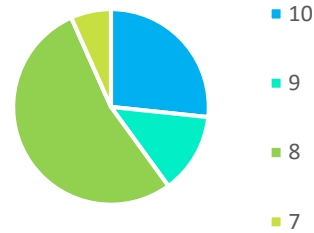
Structure of Friday



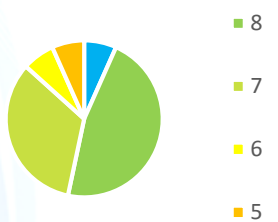
Quality of discussions during the breaks



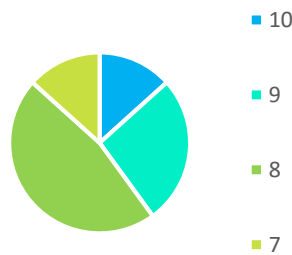
Relevance of subjects



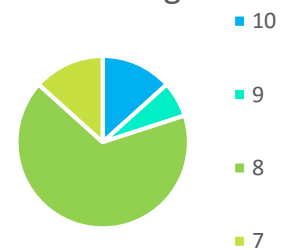
Quality of theoretical contributions



Relevance of workshops



Quality of debriefing



Average score awarded to the event: 8.29/10

A few photos

