A case study of Agile practices applied to a project seeking DO-178 certification

Kobus Coetzee
Graduate School of Technology Management
Faculty of Engineering, Built Environment and
Information Technology
University of Pretoria
Email: kobusengineer@gmail.com

Dr Fritz Solms
Department of Computer Science
Faculty of Engineering, Built Environment and
Information Technology
University of Pretoria
Email: fsolms@cs.up.ac.za

***Abstract*—This study measures the feasibility of applying agile software development practices to high reliability software projects seeking DO-178 certification; by comparing two student developer teams. One team uses traditional waterfall development methodologies while the other uses agile methodologies. The teams are compared on the Robustness, Functionality delivered, Maintainability, Certification effort required, Code bulk (LOC) and Quality of the documentation delivered.**

I. Introduction

Software development practices have progressed through a great many iterations since the first conference on software engineering sponsored and facilitated by NATO in 1968 [1]. Since the 1970s the development of software for military use has been standardised by the US Department of Defence, eventually leading to the generation of thousands of military standards of the form MIL-STDs [2]. These MIL-STDs for software developed eventually led to the creation of derivative standards for software development.

But since software engineering spans such a wide field, several best practices have developed for each field, with standards bodies regulating proper implementation of the practices, for example the RTCA (Radio Technical Commission for Aeronautics) specified standards being used in the aeronautical industry, MISRA (Motor Industry Software Reliability Association) specified standards being used in the automotive industry and the ISO (International Standards Organisation) regulating medical device software. Both these standards body has as its objective to guide their industries to ensure the development of products that are safe, reliable, robust and adheres to minimum operational performance requirements.

The RTCA has published the DO-178C [3] specification for Software Considerations in Airborne Systems and Equipment Certification. MISRA has published the MISRA-C:2012 [4] specification for Guidelines for the use of the C language in critical systems. ISO has published the IEC 62304 [5] specification for Medical Device Software - Software Lifecycle Processes.

These standards prescribe a strict adherence to process and fixed design and implementation life cycles, but software development best practises experienced a paradigm shift the last decade with the coming of agile methodologies. Software development methodologies such as Scrum or Kanban can be considered the agile equivalents to the RTCA and MISRA standards. Furthermore agile methodologies have been shown to mostly result in improved software delivery times, software quality and client satisfaction with the end product [6].

Even so, agile development is only practiced in a certain subset of software industries, mainly consumer software development [7]. Two software industries that have remained immune to agile development practices and still mainly done with 1970s era waterfall methodologies is military and safety critical software. Contrast for instance RTCAs and MISRAs focus on safety, reliability, robustness and minimum operational performance mainly through rigid process and verification, with the agile manifesto [8]:

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- *Individuals and interactions over Processes and tools*
- *Working software over Comprehensive documentation*
- *Customer collaboration over Contract negotiation*
- *Responding to change over Following a plan*

That is, while there is value in the items on the right, we value the items on the left more.

II. Literature survey

Very little research has been generated on the application of agile methods to highly regulated industries. [7] did systematic study on agile penetration in various industries.

Note the low occurrence of agile in the automation and medical industries. But problematic also is getting to a clear definition of what exactly agile software development entails. If the difference can only be regarded as iterative vs. incremental development practices, then agile methods have been in development since the 1970s [9]. The latest iteration and most popular form of agile methods takes the shape of SCRUM, a process which specifies short development sprints of roughly

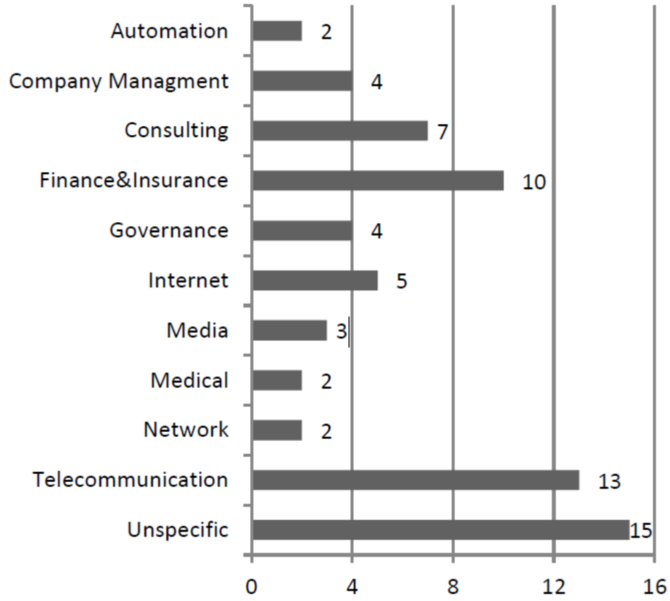


Fig. 1. Agile distribution across domains

two weeks in which certain developmental goals are specified and the team attempts to satisfy these goals. [10] did a literature review and found SCRUM did to lead to productivity improvements, but did not investigate if organisational make-up or the industry the company operates in have any effect. Also these findings are still very recent, and more in depth analysis is required.

A couple of case studies have been performed on implementing agile methodologies in regulated industries. [11] found tension points when implementing a SCRUM process in a case study where the company and product had to comply with several FDA, ISO, IEC and ISPE regulations. These tension points were in quality assurance, safety and security, effectiveness, traceability, verification and validation. But even so it was found that agile can be suitable and even bring about improvements to regulated industry if tailored correctly.

[12] also investigated if improvements can be made in the development of medical equipment when employing techniques from six sigma and agile, while complying with regulatory standards. A theoretical model was developed, and employed as a case study, resulting in improvements in productivity and a decrease in defects in the final product in the period the model was used between 2010 and 2013. It is unknown if wider applicability of the model exists for other industries with other certification requirements.

[13] pointed to further difficulties companies face in the medical device manufacturing industry when trying to employ agile methods while staying compliant with regulations. It is unclear if the case study was successful in employing agile methods into the development process.

[14] investigated a case in the application of agile methods in a aviation manufacturing setting, and identified open tool platforms and open design to be beneficial. The paper also mentioned regulatory difficulties as a problem for agile adoption, although this was not the focus of the study.

In addition to agile methodologies such as SCRUM and Kanban etc, improvements have been made in developmental tools that aid in agile development. These tools include agile centric project management software, CI (Continuous integration), CD (Continuous deployment, Unit testing, TDD (Test driven development), BDD (Behaviour driven development), pair programming and others.

[15] did a literature survey and investigated which of the different agile tools does indeed provide improvements in product quality, and found TDD and pair programming to be beneficial. It should be noted that the study was performed in 2010, and a lot of improvements to agile has happened since then.

[16] investigated uptake of TDD in industrial settings, and found the following limiting factors: increased development time, insufficient TDD experience/knowledge, lack of upfront design, domain and tool specific issues, lack of developer skill in writing test cases, insufficient adherence to TDD protocol, and legacy code. It is uncertain what is meant with an industrial setting, and if the companies involved was already engaged in other agile practices.

[17] shows that a balance between planned and agile methodologies can be found and formulates a strategy to arrive at such a balance. This paper is quite dated, especially given the velocity that agile is developing at, but represents a seminal work in blending agile and planned approaches from which several derivative works were produced.

But even with the relatively slow uptake of agile in DO-178 projects, some work has been done on what high reliability agile would look like. The FDA (US Food and Drug Administration) released TIR45 [**?**], which prescribes the application of agile software development practices for the development of safety critical medical devices seeking IEC 62304 certification.

Furthermore Open-DO [**?**] is an initiative for the discussion of the application of agile methodologies to the DO-178C specification. Open-DO is a discussion platform and facilitated a conference in 2010 in France.

III. Survey on agile uptake in DO-178 projects

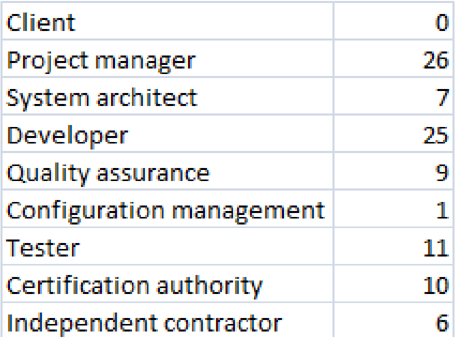
For the purposes of this study survey responses were collected to answer the following questions about the current state of the DO-178 certified software industry:

A. Is there a case for agile development within the DO-178
B. Is agile methods already being used within DO-178?
C. To get a benchmark to measure if agile methods improve DO-178

The survey was advertised on various DO-178 special interest groups and mailing lists, technical news aggregator sites as well emails sent to the authors personal contacts active with DO-178 development (This explains the high amount of South African respondents).

The survey was completed by 88 respondents, in various roles, with various team sizes and in various countries:

TABLE I
ROLES



Client	0
Project manager	26
System architect	7
Developer	25
Quality assurance	9
Configuration management	1
Tester	11
Certification authority	10
Independent contractor	6

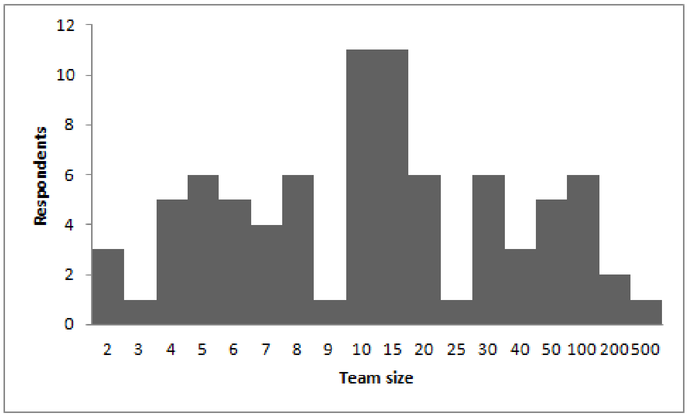
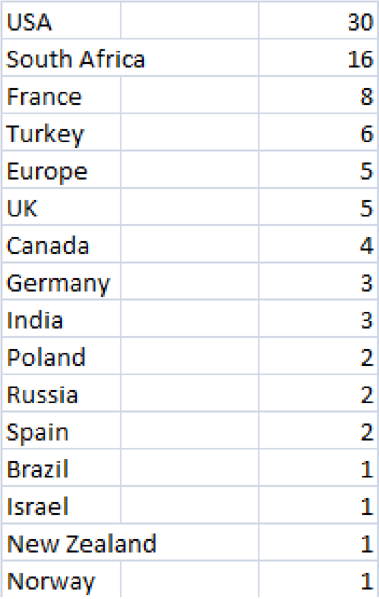


Fig. 2. Team size

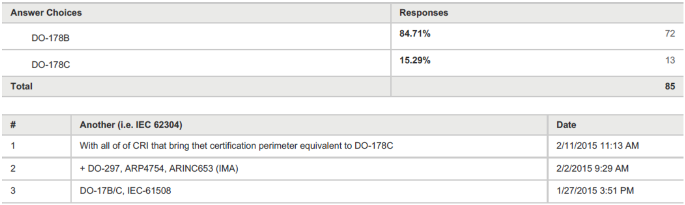
TABLE II
COUNTRIES



USA	30
South Africa	16
France	8
Turkey	6
Europe	5
UK	5
Canada	4
Germany	3
India	3
Poland	2
Russia	2
Spain	2
Brazil	1
Israel	1
New Zealand	1
Norway	1

Most respondents indicated they are still following the DO-178B specification and haven't switched over to the newer DO-178C specification which was released in January 2012.

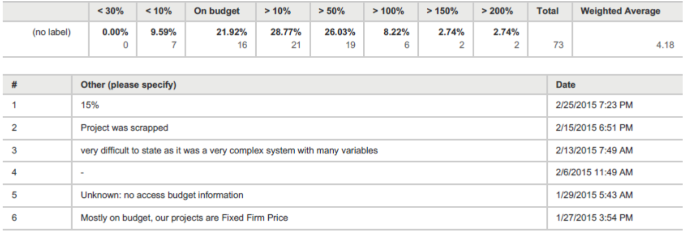
TABLE III
DO-178B AND DO-178C USE



Answer Choices	Responses	
DO-178B	84.71%	72
DO-178C	15.29%	13
Total		85

#	Another (i.e. IEC 62304)	Date
1	With all of of CRI that bring thet certification perimeter equivalent to DO-178C	2/11/2015 11:13 AM
2	+ DO-297, ARP4754, ARINC653 (IMA)	2/2/2015 9:29 AM
3	DO-17B/C, IEC-61508	1/27/2015 3:51 PM

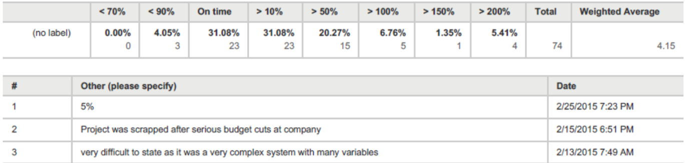
TABLE IV
WHAT WAS YOUR LAST PROJECT'S BUDGET PERFORMANCE?



	< 30%	< 10%	On budget	> 10%	> 50%	> 100%	> 150%	> 200%	Total	Weighted Average
(no label)	0.00% 0	9.59% 7	21.92% 16	28.77% 21	26.03% 19	8.22% 6	2.74% 2	2.74% 2	73	4.18

#	Other (please specify)	Date
1	15%	2/25/2015 7:23 PM
2	Project was scrapped	2/15/2015 6:51 PM
3	very difficult to state as it was a very complex system with many variables	2/13/2015 7:49 AM
4	-	2/6/2015 11:49 AM
5	Unknown: no access budget information	1/29/2015 5:43 AM
6	Mostly on budget, our projects are Fixed Firm Price	1/27/2015 3:54 PM

TABLE V
WHAT WAS YOUR LAST PROJECT'S DELIVERY DATE PERFORMANCE?



	< 70%	< 90%	On time	> 10%	> 50%	> 100%	> 150%	> 200%	Total	Weighted Average
(no label)	0.00% 0	4.05% 3	31.08% 23	31.08% 23	20.27% 15	6.76% 5	1.35% 1	5.41% 4	74	4.15

#	Other (please specify)	Date
1	5%	2/25/2015 7:23 PM
2	Project was scrapped after serious budget cuts at company	2/15/2015 6:51 PM
3	very difficult to state as it was a very complex system with many variables	2/13/2015 7:49 AM

A. *A case for agile*

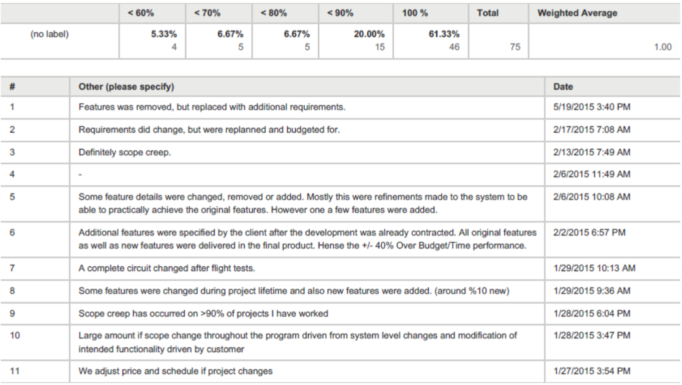
To determine if there is a case for the use of agile within DO-178, we looked at the current performance of DO-178 projects. This was done by asking the survey participants the following questions:

- What was your last projects budget performance?
- What was your last project's delivery date performance?
- How many features was delivered according to the original specification?
- Was there any requests for additional features during the development? (Scope creep)
- Was there any requests for additional features after certification was complete? (Scope creep)
- In your opinion, does DO-178 increase the costs to a project? By how much?

B. *Is agile being used*

We also wanted to find out to which extent agile techniques were already being practiced in the industry. But not everybody in the industry agrees as to the same definition of what an agile development approach is. As such we asked the respondents if they were using a waterfall, rational unified process, kanban and scrum approach. We correlated this metric by asking respondents to confirm if they were using various developmental tools, some being agile orientated and some waterfall orientated. These responses were then correlated

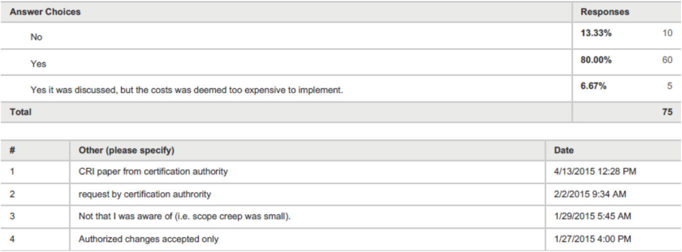
TABLE VI
HOW MANY FEATURES WAS DELIVERED ACCORDING TO THE ORIGINAL SPECIFICATION?



	< 60%	< 70%	< 80%	< 90%	100 %	Total	Weighted Average
(no label)	5.33% 4	6.67% 5	6.67% 5	20.00% 15	61.33% 46	75	1.00

#	Other (please specify)	Date
1	Features was removed, but replaced with additional requirements.	5/19/2015 3:40 PM
2	Requirements did change, but were replanned and budgeted for.	2/17/2015 7:08 AM
3	Definitely scope creep.	2/13/2015 7:49 AM
4	-	2/6/2015 11:49 AM
5	Some feature details were changed, removed or added. Mostly this were refinements made to the system to be able to practically achieve the original features. However one a few features were added.	2/6/2015 10:08 AM
6	Additional features were specified by the client after the development was already contracted. All original features as well as new features were delivered in the final product. Hense the +/- 40% Over Budget/Time performance.	2/2/2015 6:57 PM
7	A complete circuit changed after flight tests.	1/29/2015 10:13 AM
8	Some features were changed during project lifetime and also new features were added. (around %10 new)	1/29/2015 9:36 AM
9	Scope creep has occurred on >90% of projects I have worked	1/28/2015 6:04 PM
10	Large amount if scope change throughout the program driven from system level changes and modification of intended functionality driven by customer	1/28/2015 3:47 PM
11	We adjust price and schedule if project changes	1/27/2015 3:54 PM

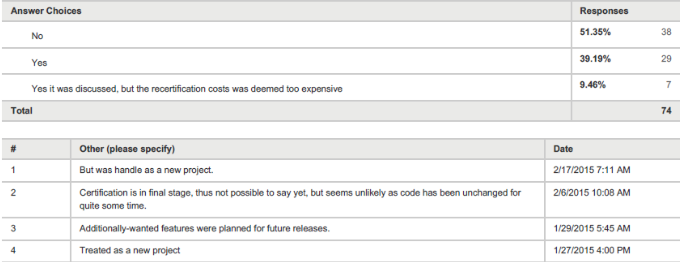
TABLE VII
WAS THERE ANY REQUESTS FOR ADDITIONAL FEATURES DURING THE DEVELOPMENT?



Answer Choices	Responses	
No	13.33%	10
Yes	80.00%	60
Yes it was discussed, but the costs was deemed too expensive to implement.	6.67%	5
Total		75

#	Other (please specify)	Date
1	CRI paper from certification authority	4/13/2015 12:28 PM
2	request by certification authrority	2/2/2015 9:34 AM
3	Not that I was aware of (i.e. scope creep was small).	1/29/2015 5:45 AM
4	Authorized changes accepted only	1/27/2015 4:00 PM

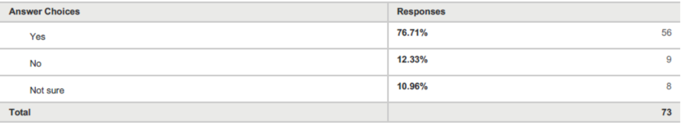
TABLE VIII
WAS THERE ANY REQUESTS FOR ADDITIONAL FEATURES AFTER CERTIFICATION WAS COMPLETE?



Answer Choices	Responses	
No	51.35%	38
Yes	39.19%	29
Yes it was discussed, but the recertification costs was deemed too expensive	9.46%	7
Total		74

#	Other (please specify)	Date
1	But was handle as a new project.	2/17/2015 7:11 AM
2	Certification is in final stage, thus not possible to say yet, but seems unlikely as code has been unchanged for quite some time.	2/6/2015 10:08 AM
3	Additionally-wanted features were planned for future releases.	1/29/2015 5:45 AM
4	Treated as a new project	1/27/2015 4:00 PM

TABLE IX
IN YOUR OPINION, DOES DO-178 INCREASE THE COSTS TO A PROJECT?



Answer Choices	Responses	
Yes	76.71%	56
No	12.33%	9
Not sure	10.96%	8
Total		73

with the developmental approach responses, and measured for correlation.

We also asked the respondents if they felt the development approach was helpful or not in their completion of their projects:

Note the percentages do not necessarily add up, as respondents may have filled in for multiple projects or multiple approaches may have been used within the same project (subcomponents for instance).

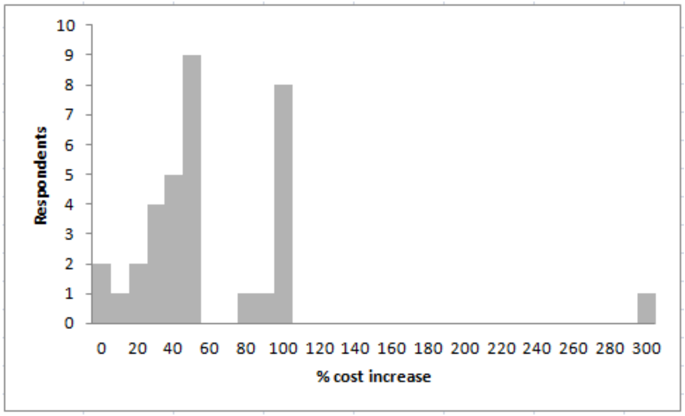
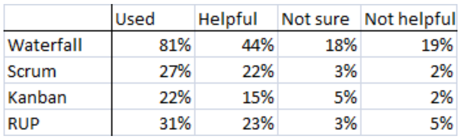


Fig. 3. By how much does costs increase?

TABLE X
WATERFALL VS AGILE USE



	Used	Helpful	Not sure	Not helpful
Waterfall	81%	44%	18%	19%
Scrum	27%	22%	3%	2%
Kanban	22%	15%	5%	2%
RUP	31%	23%	3%	5%

C. DO-178 benchmarks

To measure the case study performance, the following benchmarks were gathered from the survey:

- What is your $ cost per line of code?
- What is your bugs per 1000 lines of code?

Off course these benchmarks presents difficulties of their own, such as inconsistent measuring...

IV. CONCEPTUAL MODEL

The following conceptual model were developed, taking inputs from TIR45 [?] and Open-DO [?], but also taking general agile development principles, especially SCRUM and applying it to the required DO-178 outputs and objectives.

The conceptual model is focused on bringing continuous integration and continuous deployment to a DO-178 environment. The purposes is to reduce costs through automation of repeated processes. Since DO-178B is a software quality assurance standard, not a software development standard, it does not impose any restrictions or considerations on how software is to be developed. It does however require the following list of deliverables, with the requirements for each depending on the criticality level chosen: **Organisational** (These can be re-used across multiple projects, if the projects are similar enough off course):

- Software configuration management plan (SCM)
- Software quality assurance plan (SQA)
- Software requirements standards (SRS)
- Software design standards (SDS)
- Software code standards (SCS)

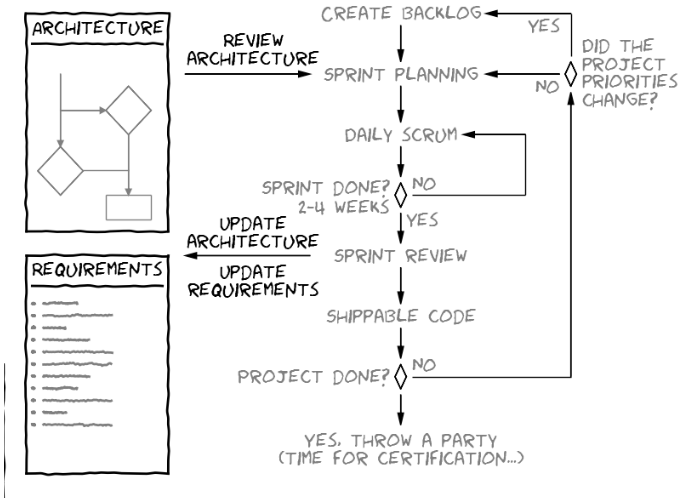


Fig. 4. Agile DO-178 framework

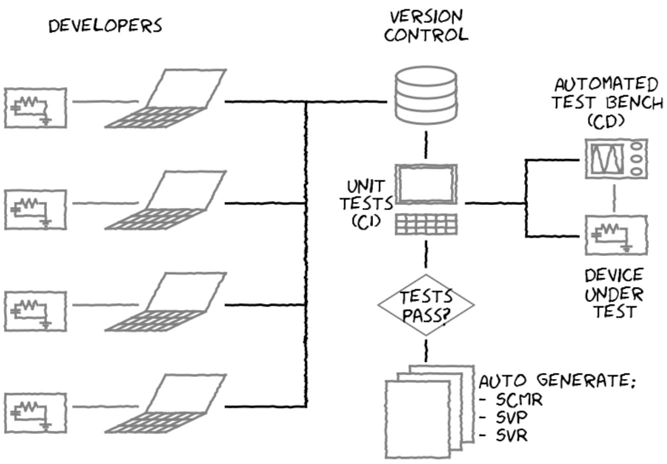


Fig. 5. Agile infrastructure

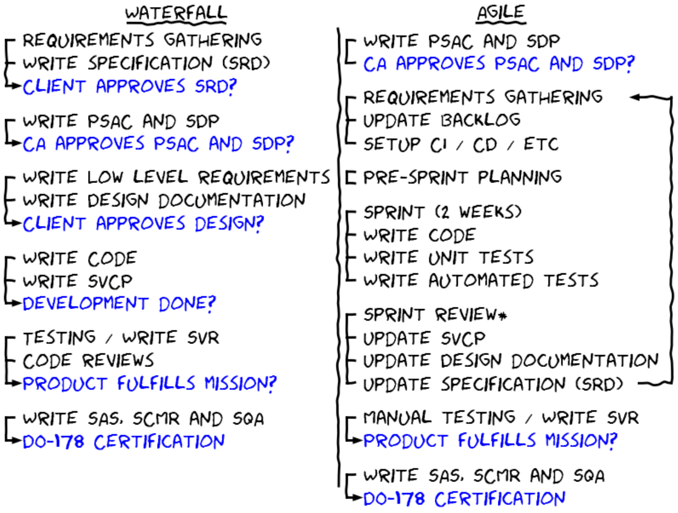


Fig. 6. Waterfall vs Agile comparison

- Software verification plan (SVP)

At the start of a project - Specification phase:

- Plan for software aspects of certification (PSAC)
- Software development plan (SDP)
- Software requirements data (SRD)
- Design description (DD)

During development - Implementation phase:

- Source code
- Object code
- Software verification Cases and Procedures (SVCP)
- Problem reports
- Software Quality Assurance Records (SQA)
- Software Configuration Management Records (SCMR)

At the end of a project - Testing and verification phase:

- Software Verification Results (SVR)
- Software Life Cycle Environment Configuration Index (SECI)
- Software Configuration Index (SCI)
- Software Accomplishment Summary (SAS)

The automated generation of some of these deliverables requires the development of infrastructure and tools that will facilitate a more agile way of working.

These automated generated outputs still has to comply with the DO-178 requirements on outputs as stipulated in the Annex A of the specification.

V. Research design and methodology

In order to test the effect that a more agile project management approach would have on a project requiring DO-178 certification, a case study was launched. In this case study two 3'rd year student teams developed and delivered a project according to exactly the same specification, with the one team running the agile workflow and the other a more traditional waterfall workflow.

The teams were tasked with developing group chat functionality for the Linphone open source project [?], and the final deliverable were measured for robustness of the solution, functionality delivered, maintainability, certification effort required, code bulk and the documentation bulk delivered.

Robustment were determined with independent functional testing of the solution, as well as the amount of unit tests and functional tests each team generated during development.

Functionality delivered was a simple count against the original specification provided to each team.

Maintainability was determined by calculating the cyclomatic complexity of their code, as well as asking each team to give an estimate as to the effort required to develop additional functionality.

Certification effort still required were determined from external evaluation of each teams deliverables as measured against the DO-178 standard, and how close each team is to achieving a successful certification.

Code bulk is simply the count of lines of code delivered, documentation bulk is the word count of the documentation.

Documentation also includes code comments.

Because the teams are not identical, in their size or their capability, a normalization task was given to each team. In this task, they were given exactly the same programming assignment to be completed in roughly half a day. The teams were measured on how long they took to complete the task and the quality of their final solution (how many bugs were still present). All case study metrics were then scaled against the results of this normalization exercise.

VI. Case study

The 3rd year computer science students at the University of Pretoria has a year long subject where they complete a project as a team. This is the students first substantial project working as a team. These students are presented with a list of projects they are allowed to tender for. On this list was included two projects for this case study. Both projects listed exactly the same requirements, but it was stipulated that the one project will be run agile and the other waterfall. So some self selection of the students biased towards agile or waterfall workflows are present in the case study. For the tenders, six teams applied, three for each project. The two best applications were selected for this case study.

Interviews with Stacy and Freda still to take place middle September on how well the teams worked together, strengths and weaknesses of each team etc.

The agile team selected consists of six members and the waterfall team selected consists of five members. Team members from both teams have some experience with class projects they have completed, served as tutors in various but no large scale projects (multiple team members). Neither teams have experience with DO-178 certified software projects, and have only a theoretical understanding of agile and waterfall development practices.

The teams submitted their tenders at the beginning of May 2015, and were notified of their successful application a week later. The first meeting between the teams and the client took place end of May after which the teams started their projects in earnest. The teams had about 6 months time from then on to complete the following list of identical requirements:

- Group chat (Invite additional members to a chat, all members receive chats)
- Secure group chat (AES256)
- Creation and deletion of groups
- Voice record and send over IM
- Rework the messaging user interface
  - Spacing between words are terrible
  - Make the text bigger
  - Block indents required to better specify who said what
  - Presence indication to show a remote user is typing
  - User picture portraits

Emphasis was placed with the student groups that the core requirement was the group chat functionality, then the encryption of the chats, and that all other requirements were of lesser importance. This was done to ensure both groups progress through the work similarly and that one group don't complete the easy work and the other the more difficult work items, which would make comparison difficult.

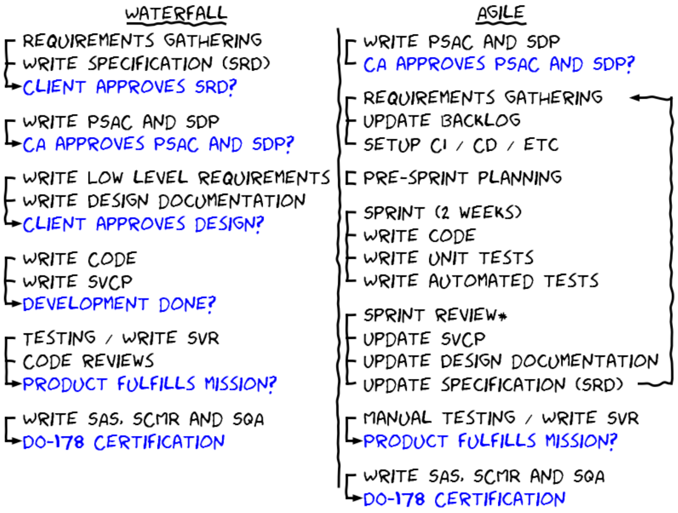


Fig. 7. Progress of the teams (TODO)

VII. Results

Study is still running... will finish October / November. Survey on students asking what they struggled with, how much time was spent on the project, estimation on adding features (robustness) etc still to be done (two weeks before final completion). Normalisation calibration still to take place middle October.

VIII. Conclusion

The conclusion goes here.

Acknowledgment

The authors would like to thank...

- Mr Kobus Steyn for his contribution on DO-178 considerations and best practices.
- Ms Feziwe Mbatyoti for her external evaluation of the groups compliance to the DO-178 standard.
- Ms Vreda Pieterse (lecturer) and Ms Stacy Omeleze (assistant lecturer) for their help with running the case study.

  The following B.Sc Comp Sci students for participating in the study:
- Patience Mtsweni
- Lerato Molokomme
- Tsepo Ntsaba
- Mpedi Mello
- Lutvyya Razak
- Ephiphania Munava
- Izak Blom
- David Breetzke
- Paul Engelke

- Prenolan Govender
- Jessica Lessev

This study was completed as partial fulfilment of the requirements for the degree of:

Master of Engineering (MEM)
in the
GRADUATE SCHOOL OF TECHNOLOGY MANAGEMENT,
FACULTY OF ENGINEERING, BUILT ENVIRONMENT AND INFORMATION TECHNOLOGY,
UNIVERSITY OF PRETORIA

References

[1] P. D. F. L. Bauer, “Software engineering,” in *NATO SOFTWARE ENGINEERING CONFERENCE 1968*, P. Naur and B. Randell, Eds., 1968.

[2] C. McDonald, “From art form to engineering discipline? a history of us military software development standards, 1974&amp;#x2013;1998,” *Annals of the History of Computing, IEEE*, vol. 32, no. 4, pp. 32–47, 2010.

[3] I. RTCA, “Do-178c / ed-12c,” 01/05/2012.

[4] MISRA, “Guidelines for the use of the c language in critical systems,” 2004.

[5] P. Jordan, “Standard iec 62304 - medical device software - software life-cycle processes,” in *Software for Medical Devices, 2006. The Institution of Engineering and Technology Seminar on*, 2006, pp. 41–47.

[6] O. Armbrust and D. Rombach, “The right process for each context: objective evidence needed,” pp. 237–241, 2011.

[7] P. Diebold and M. Dahlem, “Agile practices in practice: a mapping study,” pp. 1–10, 2014.

[8] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, and R. Jeffries, “Manifesto for agile software development,” 2001.

[9] C. Larman and V. R. Basili, “Iterative and incremental developments. a brief history,” *Computer*, vol. 36, no. 6, pp. 47–56, 2003.

[10] E. Cardozo, J. Neto, A. Barza, A. Frana, and F. da Silva, “Scrum and productivity in software projects: a systematic literature review,” in *14th International Conference on Evaluation and Assessment in Software Engineering (EASE)*, 2010.

[11] B. Fitzgerald, K.-J. Stol, R. O, #039, Sullivan, D. O, #039, and Brien, “Scaling agile methods to regulated environments: an industry case study,” pp. 863–872, 2013.

[12] M. Shen, W. Yang, G. Rong, and D. Shao, “Applying agile methods to embedded software development: a systematic review,” pp. 30–36, 2012.

[13] N. Hrgarek, “Certification and regulatory challenges in medical device software development,” pp. 40–43, 2012.

[14] M. Glas and S. Ziemer, “Challenges for agile development of large systems in the aviation industry,” pp. 901–908, 2009.

[15] P. Sfetsos and I. Stamelos, “Empirical studies on quality in agile practices: A systematic literature review,” in *Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the*, 2010, pp. 44–53.

[16] A. Causevic, D. Sundmark, and S. Punnekkat, “Factors limiting industrial adoption of test driven development: A systematic review,” in *Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on*, 2011, pp. 337–346.

[17] Boehm and R. Turner, *Balancing Agility and Discipline: A Guide for the Perplexed*. Addison-Wesley Longman Publishing Co., Inc., 2003.