Agile safety case for vehicle trial operations

Thor Myklebust[a], Tor Stålhane[b], and Sinuo Wu[c]
[a] SINTEF Digital, Trondheim, Norway, thor.myklebust@sintef.no
[b] NTNU IDI, Trondheim, Norway, stalhane@ntnu.no
[c] University of South-Eastern Norway, Norway, 238868@student.usn.no

Abstract: In the last years, there has been an increase in the use of agile development methods when developing safety-critical software. The agile approach fits well with the incremental improvement of software for autonomous vehicles, incremental expansion of the operational design domain, and new intelligent roadside units.
There will be new trials of self-driving vehicles in the years to come, due to the expected improvements in the vehicles and intelligent roadside units. Therefore, it is essential that the process, including needed evidence for a safety case, are both agile and standardized to ensure confidence and trust by all parties involved.
This paper shows how the trial operator can develop an agile safety case for vehicle trial operations to ensure frequent updates based on:
• The agile safety case
• ISO 22737:2021 Low-speed automated driving
• BSI PAS 1881:2020 "Assuring the safety of automated vehicle trials and testing - specification" standard
• The BSI PAS 1883:2020 "Operational design domain (ODD) taxonomy for an automated driving system (ADS) - Specification" standard.
The agile development approach enables approval of the safety of already developed parts to be done by the manufacture and operator in parallel with other development.
Through our more than a hundred safety case-related projects (mainly railway domain), we have also seen that an agile safety case approach results in increased safety awareness, confidence, and understanding of the safety challenges among the software developers and project engineers.

1. INTRODUCTION

Safety case (SC) – also called assurance case or safety demonstration – has for a long time been required by safety standards for safety-critical systems in important industrial areas such as nuclear, automotive, and railways. The earliest reference we have found is Def. 00-55:97 from 1997. Safety case is an efficient method for making the developing company focus on the simple but important question, “How do you know that your system is safe enough?” The idea of a safety case is not to provide mathematical or statistical proof but to argue as one would in a court of law – thus the name safety case. It is efficient to build the safety case by inserting information when it becomes available – an agile approach which also results in increased safety awareness and understanding.

The purpose of a Safety Case is to develop structured arguments supported by evidence intended to justify that a product or system is acceptably safe for a specific application in a specific operating environment. The purpose of "The Agile Safety Case" (TASC) [1, 5] also includes adaptability, flexibility, improved communication, and effective solutions.

Regarding how to show compliance for trial operations, BSI (British Standards Institution) in the UK has issued the BSI PAS 1881:2020 "Assuring the safety of automated vehicle trials and testing - specification" standard. This PAS specifies the minimum requirements for safety cases for automated vehicle trials and development testing in the UK. Normally it is the operator of the vehicle/ busses that issues this safety case.

Due to the rapid technology development, the safety standards may not include all relevant safety aspects. Evidence using a Safety case can be extended to cover safety issues beyond the scope of safety standards.

The objective of this paper is to show how we may ensure that the process, including needed evidence for a safety case, are both agile and standardized to ensure confidence and trust by all parties involved.

2. BACKGROUND

2.1. The TrustMe project

The TrustMe project started 1st of August 2020 and will last until the end of 2023. The main goal of the TrustMe project is to develop a safety case for autonomous buses, a safety case for trial operations, and a safety case for the public [2, 3]. Safety cases are important for establishing sufficient confidence in the new technology and to get approval for test-driving in normal traffic. The long-term goal is to get regular operation with passengers without an operator on-board the bus. The safety case [1, 5], together with a Trust case [6], shall justify the trust of several user groups such as passengers and fellow road users, the government, and the insurance industry, by bringing together necessary amount of information that documents the safety level.

2.2. Autonomous bus and trial testing

Development of self-driving car and buses is an important trend in the automotive world today, with trust emerging as an important factor in this process. However, the manufacturers and operators that are working on autonomous vehicle (AV) systems can't just test their technology on any public road. Cities and communities worldwide are increasingly functioning as testing grounds as the pace of autonomous vehicle innovation picks up. Local governments first need to approve self-driving pilots to operate.

There is an effort to develop fully autonomous busses that operators will use to provide mobility-as-a-service (MaaS). The MaaS segment focuses on small shuttle buses with pre-defined routes or geo-fencing to establish a safe route. Many of these shuttle buses are still unable to go around an obstacle or stop before the obstacle in their pre-defined route. They currently operate at low speeds between 12 - 30 km/h. So, according to agile terms, the busses are clearly an MVP (Minimum Viable Product) or, for automotive, an MVV (Minimum Viable Vehicle). A minimum viable product is a version of a product with just enough features to be usable by early customers who can provide feedback for further development of the product or vehicle.

Several countries have established legal frameworks for the regulation of pilot testing – e.g., Singapore, The Netherlands and Norway cording to the UK KPMG report [9]. The UK framework [12] mentions safety case as a necessary methodology.

2.3. The agile safety case

Manufacturers and operators want to convince their customers that the vehicle is safe. At the top level, a safety case goal is simple to imagine. The statement “The system is safe because…” says it all. Whatever follows “because” is a safety case [1]. The purpose of a safety case is to inform the reader – e.g., a safety assessor – about

- what you have done to make the system safe,
- how it contributes to safety and evidence that you have developed
- what you claim to have done, including proof that the persons who did the job had the right competencies.

Agile development fits well with the development of safety cases since we develop the system “one small piece at a time”. Combining agility and a safety case approach will improve project

communication. Agile development gives frequent, short, and focused meetings and ensures that everybody is updated. Building safety cases provide information on how each safety problem is solved and which safety problems are not yet solved. Building the safety case in an agile way will also improve communication with the assessor since we then can show what we have done so far and get feedback on whether our solutions are acceptable to the assessor. Some published experiences have shown that this is a feasible approach [7].

2.4. ISO 22737:2021 Low-speed automated driving

Low Speed Autonomous Driving (LSAD) systems are often used in self-driving cars for commercial areas, industrial park, or university campus areas, helping to solve the 'last or first mile' problem without carbon emissions. For example, the self-driving buses involved in the TrustMe project have applied LSAD systems. The systems in the autonomous vehicles have been set to run on a pre-defined route in a low-speed environment, as a good choice for short distance between a transportation hub and another destination.

For a long time, the development of LSAD technology has been hindered by the lack of international standards that define the needed performance and safety requirements. This makes manufacturers feel that it is difficult to describe the level of safety engineering that the vehicles have or to compare various attributes and functions against the perceived state of the vehicle. Fortunately, the emergence of ISO 22737:2021 is going to remove this obstacle.

ISO 22737 is the first international safety standard for automated driving systems. It specifies a minimum set of operating capabilities, risk maneuver, and safety requirements in LSAD systems, which allows manufactures to take relevant aspect into consideration during systems design process.
The speed limits in the standard are:

- Vehicle speed < 32 km/hour
- Pedestrian speed < 8 km/hour
- Pedal cyclist speed < 25 km/hour

This standard also offers guidance on operational design domain (ODD) limitations and how LSAD can be designed to fit different traffic situations. It also includes performance test procedures for various systems and situations.

2.5. BSI PAS 1881:2020 "Assuring the safety of automated vehicle trials and testing - specification" standard

It isn't easy to standardize safety cases since a safety case will continue to grow and evolve as the complexity of interactions increases. However, there are standards for risk management principles, and the acceptance of safety cases can remain consist with this process. This is the meaning of the safety case framework, which specifies which safety cases should include compliance with which standard.
As a safety case framework, BSI PAS 1881:2020 is the first standard published on the Connected and Automated Vehicle (CAV) safety trials. The purpose of this standard is to make it possible to deploy CAV safely on UK roads. BSI in the UK developed this standard and made it published under license from The British Standards Institution. Operators can apply this standard to assure the safety of Automated Vehicle Trials and tests.

PAS 1881 deals with how to build safety cases which include operational design domain and test objectives, operational risk assessments, safety testing, operational guidance and training, safety monitoring, compliance, and permissions granted.

The purpose of developing a robust operational safety case is to demonstrate that the potential risks for all parties are kept to the lowest level reasonably practicable – the ALARP principle – [11] during the testing and trials of automated vehicles. In addition, it focuses on operational safety and refers to the required outcomes from safety assessments. Since the security requirements are transparent, this risk management standardization will also enable the ever-changing security cases to run trails easier in the future.

2.6. BSI PAS 1883:2020 "Operational design domain (ODD) taxonomy for an automated driving system (ADS) - Specification" standard.

BSI PAS 1883:2020 is one in the series commissioned by The Centre for CAV to support the development of CAVs in the UK. It specifies the minimum hierarchical taxonomy required to specify an Operational Design Domain (ODD), so that automated driving systems (ADS) can be deployed safely by the manufacturers.

It defines a generic taxonomy for defining all environments in which an ADS may be deployed, tested, or trailed. This visualizes the safety of an ADS and can make it become easier to trust. Through the ODD taxonomy in this PAS, manufacturers can specify and implement the minimum safety requirements in their designs while end-users, operators, and regulators can reference a minimum set of ODD attributes and performance requirements in their procurements accurately and consistently.

The contribution of PAS 1881 and PAS1883 is to build confidence in autonomous vehicles and justify to the public or third parties why the vehicles are safe. Furthermore, it helps shape the future of international CAV standards, which are intended for developers, manufacturers, insurance companies, etc.

3. AGILE SAFETY CASE FOR VEHICLE TRIAL OPERATIONS

3.1 Introduction as part of this paper

There has been increasing use of agile development methods when developing safety-critical software. This approach fits well with the necessary incremental improvement of

- autonomous vehicles,
- ODD
- intelligent road products (intelligent signals, weather, traffic detection, etc.)

In addition, it is important to have an agile process for necessary security patches. An agile process is included to ensure that the patch can be performed within a given period of time, e.g., within one sprint (often 2-4 weeks).

Chapter 3 of this paper constitutes an adapted application of the agile safety case [1] and BSI PAS 1881:2021 (from now on named PAS 1881) for trials of autonomous vehicles, including an agile approach, mainly for the software part. The chapter names in the bullet points below are similar to the chapters presented in PAS 1881 to ensure a traceable link to this PAS. The chapters have been reorganized to follow a common development process. We have only added the chapter "Safety case summary and conclusion" to ensure a quick overview of the main results, together with an overview of possible reservations, non-compliance, SRAC's (Safety Related Application Conditions), and recommendations. In addition, we have added seven subchapters:

- SRAC, ch.3.7.1
- SecRAC (Security-Related Application Conditions), ch.3.9.1
- Manufacturer, vehicle type, authorization and license number, ch.3.10.1
- Hierarchy of safety cases and related certifications, ch.3.10.2
- Vehicle identification number, ch. 3.10.3

- Software identification number, ch. 3.10.4
- Vehicle authorization summary, ch. 3.10.5

3.2. Introduction, purpose, and scope of the safety case

This chapter of the safety case should include descriptions of the trial, testing, or service, including roles together with the involvement of the public.

Relevant standards that are applied should be mentioned in the safety case, e.g., ISO 26262:2018 series, ISO 21448:2019 Safety of the intended functionality (SOTIF) ISO/SAE 21434:2021 security, and ISO 22737:2021 LSAD. Change history of the safety case should be included. Summarize the change in a few sentences. Version number and date have to be included. This is also practical information for assessors and certification bodies.

3.3. Automatic/autonomous vehicle system, Description of System (DoS)

This is normally a short chapter since a reference to a separate DoS document [13] – vehicle including relevant sensors – in most cases is included in the safety case. An agile approach includes MVP (Minimum Viable Product, applies also to systems and relevant pats of the system, a bus in this case), incremental development of some of the vehicle systems as e.g., ADS (automatic driving system) and the relevant sensors, including sensor fusion development.

3.4. Operational design domain and test scenarios

This chapter is linked to the DoS and may be based on e.g.

- SAE 2020-04: AVSC00002202004. "AVSC Best Practice for Describing an Operational Design Domain: Conceptual Framework and Lexicon"
- ISO 22737:2021 LSAD

An agile approach includes MVP (Minimum Viable Product, the ODD/OEDR in this case), relevant ODD limitations at the beginning of the project, incremental development of the vehicle, and the relevant sensors, including sensor fusion development. In addition, an incremental improvement of the ODD.

3.5. Route selection and assessment

In most of the trial operations during the last years, pre-defined routes have been chosen, as described in ISO 22737:2021. The operator should describe the pre-defined route(s). We also need a description of how the operator will ensure that the vehicle does not operate outside the route(s). This could be done by, e.g., physical barriers or geo-fencing. So far we have not seen incremental development of the route chosen, but rather moving to new test sites.

3.6. Operational risk assessment

An operational risk assessment is often presented as a separate document. The process and evidence should be reusable and include an agile approach since the analysis has to be updated as part of the incremental development of the system's description – vehicle including sensors and the ODD. Highways England has issued requirements for safety risk assessments [8].

3.7. Operational guidance

This information is often included in the OM (Operation and Maintenance) manual. In some cases, the SRACs are also included in the OM. In some projects, they are included in a separate document or only

in the safety case. Often several SRACs are issued early in the project, and then the vehicle, ODD, and, e.g., HMI (Human Machine Interface) are improved incrementally as part of the project.

3.7.1. Safety-related application conditions

This section defines the rules, conditions, and constraints relevant to safety that need to be observed in the application of the vehicle. Each SRAC shall be linked to at least one hazard and should be understandable for the operator. SRAC's are sometimes included in the IOM (Installation, operation and maintenance) manual, see subchapter below.

Sometimes there are several SRAC's in the first release. The numbers can then be reduced by incremental improvements. A SRAC template is shown in the table below.

Table 1: SRAC template

Identifier	Unique identifier
Title	Short title. Useful to promptly recall or sort out the SRAC
Origin	Indication of ongoing activity and e.g., document reference (preferably also if it is a draft document)
Hazard(s)	Indication and reference to related hazard(s)
Receiver	Indication of stakeholder receiving the SRAC and sometimes also link to acceptance of SRAC
Text	Text of the SRAC
Verification	Example of how it might be possible to comply with the SRAC (test, inspection, installation, specific documentation). Examples based on past projects experience can be given if relevant.

3.8. Remote monitoring, operation, and control

If the automated vehicle is monitored remotely, the operator shall demonstrate that the system is able to deliver the same level of safety, control, and response times as if an alert and competent safety driver sat in the vehicle’s driving seat. How the vehicle(s) and Infrastructure Support for Automatic Driving (ISAD), e.g., intelligent road products, are monitored, operated, and controlled must be described. This is usually a separate document. The safety case shall provide sufficient evidence to demonstrate that the safety driver or remote operator can always resume full control of the vehicle and that the minimum risk state can be achieved

3.9. Security

The ISO/SAE 21434 was issued in September 2021. The standard includes requirements for a cybersecurity case. Normally it is sufficient to refer to this cybersecurity case and mentioned relevant limitations, assumptions, and SecRACS. Patching is weakly described in the ISO/SAE 21434. If patching is necessary, one should perform an impact analysis to evaluate whether safety is impacted. When safety is impacted, the safety case has to be updated. If not, a security patch can be performed by the manufacturer. Security patching is described in IEC TR 62443-2-3 "Security for industrial automation and control systems – Part 2-3: Patch management in the IACS environment". It is important to have an agile process for software development to be able to update the software quickly

3.9.1 Security-related application conditions, SecRAC

In addition to relevant vehicle authorisation, this part shall contain references to the safety cases, certificates of any system, subsystems, equipment, or sensors on which the system under consideration depends.

3.10. Assurance of system safety

The system (autonomous vehicle) acceptance is based on the authorized vehicle, including products or items that already have a corresponding safety case or similar. These documents have to be studied carefully and taken into account when planning and performing e.g., system tests and analysis.

In the subchapters below, we have described relevant topics when evaluating an autonomous vehicle.

3.10.1 Manufacturer, vehicle type, authorisation and license number

The supplier of the vehicle has to comply with ordinary information requirements. As a consequence, they have to include information regarding

1. Manufacturer and types
2. Type approval (in Europe according to regulation 2018/858)
3. car license number

3.10.2 Hierarchy of safety cases and related certifications

The different subsystems are often based on related safety cases. Related safety cases of any system are subsystems, items or equipment on which the product or system under consideration depends, see figure below.

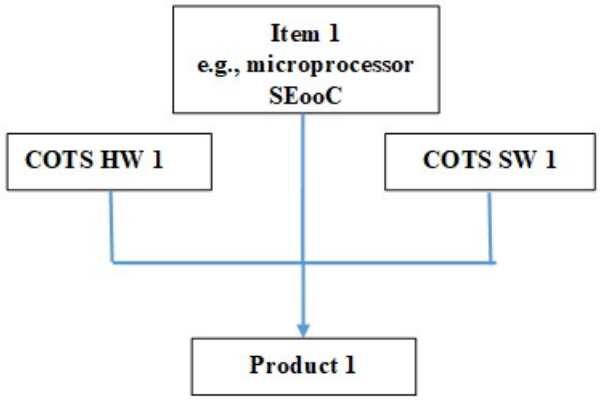

Fig. 1. Hierarchy of related safety cases. SEooC (Safety Element out of Context)

Some of the subsystems, items or equipment may not include a safety case, but they may have a certificate of compliance with e.g., the generic IEC 61508 safety standard.

3.10.3 Vehicle identification number

A vehicle identification number (VIN) (also called a chassis number or frame number) is a unique code presented on the car by the manufacturer, including a serial number used by the automotive industry to identify individual motor vehicles. The standardized information is presented in ISO 3779:2009 "Road vehicles — Vehicle identification number (VIN) — Content and structure".

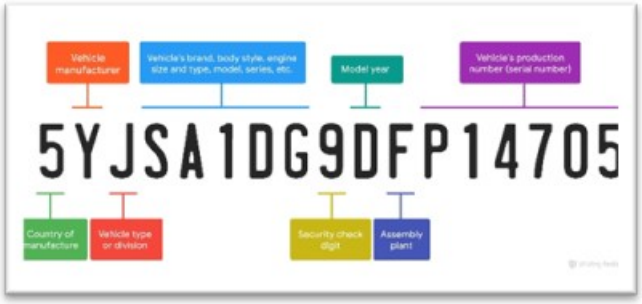

Fig. 2. Vehicle Identification Number

3.10.4 Software identification number

Software version control has become more important since software can be updated during the whole lifetime of the vehicle. Even if it is intended as an improvement, software changes will introduce new behaviour and thus new risks. This implies that the car, or at least the safety related software, should be recertified. However, recertification takes time and before this has been done, the car – and thus its users – is at risk. UNECE has issued a new regulation regarding "RX SoftWare Identification Number (RXSWIN)" (2020) [14] including a document interpreting these requirements [15]. RXSWIN is a dedicated identifier, defined by the vehicle manufacturer, representing information about the type approval relevant software of the Electronic Control System contributing to the Regulation N° X type approval relevant characteristics of the vehicle. For manufacturers of vehicles, the RXSWIN for a specific vehicle is like its “Windows version”, and manufacturers only increment this number when a type approved system is modified.

3.10.5 Vehicle authorisation Summary

In the safety case this information can be summed up as:
1. Manufacturer and vehicle type
2. Type approval
3. Car license number
4. VIN number (ISO 3779:2009)
5. Relevant safety cases and certificates, including security aspects
6. RXSWIN number

3.11. Modelling and simulator studies

The safety case shall include information about modelling and/or simulator testing conducted prior to the trial to support the overall testing program, including, e.g.:
a) the type of model or simulation used
b) the ODD of the simulation
c) details of any limitations of the simulator, including any constraints, assumptions, or imperfections of the simulation environment
d) evidence demonstrating validity and reliability of the test results

3.12. Safety testing and acceptance process

As a starting point, we will assume that the vehicle has been developed and tested according to ISO 26262:2018 series, ISO 21448:2019, and ISO 22737:2021. In addition, relevant on-site tests have to

be performed. Which on-site tests are relevant are based on evaluations of the trial site, relevant uses cases, scenarios, and risk analysis. The acceptance process may vary from country to country.

3.13. Stakeholder consultation and agreement

Communication is one of the key aspects of Scrum and SafeScrum [4]. One of the agile manifestos includes communication:

Customer collaboration and contract negotiation

The safety case shall include a list of stakeholders and detail relevant communication and consultation with the identified organizations. This could include public education and an awareness campaign towards the public. We have performed reviews during autonomous bus trials in 2021 that indicates that the passengers believe self-driving buses should be safer than normal buses. Relevant sources could be the safety case for the public [3] and the trust case [6].

3.14. Monitoring, reporting, and continuous Improvement

Monitoring conducted during the trial shall be described in the safety case. The
safety case shall include:

a) the person(s)/role responsible for data capture processes
b) what data is being collected and how (e.g., dashcams, sensors, cameras, surveys etc.)
c) how sensors, redundancy, and failure modes are monitored
d) how dynamic hazards are monitored, e.g., weather/environment
e) how the data is being downloaded, stored, and analysed
f) the security of data collection, transfer, and storage
g) any parameters (i.e., start, end, sources, etc.) of any logging/monitoring to be stored/saved. This shall also include assurance that data has not been tampered with.
h) the procedure for analysis and reporting of issues that affect trial safety and continuation
i) the name of a nominated person who is responsible for ensuring compliance to the safety case and acting as a single point of contact for any concerns or questions related to the safety case
j) the process for incident and near-miss reporting and analysis

Continuous improvement is one of the main reasons for adapting an agile approach and DevOps (Development and Operation). When doing continuous improvement, it is advantageous to have an agile contract. The Norwegian government for financial management has issued agile contract templates [10].

3.15. Safety case summary and conclusion

This chapter shall summarize the evidence presented in the previous parts of the safety case and present an argument that the system under consideration is adequately safe, subject to compliance with relevant standards, legislation, and the specified safety and security application conditions.
Findings, non-compliances, recommendations, etc., shall be referenced or presented in this chapter.

4. DISCUSSION

As mentioned above, several standards require a safety case, and several standards can be used as a basis for the safety case. For the manufacturer, the important question is whether it is worth it. An important question for people using an agile approach is how difficult it is to include building a safety case into an agile process such as SafeScrum. In our opinion, all projects that develop safety-critical software should build a safety case, if not for the certification, then in order to create a safety culture and to convince oneself that the system really is safe.

A lot of the necessary documentation will have to be written anyway due to standard requirements, but it will still require some extra paperwork and extra activities that do not benefit the customer and thus run counter to the agile manifesto's idea of customer focus.

Reuse of documents and use of document templates will, however, reduce the extra effort needed for building a safety case. Working with the safety case will increase system understanding and will thus lead to a more efficient process. Some agile approaches have been mentioned together with DevOps to ensure improved processes.

4. CONCLUSION

Based on the discussions above, we can make the following important conclusions:

1. We have developed a template (chapter 3) for an agile safety case for trial operations
2. Working with safety cases will increase the stakeholder's safety awareness.
3. A safety cases can be developed incrementally. The trial Safety Case shall be continually updated to provide a record of the progress of the project

Relevant agile approaches are: MVP, incremental development, agile contracts, agile software development process, customer collaboration and DevOps.

Acknowledgements

This work has been supported by The Norwegian Research Council, Project name "TrustMe", project number: 309207 - IPOFFENTLIG19, https://prosjektbanken.forskningsradet.no/#/project/NFR/309207

References

[1] T. Myklebust and T. Stålhane. The Agile Safety Case. ISBN 9783319702643. Springer International Publishing. February 2018.
[2] T. Myklebust, T. Stålhane, Gunnar D. Jenssen and I. Wærø. Autonomous cars, trust and safety case for the public. RAMS 2020, Palm Springs USA.
[3] T. Myklebust, T. Stålhane, G. D. Jenssen and I. Haug. TrustMe, we have a safety case for the public. ESREL 2021 Angers France
[4] G. K. Hanssen, T. Stålhane and T. Myklebust. SafeScrum – Agile Development of Safety-Critical Software. ISBN 9783319993348.Springer. December 2018.
[5] T. Myklebust, T. Stålhane and Geir K. Hanssen. Agile Safety Case and DevOps for the automotive industry. ESREL2020/PSAM15 Venice, Italy 2020
[6] T. Stålhane and T. Myklebust. Trust Case and the link to safety case. SAFE 9th International Conference on Safety and Security Engineering. Rome, Italy 2021
[7] Pettersson L., Ragnevi T. and Olsson E. implementation of agile processes in human factors and safety management – A case study from the Swedish national train management project. International Human Factors Rail 2021
[8] Highways England GG 104: Requirements for safety risk assessment, revision 0 June 2018.
[9] KMPG:2019. Autonomous Vehicle readiness Index. Assessing Countries Preparedness for Autonomous Vehicles. https://home.kpmg/xx/en/home/insights/2019/02/2019-autonomous-vehicles-readiness-index.html
[10] DFØ 2021, seen 2021-11-07. https://anskaffelser.prod.dfo-dev.no/nb/verktoy/maler-ogsa-kontrakt-og-avtalemaler/smidigavtalen-ssa-s
[11] F. Redmill: ALARP explored. Computing science, Newcastle University 2010. Can be downloaded at https://eprints.ncl.ac.uk/161155
[12] Zenzic report: Safety case Framework: The guidance edition. 2021

[13] T. Myklebust and T. Stålhane. Functional safety and proof of compliance. ISBN 978-3-030-86151-3, ISBN 978-3-030-86152-0 (eBook) https://doi.org/10.1007/978-3-030-86152-0. Springer International Publishing. December 2021.
[14] UN Regulation ECE/TRANS/WP.29/2020/80: UN Regulation on uniform provisions concerning the approval of vehicles with regards to software update and software updates management system
[15] Interpretation document: ECE/TRANS/WP.29/2021/60 Proposals for Interpretation Documents for UN Regulation No. 156 on software update and software update management system